

AGGREGATION AND ARITHMETIC OPERATIONS ON FUZZY NATURAL NUMBER-VALUED MULTISSETS

Jaume Casasnovas¹ J. Vicente Riera¹

¹ Univ. of the Balearic Islands, {jaume.casasnovas, jvicente.riera}@uib.es

Abstract

In this paper we propose the extension of some aggregation or arithmetic operations between discrete fuzzy numbers to discrete fuzzy number-valued multisets such as the addition, the union and the intersection and, even, the order. For this reason, we prove or recall some properties of discrete fuzzy numbers and we undertake their study on multisets.

Keywords: Aggregation operators, Multisets, discrete fuzzy number, lattice, monoid.

1 INTRODUCTION

A (crisp) *multiset* over a set of *types* X is a mapping $M : X \rightarrow \mathbb{N}$. A survey of the mathematics of multisets, including their axiomatic foundation, can be found in [1]. In [2], order and triangular operations between multisets are studied. Multisets are also called *bags* in the literature [17].

According to the usual interpretation of a multiset $M : X \rightarrow \mathbb{N}$, it describes a set or *universe*, Ω , which consists of $M(x)$ “exact” copies of each type $x \in X$, without specifying which element of the universe is a copy of which element of X . The number $M(x)$ is usually called the *multiplicity* of x in the multiset M . Notice, in particular, that the set or universe described by the multiset does not contain any element that is not a copy of some $x \in X$, and that an element of it cannot be a copy of two different types. So, each type $x \in X$ defines a subset Ω_x of the universe constituted by their “exact” copies and we have $\Omega_x \cap \Omega_y = \emptyset$, if $x \neq y$ and $\Omega = \bigcup_{x \in X} \Omega_x$. The *multiplicity*, $M(x)$, of x in the multiset M is the *cardinal* of the subset Ω_x .

In [2], the authors introduced a more general definition of “extended multiset” as mappings $M : X \rightarrow L$, where L is a finite or infinite chain of natural numbers, or, even, it can be $\bar{\mathbb{N}} = \mathbb{N} \cup \{\infty\}$, with the usual operations and order. This definition allows to extend several aggregation operators defined in L , such as t-norms or t-conorms, to multisets.

A natural generalization of this interpretation of multisets leads to the notion of *multisets with fuzzy values* [11, 12] over a set of *types* X . Such a multiset describes for each $x \in X$, a set consisting of “possibly inexact” copies of x with a degree of similarity valued in $[0,1]$. In this way, in [12] an immediate generalization of crisp multisets using fuzzy numbers instead of natural numbers is proposed. So, provided a suitable definition of fuzzy number (triangular, trapezoidal, Gauss-shaped, etc [10]), it is possible to consider fuzzy Number-Valued multisets defined over X .

In this paper we impose two restrictions on this interpretation of a fuzzy multiset, parallel to those highlighted in the crisp case, that allows us to slightly modify this definition. First, we assume that if an element of the set is an inexact copy of x with a degree of similarity $t > 0$, then it cannot be an inexact copy of any other type in X with a non-negative degree of similarity. And second, the set or universe described by the fuzzy multiset does not contain any element that is not a copy of some $x \in X$ with some non-negative degree of similarity. These two conditions entail that each type $x \in X$ defines a fuzzy subset Ω_x of the universe Ω constituted by all the copies of x , each one with its degree of similarity with x and we have $\Omega_x \cap \Omega_y = \emptyset$, if $x \neq y$ and $\Omega = \bigcup_{x \in X} \Omega_x$, i.e. $\Omega_x(w) \wedge \Omega_y(w) = 0$ and $\bigvee_{x \in X} \Omega_x(w) = 1$ for all $w \in \Omega$.

In order to define a “multiplicity” or *fuzzy multiplicity* of each type for a *fuzzy multiset* over X , we need to associate to each $x \in X$ the *cardinality* of the fuzzy set Ω_x . The problem of “counting” fuzzy sets has generated a lot of literature since Zadeh’s first definition

of the cardinality of fuzzy sets [9, 10]. In particular, the scalar cardinalities of fuzzy sets, which associate to each fuzzy set a positive real number, have been studied from the axiomatic point of view [8, 16] with the aim of capturing different ways of counting additive aspects of fuzzy sets like the cardinalities of supports, of levels, of cores, etc. In a similar way, the fuzzy cardinalities of fuzzy sets [7, 9, 16], which associate to any fuzzy set a convex fuzzy natural number, have also been studied from the axiomatic point of view.

In this paper, taking into account that the fuzzy cardinality of a fuzzy set is a fuzzy natural number, i.e., a discrete fuzzy number whose support is a subset of consecutive natural numbers, we consider a *Fuzzy Natural Number-Valued multiset* defined over a set X as a mapping $M : X \rightarrow FNN$ where FNN is the set of discrete fuzzy numbers with support a subset of consecutive natural numbers.

In order to define some operations in this framework, the first step is to recall the operations between fuzzy natural numbers. If we know that the set of fuzzy natural numbers is closed with respect to an operation, $\bar{O}$, we can define an operation between multisets functionally: $\bar{O}(M, N)(x) = \bar{O}(M(x), N(x)), \forall x \in X$. Thus, we study the addition, the union and the intersection as functionally defined extensions of operations between fuzzy natural numbers.

2 PRELIMINARIES

2.1 Multisets

Let X be a crisp set. A (*crisp*) *multiset* over X is a mapping $M : X \rightarrow \mathbb{N}$, where $\mathbb{N}$ stands for the set of natural numbers including the 0. A multiset M over X is *finite* if its *support*

$$\text{supp}(M) = \{x \in X \mid M(x) > 0\}$$

is a finite subset of X . We shall denote the sets of all multisets and of all finite multisets over a set X by $MS(X)$ and $FMS(X)$, respectively, and by $\perp$ the *null multiset*, defined by $\perp(x) = 0$ for each $x \in X$.

For every $A, B \in MS(X)$, their *sum* [13] $A + B$ is the multiset defined pointwise by

$$(A + B)(x) = A(x) + B(x), \quad x \in X.$$

Let us mention here that it has been argued that this sum $+$, also called *additive union*, is the right notion of union of multisets. According to the interpretation of multisets as sets of copies of types explained in the introduction, this sum corresponds to the disjoint union of sets, as it interprets that all copies of each x in the set represented by A are different from all copies of it

in the set represented by B . This additive sum has quite different properties from the ordinary union of sets. For instance, the collection of submultisets of a given multiset is not closed under this operation and consequently no sensible notion of complement within this collection exists.

For every $A, B \in MS(X)$, their *join* $A \vee B$ and *meet* $A \wedge B$ are respectively the multisets over X defined pointwise by $(A \vee B)(x) = \max(A(x), B(x))$ and $(A \wedge B)(x) = \min(A(x), B(x))$, $x \in X$. If A and B are finite, then $A + B$, $A \vee B$ and $A \wedge B$ are also finite. A partial order $\leq$ on $MS(X)$ is defined by $A \leq B$ if and only if $A(x) \leq B(x)$ for every $x \in X$. If $A \leq B$, then their *difference* $B - A$ is the multiset defined pointwise by

$$(B - A)(x) = B(x) - A(x).$$

2.2 Discrete Fuzzy Numbers

By a fuzzy subset of the set of real numbers, we mean a function $u : \mathbb{R} \rightarrow [0, 1]$. For each fuzzy subset u , let $u^\alpha = \{x \in \mathbb{R} : u(x) \geq \alpha\}$ for any $\alpha \in (0, 1]$ be its α -level set (or α -cut). By $\text{supp}(u)$, we mean the support of u , i.e. the set $\{x \in \mathbb{R} : u(x) > 0\}$. By u^0 , we mean the closure of $\text{supp}(u)$.

Definition 2.1 [14] A fuzzy subset u of $\mathbb{R}$ with membership mapping $u : \mathbb{R} \rightarrow [0, 1]$ is called *discrete fuzzy number* if its support is finite, i.e., there are $x_1, \dots, x_n \in \mathbb{R}$ with $x_1 < x_2 < \dots < x_n$ such that $\text{supp}(u) = \{x_1, \dots, x_n\}$, and there are natural numbers s, t with $1 \leq s \leq t \leq n$ such that:

1. $u(x_i) = 1$ for any natural number i with $s \leq i \leq t$ (core)
2. $u(x_i) \leq u(x_j)$ for each natural number i, j with $1 \leq i \leq j \leq s$
3. $u(x_i) \geq u(x_j)$ for each natural number i, j with $t \leq i \leq j \leq n$

Remark 2.2 If the fuzzy subset u is a discrete fuzzy number then the support of u coincides with its closure, i.e. $\text{supp}(u) = u^0$.

From now on, the notation *DFN* stands for the set of discrete fuzzy numbers.

Remark 2.3 In general, the operations on fuzzy numbers can be approached either by the direct use of their membership function as fuzzy subsets of $\mathbb{R}$ using the Zadeh's extension principle or by the equivalent use of the α -cuts representation [10]. Nevertheless, if u, v are discrete fuzzy numbers, these processes:

$$(u \oplus v)(z) = \sup_{z=x+y} \min(u(x), v(y)), \forall z \in \mathbb{R}$$

$$MIN(u, v)(z) = \sup_{z=\min(x,y)} \min(u(x), v(y)), \forall z \in \mathbb{R}$$

$$MAX(u, v)(z) = \sup_{z=\max(x,y)} \min(u(x), v(y)), \forall z \in \mathbb{R}$$

can yield fuzzy subsets that do not satisfy the conditions to be discrete fuzzy numbers [3, 15].

In [3, 4, 5, 15], this drawback is studied and a new method to define these operations is proposed. So, let u, v be two discrete fuzzy numbers and $u^\alpha = \{x_1^\alpha, \dots, x_p^\alpha\}$, $v^\alpha = \{y_1^\alpha, \dots, y_k^\alpha\}$ their α -cuts respectively. And, for each pair of subsets $X, Y \subset \mathbb{R}$ and O a binary operation in $\mathbb{R}$, we will consider the set $XOY = \{z = xOy, x \in X, y \in Y\}$. Then, the next result holds [15]:

Theorem 2.4 *Let $u, v \in DFN$, the fuzzy subset denoted by $u \oplus_W v$, such that it has as r -cuts the sets $[u \oplus_W v]^r = \{x \in \text{supp}(u) + \text{supp}(v) : \min([u]^r + [v]^r) \leq x \leq \max([u]^r + [v]^r)\}$ for each $r \in [0, 1]$ where $\min([u]^r + [v]^r) = \min\{x : x \in [u]^r + [v]^r\}$, $\max([u]^r + [v]^r) = \max\{x : x \in [u]^r + [v]^r\}$ and $(u \oplus_W v)(x) = \sup\{r \in [0, 1] \text{ such that } x \in [u \oplus_W v]^r\}$ is a discrete fuzzy number.*

On the other hand, in [5], for each $u, v \in DFN$ the following sets are considered:

$$MIN_w(u, v)^\alpha = \{z \in \text{supp}(u) \bigwedge \text{supp}(v) \text{ such that } \min(x_1^\alpha, y_1^\alpha) \leq z \leq \min(x_p^\alpha, y_k^\alpha)\} \text{ and}$$

$$MAX_w(u, v)^\alpha = \{z \in \text{supp}(u) \bigvee \text{supp}(v) \text{ such that}$$

$$\max(x_1^\alpha, y_1^\alpha) \leq z \leq \max(x_p^\alpha, y_k^\alpha)\} \text{ for each } \alpha \in [0, 1] \text{ where } \text{supp}(u) \bigwedge \text{supp}(v) = \{z = \min(x, y) | x \in \text{supp}(u), y \in \text{supp}(v)\} \text{ and } \text{supp}(u) \bigvee \text{supp}(v) = \{z = \max(x, y) | x \in \text{supp}(u), y \in \text{supp}(v)\}.$$

And the following result is obtained:

Proposition 2.5 [5] *For each $u, v \in DFN$, there exist two unique discrete fuzzy numbers, which we will denote by $MIN_w(u, v)$ and $MAX_w(u, v)$, such that they have the above defined sets $MIN_w(u, v)^\alpha$ and $MAX_w(u, v)^\alpha$ as α -cuts respectively.*

3 Addition of Fuzzy Natural Numbers

Now, we wish to study some properties of the addition of fuzzy natural numbers (discrete fuzzy numbers whose support is a subset of consecutive natural numbers).

It is well known [10] that, in the case of continuous fuzzy numbers the addition obtained by extending the

usual addition of real numbers through the extension principle is associative and commutative. But the fuzzy natural numbers are not continuous on $\mathbb{R}$.

In [4], the authors proved that in the case in which the discrete fuzzy numbers have as support an arithmetic sequence or a subset of consecutive natural numbers it is possible to use the Zadeh's extension principle to obtain its addition. Moreover, we know [15] the next result:

Proposition 3.1 *Let's consider $u, v \in DFN$. If $u \oplus v \in DFN$ where $u \oplus v$ denotes the addition of u and v using the Zadeh's extension principle, then $u \oplus v$ and $u \oplus_W v$ are identical, where $u \oplus_W v$ is the discrete fuzzy number obtained from u and v using theorem 2.4.*

Remark 3.2 *A consequence of the previous proposition is that if we prove a property for the operation $\oplus$ in the set of fuzzy natural numbers FNN , we will obtain the same property for the operation $\oplus$ in this set.*

From now on, the notation fnn stands for a fuzzy natural number.

Lemma 3.3 *Let u, v and w be three fnn such that their supports are $\text{supp}(u)$, $\text{supp}(v)$ and $\text{supp}(w)$ respectively. The following properties hold:*

1. *Commutativity*

$$\text{supp}(u) + \text{supp}(v) = \text{supp}(v) + \text{supp}(u)$$

2. *Associativity*

$$\begin{aligned} (\text{supp}(u) + \text{supp}(v)) + \text{supp}(w) &= \\ &= \text{supp}(u) + (\text{supp}(v) + \text{supp}(w)) \end{aligned}$$

Proof Let's denote by X, Y and Z the sets $\text{supp}(u), \text{supp}(v)$ and $\text{supp}(w)$ respectively.

1. Using the commutative property of the addition of real numbers the proof is straightforward.
2. Associativity

If $z \in (X + Y) + Z$ then $z = x + c$ where $x = a + b$, $a \in X, b \in Y$ and $c \in Z$. So $z = (a + b) + c$. Using the associativity of the addition of real numbers, $z = (a + b) + c = a + (b + c)$. Then $z \in X + (Y + Z)$. Therefore

$$(X + Y) + Z \subseteq X + (Y + Z)$$

If $z \in X + (Y + Z)$ then $z = a + x$ where $x = b + c$, $a \in X, b \in Y$ and $c \in Z$. So $z = a + (b + c)$. Using

the associativity of the addition of real numbers, $z = a + (b + c) = (a + b) + c$. Then $z \in (X + Y) + Z$. Therefore

$$X + (Y + Z) \subseteq (X + Y) + Z$$

Theorem 3.4 For any, $u, v, w \in FNN$, the following properties hold:

a) Commutativity:

$$u \oplus_W v = v \oplus_W u$$

b) Associativity:

$$(u \oplus_W v) \oplus_W w = u \oplus_W (v \oplus_W w)$$

c) Neutral element, i.e., $u \oplus_W \hat{0} = u$ for all $u \in DFN$

$$\text{where } \hat{0} \text{ is the dfn } \hat{0}(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{if } x \neq 0 \end{cases}$$

Proof: Let u, v and w be three fnn. Let's consider the α -cut sets: $u^\alpha = \{x_1^\alpha, \dots, x_p^\alpha\}$, $v^\alpha = \{y_1^\alpha, \dots, y_k^\alpha\}$, $w^\alpha = \{w_1^\alpha, \dots, w_l^\alpha\}$ for u, v and w respectively.

a) We want to show that

$$u \oplus_W v = v \oplus_W u$$

It is enough to prove that the fnn $u \oplus_W v$ and $v \oplus_W u$ are the same α -cut sets for each $\alpha \in [0, 1]$, where $(u \oplus_W v)^0$ denotes the support of $u \oplus_W v$.

By definition $(u \oplus_W v)^\alpha = \{z \in \text{supp}(u) + \text{supp}(v) \text{ such that}$

$$\min(u^\alpha + v^\alpha) \leq z \leq \max(u^\alpha + v^\alpha)\} =$$

(by the monotonicity of the addition of natural numbers) $= \{z \in \text{supp}(u) + \text{supp}(v) \text{ such that}$

$$\begin{aligned} &(\min u^\alpha + \min v^\alpha) \leq z \leq (\max u^\alpha + \max v^\alpha)\} = \\ &\{z \in \text{supp}(u) + \text{supp}(v) | (x_1^\alpha + y_1^\alpha) \leq z \leq (x_p^\alpha + y_k^\alpha)\} = \\ &\{z \in \text{supp}(u) + \text{supp}(v) | (y_1^\alpha + x_1^\alpha) \leq z \leq (y_p^\alpha + x_k^\alpha)\} = \\ &\quad (v \oplus_W u)^\alpha \end{aligned}$$

b) We want to see that

$$(u \oplus_W v) \oplus_W w = u \oplus_W (v \oplus_W w)$$

By definition $((u \oplus_W v) \oplus_W w)^\alpha = \{z \in \text{supp}(u \oplus_W v) + \text{supp}(w) \text{ such that}$

$$\min((u \oplus_W v)^\alpha + w^\alpha) \leq z \leq \max((u \oplus_W v)^\alpha + w^\alpha)\} =$$

(by the monotonicity of the addition of natural numbers) $= \{z \in \text{supp}(u \oplus_W v) + \text{supp}(w) \text{ such that}$

$$\begin{aligned} &\min(u \oplus_W v)^\alpha + \min w^\alpha \leq z \leq \max(u \oplus_W v)^\alpha + \max w^\alpha\} = \\ &\{z \in \text{supp}(u \oplus_W v) + \text{supp}(w) \text{ such that} \end{aligned}$$

$$(x_1^\alpha + y_1^\alpha) + w_1^\alpha \leq z \leq (x_p^\alpha + y_k^\alpha) + w_l^\alpha\} =$$

$\{z \in (\text{supp}(u) + \text{supp}(v)) + \text{supp}(w) \text{ such that}$

$$(x_1^\alpha + y_1^\alpha) + w_1^\alpha \leq z \leq (x_p^\alpha + y_k^\alpha) + w_l^\alpha\} =$$

$= \{z \in (\text{supp}(u) + \text{supp}(v)) + \text{supp}(w) \text{ such that}$

$$x_1^\alpha + (y_1^\alpha + w_1^\alpha) \leq z \leq x_p^\alpha + (y_k^\alpha + w_l^\alpha)\} =$$

$\{z \in \text{supp}(u) + (\text{supp}(v) + \text{supp}(w)) \text{ such that}$

$$x_1^\alpha + (y_1^\alpha + w_1^\alpha) \leq z \leq x_p^\alpha + (y_k^\alpha + w_l^\alpha)\} =$$

$\{z \in \text{supp}(u) + (\text{supp}(v \oplus_W w)) \text{ such that}$

$$\begin{aligned} &x_1^\alpha + (y_1^\alpha + w_1^\alpha) \leq z \leq x_p^\alpha + (y_k^\alpha + w_l^\alpha)\} = \\ &= (u \oplus_W (v \oplus_W w))^\alpha \end{aligned}$$

c) Neutral element: $(u \oplus_W \hat{0})(x) = (u \oplus_W \hat{0})(x + 0) =$

$$\sup_{z=x+0}(\min(u(x), \hat{0}(0))) =$$

$$\sup_{z=x+0}(\min(u(x), 1)) = u(x) \text{ for all } x \in \text{supp}(u).$$

Corollary 3.5 The set FNN of the fuzzy natural numbers is a commutative monoid with the Zadeh's addition as a monoidal operation.

4 Maximum and Minimum of Fuzzy Natural Numbers

With respect to the maximum and the minimum of two fuzzy natural numbers, the authors have proved in [5] the following proposition:

Proposition 4.1 [5] Let u, v be two fuzzy natural numbers. Then $MAX(u, v)$, defined through the extension principle, coincides with $MAX_w(u, v)$. So, if $u, v \in FNN$, $MAX(u, v)$ is a fuzzy natural number and $MAX(u, v) \in FNN$. Analogously, $MIN(u, v)$, defined through the extension principle, coincides with $MIN_w(u, v)$. So, if $u, v \in FNN$, then $MIN(u, v)$ is a fuzzy natural number and $MIN(u, v) \in FNN$.

But we have studied in [6] the associativity, commutativity, idempotence, absorption and distributivity for the operations MIN_w and MAX_w between discrete fuzzy numbers in general and between fuzzy natural numbers in particular and we obtained the following proposition:

Proposition 4.2 [6] *The set of discrete fuzzy numbers whose support is a sequence of consecutive natural numbers (FNN, MIN_w, MAX_w) is a distributive lattice.*

If we gather the previous propositions 4.1 and 4.2, then we obtain the following proposition:

Proposition 4.3 [6] *The set of discrete fuzzy numbers whose support is a sequence of consecutive natural numbers (FNN, MIN, MAX) is a distributive lattice.*

With the aim of studying the monotony for the addition of two fuzzy natural numbers, we need a definition of order:

Definition 4.4 [6] *Using the operations MIN_w and MAX_w , we can define a partial order on FNN on the following way:*

$u \preceq v$ if and only if $MIN_w(u, v) = u$, or equivalently, $u \preceq v$ if and only if $MAX_w(u, v) = v$ for any $u, v \in FNN$. Equivalently, we can also define the partial ordering in terms of α -cuts:

$u \preceq v$ if and only if $\min(u^\alpha, v^\alpha) = u^\alpha$

$u \preceq v$ if and only if $\max(u^\alpha, v^\alpha) = v^\alpha$

Proposition 4.5 *Let $u, v, w, t \in FNN$. If $u \preceq v$ and $w \preceq t$ where $\preceq$ denotes the partial order in FNN defined in definition 4.4 then $u \oplus w \preceq v \oplus t$, where $\oplus$ denotes the Zadeh's addition.*

Proof: From [6], we know that $u \preceq v$ iff $MIN(u, v) = u$ iff $\min(u^\alpha, v^\alpha) = u^\alpha$ for all $\alpha \in [0, 1]$. And analogously, $w \preceq t$ iff $MIN(w, t) = w$ iff $\min(w^\alpha, t^\alpha) = w^\alpha$ for all $\alpha \in [0, 1]$.

Let $u^\alpha = \{a_1^\alpha, \dots, a_k^\alpha\}$, $v^\alpha = \{b_1^\alpha, \dots, b_m^\alpha\}$, $w^\alpha = \{c_1^\alpha, \dots, c_n^\alpha\}$ and $t^\alpha = \{d_1^\alpha, \dots, d_p^\alpha\}$ the α -cuts of u, v, w and t respectively for $\alpha \in [0, 1]$. By definition of the addition of natural fuzzy numbers [4] the α -cuts of $u \oplus w$ and $v \oplus t$ are the sets of consecutive natural numbers $(u \oplus w)^\alpha = \{a_1^\alpha + c_1^\alpha, \dots, a_k^\alpha + c_n^\alpha\}$ and $(v \oplus t)^\alpha = \{b_1^\alpha + d_1^\alpha, \dots, b_m^\alpha + d_p^\alpha\}$.

Now, we want to prove that $MIN(u \oplus w, v \oplus t) = u \oplus w$, or equivalently [6], $MIN(u \oplus w, v \oplus t)^\alpha = (u \oplus w)^\alpha$ for all $\alpha \in [0, 1]$.

So, $MIN(u \oplus w, v \oplus t)^\alpha = \{z \in \text{supp}(u \oplus w) \wedge \text{supp}(v \oplus t) : \min((u \oplus w)^\alpha \wedge (v \oplus t)^\alpha) \leq z \leq \max((u \oplus w)^\alpha \wedge (v \oplus t)^\alpha)\}$

$\} = \{z \in \text{supp}(u \oplus w) \wedge \text{supp}(v \oplus t) : a_1^\alpha + c_1^\alpha \leq z \leq a_k^\alpha + c_n^\alpha\}$ (as $u \oplus w$ has a set of consecutive natural numbers as a support and the addition is a monotone operation) $= \{z \in \text{supp}(u \oplus w) : a_1^\alpha + c_1^\alpha \leq z \leq a_k^\alpha + c_n^\alpha\} = (u \oplus w)^\alpha$.

5 Operations on FNN-VALUED MULTISETS

Definition 5.1 *A Fuzzy Natural Number-valued multiset defined over an universe X is a mapping $M : X \rightarrow FNN$ i.e. for all $x \in X$, $M(x)$ is a fuzzy natural number.*

Remark 5.2 *We will denote the set of Fuzzy Natural Number-valued multisets defined over an universe X by $FNNM(X)$. Finally, the abbreviation $fnnm$ will denote a Fuzzy Natural Number-valued multiset.*

The properties of the addition of fuzzy natural numbers studied in the previous section, will allow us to define the addition of fuzzy natural number-valued multisets and to study the monoidal structure of this set.

5.1 Monoid structure

Definition 5.3 *Let $A, B : X \rightarrow FNN$ be two Fuzzy Natural Number-valued multisets. The sum of A and B will be the Fuzzy Natural Number-valued Multiset pointwise defined for all $x \in X$ by*

$$(A + B)(x) = A(x) \oplus B(x)$$

where the fnn $A(x) \oplus B(x)$ is obtained following the Zadeh's extension principle or equivalently using the method considered in theorem 2.4.

Proposition 5.4 *The set $FNNM(X)$ of the fuzzy natural number-valued multisets over X is a commutative monoid with the addition as a monoidal operation.*

Proof: The proof is straightforward from the corollary 3.5 and the definition of addition of fuzzy natural number-valued multisets 5.3.

5.2 Lattice structure and order

Analogously to the addition, the properties of the maximum and minimum of fnn studied in the previous section will allow us to define the maximum and minimum of fuzzy natural number-valued multisets and to study the order and the lattice structure of this set.

Definition 5.5 *Let $A, B : X \rightarrow FNN$ be two Fuzzy Natural Number-valued Multisets. The join and the*

meet of A and B will be the Fuzzy Natural Number-valued Multiset, pointwise defined for all $x \in X$ as

$$(A \vee B)(x) = \text{MAX}\{A(x), B(x)\}$$

$$(A \wedge B)(x) = \text{MIN}\{A(x), B(x)\}$$

respectively,

where the fuzzy natural numbers $\text{MAX}\{A(x), B(x)\}$ and $\text{MIN}\{A(x), B(x)\}$ are obtained according to the method presented in proposition 2.5.

Proposition 5.6 *As long as, for all $x \in X$, $A(x) \in \text{FNN}$ and $B(x) \in \text{FNN}$, then $\text{MAX}\{A(x), B(x)\}$ and $\text{MIN}\{A(x), B(x)\}$ can be obtained by means of the extension principle.*

Proof: By proposition 4.1, if $A(x) \in \text{FNN}$ and $B(x) \in \text{FNN}$, then the extension principle yields a discrete fuzzy number that it coincides with $\text{MAX}\{A(x), B(x)\} \in \text{FNN}$ (or $\text{MIN}\{A(x), B(x)\}$).

Proposition 5.7 *Let $A, B : X \rightarrow \text{FNN}$ be two Fuzzy Natural Number-valued Multisets. The binary relationship:*

$A \leq B$ *if and only if $A \vee B = B$ and/or $A \wedge B = A$ i.e. $\text{MAX}\{A(x), B(x)\} = B(x), \forall x \in X$ (or $\text{MIN}\{A(x), B(x)\} = A(x), \forall x \in X$) is a partial order on the set $\text{FNNM}(X)$.*

Proposition 5.8 *The set $\text{FNNM}(X)$ of the fuzzy natural number-valued multisets over X is a lattice with the partial order defined in proposition 5.7 and the meet and join operations proposed in definition 5.5.*

Proof: The proof is straightforward from the lattice structure considered in proposition 4.2.

Acknowledgements

We would like to express our thanks to anonymous reviewers who have contributed to improve this article. This work has been partially supported by the MTM2009-10962 project grant.

References

- [1] W.D. Blizard. The development of multiset theory. *Modern logic*, pp. 319-352, 1991.
- [2] J. Casasnovas, G. Mayor. Discrete t-norms and operations on extended multisets. *Fuzzy sets and Systems*(159), pp. 1165-1177, 2008.
- [3] J. Casasnovas, J. Vicente Riera. On the addition of discrete fuzzy numbers. *Wseas Transactions on Mathematics*, pp. 549-554, 2006.
- [4] J. Casasnovas, J. Vicente Riera. Discrete fuzzy numbers defined on a subset of natural numbers. *Theoretical Advances and Applications of Fuzzy Logic and Soft Computing: Advances in Soft Computing*(42), pp. 573-582, 2007.
- [5] J. Casasnovas, J. Vicente Riera. Maximum and minimum of discrete fuzzy numbers. *Frontiers in Artificial Intelligence and Applications: artificial intelligence research and development* (163), pp. 273-280, 2007.
- [6] J. Casasnovas, J. Vicente Riera. Lattice properties of discrete fuzzy numbers under extended min and max. *Proceedings IFSA-EUSFLAT*, pp. 647-652, 2009.
- [7] J. Casasnovas, J. Torrens. An Axiomatic Approach to the fuzzy cardinality of finite fuzzy sets. *Fuzzy Sets and Systems* (133), pp. 193-209, 2003.
- [8] J. Casasnovas, J. Torrens. Scalar cardinalities of finite fuzzy sets for t-norms and t-conorms. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* (11), pp. 599-615, 2003.
- [9] D. Dubois. A new definition of the fuzzy cardinality of finite sets preserving the classical additivity property. *Bull. Stud. Exch. Fuzziness Appl.(BUSEFAL)*(5), pp. 11-12, 1981.
- [10] George Klir, Yuan Bo. *Fuzzy sets and fuzzy logic (Theory and applications)*, Prentice Hall, 1995.
- [11] B. Li. Fuzzy Bags and Applications. *Fuzzy Sets and Systems*(34), pp. 61-71, 1990.
- [12] S. Miyamoto. Remarks on basics of fuzzy sets and fuzzy multisets. *Fuzzy Sets and Systems*(155), pp. 426-431, 2005.
- [13] A. Syropoulos. Mathematics of Multisets. *Multiset Processing LNCS* (2235), pp. 154-160, 2001.
- [14] W. Voxman. Canonical representations of discrete fuzzy numbers. *Fuzzy Sets and Systems*(54), pp. 457-466, 2001.
- [15] Guixiang Wang, Cong Wu, Chunhui Zhao. Representation and Operations of discrete fuzzy numbers. *Southeast Asian Bulletin of Mathematics*(28), pp. 1003-1010, 2005.
- [16] M. Wygalak. An axiomatic approach to scalar cardinalities of fuzzy sets. *Fuzzy Sets and Systems*(110), pp. 175-179, 2000.
- [17] R. Yager. On the theory of bags. *Inter. J. General Systems*(13), pp. 23-37, 1986.

MEANS ON DISCRETE PRODUCT LATTICES

Gleb Beliakov¹ Humberto Bustince² Javier Fernandez² Daniel Paternain²

¹ School of Information Technology, Deakin University, Australia gleb@deakin.edu.au

² Dept. Automatics and Computation, UPNA, Pamplona, Spain {bustince, fcojavier.fernandez, daniel.paternain}@unavarra.es

Abstract

We investigate the problem of averaging values on lattices, and in particular on discrete product lattices. This problem arises in image processing when several color values given in RGB, HSL, or another coding scheme, need to be combined. We show how the arithmetic mean and the median can be constructed by minimizing appropriate penalties. We also discuss which of them coincide with the Cartesian product of the standard mean and median.

Keywords: Aggregation Functions, Mean, Median, Lattice.

1 INTRODUCTION

The need to aggregate several inputs into a single representative output arises in many practical applications. In image processing, it is often necessary to average the values of several neighboring pixels (to reduce the image size or apply a filter), or average pixel values in two different but related images (e.g., in stereovision). When the images are in color, typically coded as discrete RGB, CMY, or HSL values, then it is customary to average the values in the respective channels. Nevertheless, it would be of interest to find other ways to average color values.

In this paper we study averaging on product lattices (RGB or another color coding scheme is an example of a product lattice). We note previous works related to triangular norms on posets and lattices [5, 10] and on discrete chains [11]. In these works, associativity was required. In the setting we are going to develop, however, we will require an averaging behavior of the aggregation functions to be considered. We consider

this can be a key requirement, specially from the point of view of image processing applications.

The structure of the paper is as follows. In Section 2 we provide preliminary definitions. In Section 3 we give the definitions of aggregation functions based on penalties, defined on product lattices. We discuss solutions to resulting optimization problems in Section 4. Conclusions are presented in Section 5.

2 PRELIMINARIES

2.1 Aggregation functions

The research effort concerning aggregation functions, their behavior and properties, has been disseminated throughout various fields including decision making, knowledge based systems, artificial intelligence and image processing. Recent books providing a comprehensive overview include [1, 3, 7, 13].

Definition 1 A function $f : [a, b]^n \rightarrow [a, b]$ is called an aggregation function if it is monotone non-decreasing in each variable and satisfies $f(\mathbf{a}) = a$, $f(\mathbf{b}) = b$, with $\mathbf{a} = (a, a, \dots, a)$, $\mathbf{b} = (b, b, \dots, b)$.

Definition 2 An aggregation function f is called averaging if it is bounded by the minimum and maximum of its arguments

$$\begin{aligned} \min(\mathbf{x}) &:= \min(x_1, \dots, x_n) \leq f(x_1, \dots, x_n) \\ &\leq \max(x_1, \dots, x_n) =: \max(\mathbf{x}). \end{aligned}$$

It is immediate (because of monotonicity) that averaging aggregation functions are idempotent (i.e., $\forall t \in [a, b] : f(t, t, \dots, t) = t$) and vice versa. Then clearly the boundary conditions $f(\mathbf{a}) = a$, $f(\mathbf{b}) = b$ are satisfied.

Well known examples of averaging functions are the arithmetic mean and the median. It is known that the

arithmetic means and the median are solutions to simple optimization problems, in which a measure of disagreement between the inputs is minimized (see [13]), [2, 4, 6, 14]. The main motivation is the following. Let $\mathbf{x}$ be the inputs and y the output. If all the inputs coincide $x = x_1 = \dots, x_n$, then the output is $y = x$, and we have a unanimous vote. If some input $x_i \neq y$, then we impose a “penalty” for this disagreement. The larger the disagreement, and the more inputs disagree with the output, the larger (in general) is the penalty. We look for an aggregated value which minimizes the penalty.

Thus we need to define a suitable measure of disagreement, or dissimilarity.

Definition 3 Let $P : [a, b]^{n+1} \rightarrow \mathfrak{R}$ be a penalty function with the properties

- i) $P(\mathbf{x}, y) \geq 0$ for all $\mathbf{x}, y$;
- ii) $P(\mathbf{x}, y) = 0$ if all $x_i = y$;
- iii) $P(\mathbf{x}, y)$ is quasiconvex in y for any $\mathbf{x}$, i.e., $P(\mathbf{x}, \lambda y_1 + (1 - \lambda)y_2) \leq \max(P(\mathbf{x}, y_1), P(\mathbf{x}, y_2))$ for all $\lambda, y_1, y_2 \in [0, 1]$.

Observe that a quasiconvex function attains its minimum either at a single point or for a whole interval. The penalty based function is

$$f(\mathbf{x}) = \arg \min_y P(\mathbf{x}, y),$$

if y is the unique minimizer, and $y = \frac{a+b}{2}$ if the set of minimizers is the interval $[a, b]$.

In [2] it was shown that any averaging aggregation function can be represented as a penalty based function. Further, the classical means, such as the arithmetic and the median are represented via the following penalty functions. The arithmetic mean is the solution to

$$\text{minimize}_y \sum_{i=1}^n (x_i - y)^2$$

whereas the median is a solution to

$$\text{minimize}_y \sum_{i=1}^n |x_i - y|.$$

In this work we will deal with penalty based functions defined on lattices, rather than the interval $[a, b]$.

2.2 Lattices

Definition 4 Let L be a set. A lattice $\mathcal{L} = (L, \leq, \wedge, \vee)$ is a poset with the partial order $\leq$ on L , and meet and join operations $\wedge, \vee$, if every pair of elements from L has both meet and join.

Definition 5 Let P be a poset. A chain in P is a totally ordered subset of P . The length of a chain is its cardinality.

Definition 6 If $\mathcal{L}_1 = (L_1, \leq_1, \wedge_1, \vee_1)$ and $\mathcal{L}_2 = (L_2, \leq_2, \wedge_2, \vee_2)$ are two lattices, their Cartesian product is the lattice $\mathcal{L}_1 \times \mathcal{L}_2 = (L_1 \times L_2, \leq, \wedge, \vee)$ with $\leq$ defined by

$$(x_1, y_1) \leq (x_2, y_2) \Leftrightarrow x_1 \leq_1 x_2 \text{ and } y_1 \leq_2 y_2$$

and

$$(x_1, y_1) \wedge (x_2, y_2) = (x_1 \wedge_1 x_2, y_1 \wedge_2 y_2)$$

$$(x_1, y_1) \vee (x_2, y_2) = (x_1 \vee_1 x_2, y_1 \vee_2 y_2)$$

We will deal with Cartesian products of finite chains $\mathcal{C}$, which is precisely the type of product lattice representing colors in image processing, with the length of each chain typically being 256. We note that all finite chains of the same length are isomorph to each other, and hence we can represent them as non-negative integers $0, 1, \dots, K$, and elements of product lattices as tuples $x = (x_1, x_2, \dots, x_m)$, $x_i \in \mathcal{C}_i$.

Definition 7 Let f_1, f_2 be two aggregation functions defined on sets X_1 and X_2 respectively. The Cartesian product of aggregation functions is $f = f_1 \times f_2 : X_1 \times X_2 \rightarrow Y_1 \times Y_2$ defined by

$$f(x_1, x_2) = (f_1(x_1), f_2(x_2)).$$

Observe that this definition is given from a strict mathematical point of view, so no interaction or relation is assumed between f_1 and f_2 .

3 MAIN DEFINITIONS

Following representations of the arithmetic mean and the median as penalty-based aggregation functions, we now define similar constructions on lattices.

Definition 8 Let $\mathcal{L} = (L, \leq, \wedge, \vee)$ be a product lattice of chains. The distance between $x, y \in \mathcal{L}$ is defined as the length of a chain $\mathcal{C}$ with the least element $a = x \wedge y$ and the greatest element $b = x \vee y$ minus 1,

$$d(x, y) = \text{length}(\mathcal{C}) - 1$$

We note that all the chains in Definition 8 have the same length, since we are dealing with product of chains. This definition is equivalent to the following

$$d(x, y) = \sum_{i=1}^m d_i(x_i, y_i) = \sum_{i=1}^m |x_i - y_i|,$$

where d_i is the distance on the i -th chain.

Definition 9 Let $\mathcal{L}$ be a product lattice. Consider n elements $x_1, \dots, x_n \in \mathcal{L}$, that need to be averaged. Let the penalty function be $P : Z_+^n \rightarrow \mathbb{R}_+$. The penalty based function on $\mathcal{L}$ is f given by

$$f(x_1, \dots, x_n) = \mu = \arg \min_y P(d(x_1, y), d(x_2, y), \dots, d(x_n, y)),$$

where μ is rounded to the closest value in the lattice, if necessary.

Note that the minimum always exists. Also note that there can be several minimizers (see Example 1 in Section 4 for an example). From an applied point of view, these means that there are several elements which average the given ones. In this case one can take any minimizer. Finally, f is not necessarily monotone, i.e., an aggregation function.

Theorem 1 The function in Definition 9, is an averaging (and hence idempotent) function.

PROOF. Clearly $\bigwedge x_i = a \leq \mu \leq b = \bigvee x_i$, because for any x_i , $d(x_i, a) > d(x_i, t)$ with $t < a$, and similarly at the other end. Also f is idempotent, since every averaging function is. $\square$

A special case of penalty based functions was considered in [12], called dissimilarity functions, where the penalty P is given by

$$P(\mathbf{x}, y) = \sum_{i=1}^n K(x_i - y), \quad (1)$$

where K is a convex function with the unique minimum $K(0) = 0$. In this case the penalty based function is monotone, i.e., an aggregation function. By adapting this definition to our case we have

Theorem 2 The function in Definition 9, with P given by

$$P(\mathbf{x}, y) = \sum_{i=1}^n K(d(x_i, y)),$$

(rounded if necessary) is an averaging aggregation function on a lattice.

PROOF. We only need to prove monotonicity. The proof follows the same lines as in [12], see also [2]. A convex function K , increasing on $[0, \infty)$ has the property

$$K(u) - K(v) \leq K(u') - K(v')$$

if $u - v \leq u' - v'$ and $u, v, u', v' \geq 0$, $u' > u$. Consider $\mathbf{x}$ and $\mathbf{x}'$, such that $x'_{ij} = x_{ij}$ for all i, j except one pair, $x'_{kl} = x_{kl} + 1$. We need to show that if $P(\mathbf{x}, y^*) \leq P(\mathbf{x}, y)$ for all $y \in \mathcal{L}$, then $P(\mathbf{x}', y^*) \leq P(\mathbf{x}', y)$ for all $y < y^*$, that is, the argument minimizing our function should increase.

Suppose $y < y^*$. Take

$$u = \sum_{j=1}^m = \sum_{j \neq l} |x_{kj} - y_j^*| + |x_{kl} + 1 - y_l^*|,$$

$$v = \sum_{j=1}^m |x_{kj} - y_j^*|,$$

$$u' = \sum_{j \neq l} |x_{kj} - y_j| + |x_{kl} + 1 - y_l|$$

and

$$v' = \sum_{j=1}^m |x_{kj} - y_j|.$$

Now,

$$u - v = |x_{kl} + 1 - y_l^*| - |x_{kl} - y_l^*|,$$

which is equal to $+1$ if $y_l^* \leq x_{kl}$ or -1 otherwise. Also

$$u' - v' = |x_{kl} + 1 - y_l| - |x_{kl} - y_l|$$

which is also either $+1$ or -1 , but because $y < y^*$, it takes value -1 only if $u - v = -1$. Hence $u - v \leq u' - v'$.

Now from $P(\mathbf{x}, y^*) \leq P(\mathbf{x}, y)$ we have

$$P(\mathbf{x}', y^*) = \sum_{i \neq k} K(d(x_i, y^*)) + K(u)$$

$$\leq \sum_{i \neq k} K(d(x_i, y)) + K(u') = P(\mathbf{x}', y)$$

$\square$

Below we provide definitions for some specific instances of penalty based aggregation, based on the analogs with the classical means. In all cases we have penalties in the form (1), so Theorem 2 applies. We also round the results when necessary.

Definition 10 Let $P(\mathbf{x}, y) = \sum_{i=1}^n d(x_i, y)^2$. Then the resulting aggregation function is the arithmetic mean.

Definition 11 Let $P(\mathbf{x}, y) = \sum_{i=1}^n w_i d(x_i, y)^2$, and $\mathbf{w}$ be a weighting vector, $w_i \geq 0$, $\sum w_i = 1$. Then the resulting aggregation function is a weighted arithmetic mean.

Definition 12 Let $P(\mathbf{x}, y) = \sum_{i=1}^n d(x_i, y)$. Then the resulting aggregation function is the median.

4 SOLUTION TO PENALTY MINIMIZATION PROBLEMS

Consider now how to obtain solution to minimization problem in Definition 9. First, consider the arithmetic mean. We have the problem

$$\text{minimize}_y \sum_{i=1}^n \left(\sum_{j=1}^m |x_{ij} - y_j| \right)^2,$$

where x_{ij} denotes the j -th component of the i -th tuple $x_i \in \mathcal{L}$. We note that this problem is convex in y . We also note that the solution is different from the Cartesian product of the means, as the following example illustrates, and the differences are not just due to the rounding problem.

Example 1 Let $\mathcal{L}$ be the product of two chains $0, \dots, 10$. Take the mean of $(10, 10), (8, 0), (3, 2)$. The Cartesian product of means gives $(7, 4)$, with the objective value $9^2 + 5^2 + 6^2 = 142$. The solutions to the minimization problem are $(9, 2)$ with the objective $9^2 + 3^2 + 6^2 = 126$ and $(8, 3)$ with the same objective.

While we could not obtain a closed form solution, we note that starting from any $y \in \mathcal{L}$, and in particular the Cartesian product of means or the medians, and performing coordinate descent (because of the convexity of the objective), one can reach the minimum algorithmically.

For the median consider the problem

$$\text{minimize}_y \sum_{i=1}^n \sum_{j=1}^m |x_{ij} - y_j| = \sum_{j=1}^m \sum_{i=1}^n |x_{ij} - y_j|.$$

Each term in the inner sum depend on y_j only, thus the problem is equivalent to m separate problems

$$\min_{y_j} \sum_{i=1}^n |x_{ij} - y_j|.$$

The solution is the median function. Hence the minimum is achieved at $y = (\text{Med}(x_{\cdot,1}), \dots, \text{Med}(x_{\cdot,m}))$, i.e., the result is the Cartesian product of the medians.

5 CONCLUSIONS

Based on the representation of the classical mean and the median as solution to minimization problems, we have defined the mean and the median on discrete product lattices in the same way. We have shown that the median becomes the Cartesian product of the medians defined on discrete chains, and that the mean is not the Cartesian product of the respective means.

The product lattices that we have considered in this work appear in the image processing setting. The main application that we intend to develop of this work is aggregation of colors in this particular setting of image processing. We hope that our theoretical background applied to image processing, specially when considering median instead of means, will provide some advantages. There are two reasons. First, the median on a product lattice can be conveniently computed as the Cartesian product of the medians, whereas for the mean a more complex algorithm is required. Secondly, the median is much less sensitive to noisy values. In image processing non-Gaussian noise (e.g., salt and pepper noise) is quite common, and the use of the median allows one to filter out such noise efficiently.

An interesting line of research is to investigate the relation to aggregation of linguistic labels, in particular linguistic OWA [8, 9].

Acknowledgments

This research was partially supported by Grant TIN2007-65981.

References

- [1] G. Beliakov, A. Pradera, and T. Calvo. *Aggregation Functions: A Guide for Practitioners*. Springer, Heidelberg, Berlin, New York, 2007.
- [2] T. Calvo and G. Beliakov. Aggregation functions based on penalties. *Fuzzy Sets and Systems*, in press doi:10.1016/j.fss.2009.05.012, 2009.
- [3] T. Calvo, G. Mayor, and R. Mesiar, editors. *Aggregation Operators. New Trends and Applications*. Physica-Verlag, Heidelberg, New York, 2002.
- [4] T. Calvo, R. Mesiar, and R. Yager. Quantitative weights and aggregation. *IEEE Trans. on Fuzzy Systems*, 12:62–69, 2004.
- [5] B. De Baets and R. Mesiar. Triangular norms on product lattices. *Fuzzy Sets and Systems*, 104:61–75, 1999.
- [6] C. Gini. *Le Medie*. Unione Tipografico-Editoriale Torinese, Milan (Russian translation, Srednie Velichiny, Statistika, Moscow, 1970), 1958.
- [7] M. Grabisch, J.-L. Marichal, R. Mesiar, and E. Pap. *Aggregation Functions*. Cambridge University press, Cambridge, 2009.
- [8] F. Herrera and E. Herrera-Viedma. Aggregation operators for linguistic weighted information. *IEEE Trans. on Syst., Man, and Cybernetics*, 27:646–656, 1997.

- [9] F. Herrera and E. Herrera-Viedma. Linguistic decision analysis: Steps for solving decision problems under linguistic information. *Fuzzy Sets and Systems*, 115:67–82, 2000.
- [10] S. Jenei and B. De Baets. On the direct decomposability of t-norms on product lattices. *Fuzzy Sets and Systems*, 139:699–707, 2003.
- [11] G. Mayor and J. Monreal. Additive generators of discrete conjunctive aggregation operations. *IEEE Trans. on Fuzzy Systems*, 15:1046–1052, 2007.
- [12] R. Mesiar. Fuzzy set approach to the utility, preference relations, and aggregation operators. *Europ. J. Oper. Res.*, 176:414–422, 2007.
- [13] Y. Torra, V. Narukawa. *Modeling Decisions. Information Fusion and Aggregation Operators*. Springer, Berlin, Heidelberg, 2007.
- [14] R. Yager and A. Rybalov. Understanding the median as a fusion operator. *Int. J. General Syst.*, 26:239–263, 1997.

SPECIFICITY FOR INTERVAL-VALUED FUZZY SETS USING AGGREGATION OPERATORS

Ramón González-del-Campo¹ Luis Garmendia¹ Victoria López¹

Universidad Complutense de Madrid¹ rgonzale@estad.ucm.es, lgarmend@fdi.ucm.es, vlopez@fdi.ucm.es

Abstract

This paper proposes a new approach of measure of specificity for interval-valued fuzzy sets. It is showed a general expression of specificity which generalize the linear measure of specificity of Yager. Some examples are proposed.

Keywords: Specificity, Interval-valued fuzzy set.

1 Introduction

Interval-valued fuzzy sets ($\mathcal{IVFS}$) were introduced in the 60s by Grattan-Guinness [7], Jahn [8], Sambuc [9] and Zadeh [12]. They are extensions of classical fuzzy sets where the membership degree of the elements on the universe of discourse (between 0 and 1) is replaced by an interval in $[0, 1] \times [0, 1]$. They easily allow to model uncertainty and vagueness generalizing the fuzzy sets. Sometimes it is easier for experts to give a "membership interval" than a membership degree to objects on a universe. $\mathcal{IVFS}$ are a special case of type-2 fuzzy sets that simplifies the calculations while preserving their richness as well.

Fuzzy sets are a good tool to model imprecision due to vagueness. $\mathcal{IVFS}$ can also measure the imprecision due to uncertainty.

The concept of specificity provides a measure of the amount of information contained in a fuzzy set. It is strongly related to the inverse of the cardinality of a set. Specificity measures were introduced by Yager [10, 11] showing its usefulness as a measure of tranquillity when making a decision. The output information of expert systems and other knowledgebased system should be both specific and correct to be useful.

Specificity for fuzzy sets has been widely analyzed in [3, 4, 5].

2 Preliminaries

Let $X = \{a_1, \dots, a_n\}$ be a finite set.

Definition 2.1 A fuzzy set μ on X is normal if there exists an element $x \in X$ such that $\mu(x) = 1$

Definition 2.2 [11] Let $[0, 1]^X$ be the set of fuzzy sets on X . Let a_j be the j^{th} greatest membership degree of μ . A measure of specificity is a function $Sp: [0, 1]^X \rightarrow [0, 1]$ such that:

- $Sp(\mu) = 1$ if and only if μ is a singleton.
- $Sp(\emptyset) = 0$
- $-\frac{\partial Sp(\mu)}{\partial a_1} > 0$
 $-\frac{\partial Sp(\mu)}{\partial a_j} \leq 0$ for all $j \geq 2$

Definition 2.3 [11] Let $[0, 1]^X$ be the set of fuzzy sets on X . A weak measure of specificity is a function $Sp: [0, 1]^X \rightarrow [0, 1]$ such that:

- $Sp(\mu) = 1$ if and only if μ is a singleton.
- $Sp(\emptyset) = 0$
- If μ and η are normal fuzzy sets in X and $\mu \subset \eta$, then $Sp(\mu) \geq Sp(\eta)$.

Definition 2.4 Let Sp and Sp' be two measures of specificity. Sp is more strict than Sp' , denoted by $Sp \leq Sp'$, if for all set, μ , it verifies: $Sp(\mu) \leq Sp'(\mu)$.

Yager introduced [11] the linear measure of specificity on a finite space X as:

$$Sp_{\overline{w}}(\mu) = y_1 - \sum_{j=2}^n w_j y_j$$

where y_j is the j^{th} greatest membership degree of μ and $\{w_j\}$ is a set of weights verifying:

- $w_j \in [0, 1]$
- $\sum_{j=2}^n w_j = 1$
- $\{w_j\}$ is not increasing.

Definition 2.5 [2] Let $\mathcal{L} = (L, \leq_L)$ be a lattice that satisfies:

1. $L = \{[x_1, x_2] \in [0, 1]^2 \text{ with } x_1 \leq x_2\}$.
2. $[x_1, x_2] \leq_L [y_1, y_2]$ if and only if $x_1 \leq y_1$ and $x_2 \leq y_2$

Also by definition:

$$\begin{aligned} [x_1, x_2] <_L [y_1, y_2] &\Leftrightarrow x_1 < y_1, x_2 \leq y_2 \text{ or } \\ &x_1 \leq y_1, x_2 < y_2 \\ [x_1, x_2] =_L [y_1, y_2] &\Leftrightarrow x_1 = y_1, x_2 = y_2. \end{aligned}$$

$0_L =_L [0, 0]$ and $1_L =_L [1, 1]$ are the smallest and the greatest elements in L respectively.

$\mathcal{L}$ is a complete lattice and the supremum and infimum are defined as follows.

Definition 2.6 [1] Let $\{[v_i, w_i]\}$ be a set of intervals on L . Then

1. Infimum: $\text{Meet}\{[v_i, w_i]\} \equiv [\inf\{v_i\}, \inf\{w_i\}]$
2. Supremum: $\text{Joint}\{[v_i, w_i]\} \equiv [\sup\{v_i\}, \sup\{w_i\}]$

Definition 2.7 [2] An interval-valued fuzzy set A on a universe X can be represented by the mapping:

$$A = \{(a, [x_1, x_2]) \mid a \in X, [x_1, x_2] \in L\}$$

Definition 2.8 [2] Let X be a universe and A and B two interval-valued fuzzy sets. The equality between A and B is defined as: $A =_L B$ if and only if $A(a) =_L B(a) \forall a \in X$.

Definition 2.9 [2] Let X be a universe and A and B two interval-valued fuzzy sets. The inclusion of A in to B is defined as: $A \leq_L B$ if and only if $A(a) \leq_L B(a) \forall a \in X$.

Definition 2.10 [2] A negation function for interval-valued fuzzy sets $\mathcal{N}$ is a decreasing function, $N : L \rightarrow L$, that satisfies:

1. $N(0_L) =_L 1_L$

2. $N(1_L) =_L 0_L$

If $N(N([x_1, x_2])) =_L [x_1, x_2]$ then N is called an involutive negation.

Definition 2.11 A strong negation function for interval-valued fuzzy sets, N , is a strictly decreasing and involutive function, $N : L \rightarrow L$, that satisfies:

1. $N(0_L) =_L 1_L$
2. $N(1_L) =_L 0_L$

Example 2.1 Let N be the involutive mapping defined by :

$$\begin{aligned} N : L &\rightarrow L \\ N([x_1, x_2]) &=_L [1 - x_2, 1 - x_1] \end{aligned}$$

Then N is a negation operator for interval-valued fuzzy sets. It is trivial to prove that: $N(0_L) =_L 1_L$, $N(1_L) =_L 0_L$ and $N(N([x_1, x_2])) =_L [x_1, x_2]$.

3 Specificity for Interval-valued Fuzzy Sets

Definition 3.1 An operator $f : [0, 1]^2 \rightarrow [0, 1]$ with $x \leq y$ is called transformation operator if it is continuous, increasing and verifies:

1. $f(1, 1) = 1$
2. $f(0, 0) = 0$
3. $f(0, x) > 0$ for all $x \in (0, 1]$
4. $f(x, 1) < 1$ for all $x \in [0, 1)$

Definition 3.2 Let μ be an interval-valued fuzzy set on X and let $\mu(a_q) = [x_{1_q}, x_{2_q}]$ be its membership intervals. Let f be a transformation operator. Then, the f -list of μ is the set of all the membership intervals of elements of X , ordered through the operator f , that is, $[x, y] \leq_f [z, t]$ if and only if $f(x, y) \leq f(z, t)$:

Definition 3.3 An interval-valued fuzzy set μ on X is a singleton if there exists an element $a_i \in X$ such that $\mu(a_i) = 1_L$ and $\mu(a_j) = 0_L$ (for all $j \neq i$) for the others.

Definition 3.4 Let $([0, 1]^2)^X$ be the set of interval-valued fuzzy sets on X . Let f be a transformation operator. Let $\{[x_{1_q}, x_{2_q}]\}$ for all $q = 1..n$ be the f -list of μ . A f -measure of specificity for interval-valued fuzzy sets is a function $Sp_f : ([0, 1]^2)^X \rightarrow [0, 1]$ such that:

- $Sp_f(\mu) = 1$ if and only if μ is a singleton.
- $Sp_f(\emptyset) = 0$.
- If $[x_{1_1}, x_{2_1}]$ increases (according to $\leq_L$) then $Sp_f(\mu)$ increases.
- If $[x_{1_q}, x_{2_q}]$ increases (according to $\leq_L$) then $Sp_f(\mu)$ decreases for all $q : 2..n$.

Definition 3.5 An interval-valued fuzzy set μ on X is normal if there exists an element $a \in X$ such that $\mu(a) = 1_L$

Definition 3.6 [6] Let $([0, 1]^2)^X$ be the set of interval-valued fuzzy sets on X . A weak measure of specificity for interval-valued fuzzy sets is a function $Sp:([0, 1]^2)^X \rightarrow [0, 1]$ such that:

- $Sp(\mu) = 1$ if and only if μ is a singleton.
- $Sp(\emptyset) = 0$
- If μ and η are normal fuzzy sets in X and $\mu \subseteq_L \eta$, then $Sp(\mu) \geq Sp(\eta)$.

Lemma 3.1 If Sp_f is a f -measure of specificity for interval-valued fuzzy sets then Sp_f is a weak measure of specificity for interval-valued fuzzy sets.

Proof

Let $\{[x_{1_q}, x_{2_q}]\}$ and $\{[y_{1_q}, y_{2_q}]\}$ for all $q = 1..n$ be the f -list of μ and η respectively. If μ and η are normal and $\mu \subseteq_L \eta$ then $[x_{1_q}, x_{2_q}] \leq_L [y_{1_q}, y_{2_q}]$ for all $q = 2..n$. According to the fourth axiom of the definition 3.4 $Sp_f(\mu) \geq Sp_f(\eta)$.

Theorem 3.1 Let f be a transformation operator and $\{\alpha_i\}$ a set of weights that satisfies:

- $\alpha_j \in [0, 1]$
- $\sum_{j=2}^n \alpha_j = 1$
- $\{\alpha_j\}$ is not increasing.

Let T, T', S and N be, two t -norms, a t -conorm and a negation (in $[0, 1], \leq$) respectively. Let $[x_{1_q}, x_{2_q}]$ be the f -list of an interval-valued fuzzy set μ . Then, the next expression is a f -measure of specificity for interval-valued fuzzy sets:

$$Sp_f(\mu) = T(f(x_{1_1}, x_{2_1}), N(S(T'(\alpha_2, f(x_{1_2}, x_{2_2}))), \dots, T(\alpha_n, f(x_{1_n}, x_{2_n}))))))$$

This expression is a generalization of the t -norm based measure of specificity given in [3] but extended for $\mathcal{IVFS}$.

Proof

1. $Sp_f(\mu) = 1$ if and only if μ is a singleton:
 - If μ is a singleton then $[x_{1_1}, x_{2_1}] = [1, 1]$ and $[x_{1_k}, x_{2_k}] = [0, 0]$ for all $k > 1$. Then $f(x_{1_1}, x_{2_1})_1 = 1$ and $f(x_{1_1}, x_{2_1})_k = 0$ for all $k > 1$.
 - If $Sp_f(\mu) = 1$, it is necessary that $f(x_{1_1}, x_{2_1})_1 = 1$ and $S(T(\alpha_2, f(x_{1_1}, x_{2_1})_2), \dots, T(\alpha_n, f(x_{1_1}, x_{2_1})_n)) = 0$. Then $T(\alpha_k, f(x_{1_1}, x_{2_1})_k) = 0$ for all k and $f(x_{1_1}, x_{2_1})_k = 0$ for all k .
2. $Sp_f(\emptyset) = 0$: trivial.
3. Trivial due to the fact T, T' and S are monotonic.

Example 3.1 With $T(a, b) = \text{Max}\{0, a + b - 1\}$, $N(a) = 1 - a$, $S(a, b) = \text{Min}\{1, a + b\}$, $T'(a, b) = a * b$ and $f(x, y) = \frac{x+y}{2}$, it is obtained:

$$Sp(\mu) = \frac{1}{2}(x_{1_1} + x_{2_1}) - \sum_{j=2}^n \alpha_j(x_{1_j} + x_{2_j})$$

Example 3.2 With $T(a, b) = \text{Max}\{0, a + b - 1\}$, $N(a) = 1 - a$, $S(a, b) = \text{Min}\{1, a + b\}$, $T'(a, b) = a * b$ and $f(x, y) = \alpha * x + \beta * y$ with $\alpha + \beta = 1$, it is obtained:

$$Sp(\mu) = \alpha * x_{1_1} + \beta * x_{2_1} - \sum_{j=2}^n \alpha_j(\alpha * x_{1_j} + \beta * x_{2_j})$$

Example 3.3 With $T(a, b) = \text{Max}\{0, a + b - 1\}$, $N(a) = 1 - a$, $S(a, b) = \text{Min}\{1, a + b\}$, $T'(a, b) = a * b$ and $f(x, y) = x * y$, it is obtained:

$$Sp(\mu) = x_{1_1} * x_{2_1} - \sum_{k>1} \alpha_k * x_{1_k} * x_{2_k}$$

Those examples 3.1, 3.2 and 3.3 are extensions of R. Yager's linear measure of specificity [11] for $\mathcal{IVFS}$.

4 Conclusions and future work

An approach of measures of specificity of interval-Valued fuzzy sets is given. It is based on the R. Yager's concept of measure of specificity as a measure of information that evaluates a degree of usefulness of information contained on a fuzzy set in order to make a decision.

A general expression of measures of specificity of interval-Valued fuzzy sets is given using t -norms, t -conorms, a negation and a new proposed transformation operator in its definition, and some examples of

this expression are provided using different connectives and different transformation operators.

Acknowledgements

This research is partially supported by the Spanish Ministry of Science and Technology, grant number TIN2009-07901, the Research Group CAM GR58/08 at Complutense University of Madrid.

References

- [1] C. Cornelis, G. Deschrijver, and E. Kerre. Advances and challenges in interval-valued fuzzy logic. *Fuzzy Sets and Systems*, 157(5):622–627, 2006.
- [2] C. Cornelis, G. Deschrijver, and E. Kerre. Implication in intuitionistic fuzzy and interval-valued fuzzy set theory: construction, classification, application. *Int. J. Approx. Reasoning*, 35(1):55–95, 2004.
- [3] L. Garmendia, R.R. Yager, E. Trillas and A. Salvador. On t-norms based specificity measures. *Fuzzy Sets and Systems*, 133(2):237–248, 2003.
- [4] L. Garmendia, R.R. Yager, E. Trillas and A. Salvador. General measures of specificity of fuzzy sets under t-indistinguishabilities. *IEEE Transactions on Fuzzy Systems*, 14(4):568–572, 2006.
- [5] L. Garmendia, R.R. Yager, E. Trillas and A. Salvador. A t-norm based specificity for fuzzy sets on compact domains. *International Journal of General Systems*, 35(6):687–698, 2006.
- [6] R. González del Campo and L. Garmendia. Specificity, uncertainty and entropy measures of interval-valued fuzzy sets. *Proceedings EURO-FUSE Workshop Preference Modelling and Decision Analysis*, pages 273–278, 2009.
- [7] I. Grattan-Guinness. Fuzzy membership mapped onto interval and many-valued quantities. *Math. Logik. Grundlehren Math*, 22:149–160, 1975.
- [8] K.U. Jahn. Intervall-wertige Mengen. *Math. Nach.*, 68:115–132, 1975.
- [9] E. Sanchez and R. Sambuc. Fuzzy relationships. phi-fuzzy functions. application to diagnostic aid in thyroid pathology. *Proceedings of an International Symposium on Medical Data Processing*, pages 513–524, 1976.
- [10] R.R. Yager. Measuring tranquility and anxiety in decision-making - an application of fuzzy-sets. *International Journal of General Systems*, 8:139–146, 1982.
- [11] R.R. Yager. Ordinal measures of specificity. *International Journal of General Systems*, 17:57–72, 1990.
- [12] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning i. *Information Sciences*, 8:199–249, 1975.

Clustering con conjuntos intervalo-valorados difusos para la segmentación de imágenes de ultrasonidos

Aranzazu Jurio Miguel Pagola Edurne Barrenechea Carlos Lopez-Molina

Departamento de Automática y Computación, Universidad Pública de Navarra. Pamplona (España)
{aranzazu.jurio,miguel.pagola,edurne.barrenechea,carlos.lopez}@unavarra.es

Resumen

En este trabajo utilizamos los conjuntos intervalo-valorados difusos en el algoritmo Fuzzy Cluster Means para segmentar imágenes. Presentamos la definición de Función de Equivalencia Restringida Intervalo-Valorada y dos métodos para su construcción. También demostramos experimentalmente que al utilizar estos nuevos conceptos, el algoritmo obtiene mejores resultados en la segmentación de imágenes de ultrasonidos que cuando no se utilizan intervalos.

Palabras clave: Segmentación, Fuzzy Cluster Means, Conjuntos Intervalo-valorados difusos, Funciones de Equivalencia Restringida, Imágenes de ultrasonidos.

1 Introducción

Los quistes en el pecho, pequeñas acumulaciones de fluido, son la causa más importante de los tumores benignos (no cancerígenos). Durante los exámenes mamográficos, se utilizan imágenes de ultrasonidos para evaluar las anomalías presentes en las mamas. De cada anomalía, un radiólogo experto estudia su tamaño, área y otras características morfológicas [15] para determinar si es un quiste o no. Por tanto, el desarrollo de métodos que separen automáticamente la zona perteneciente al quiste del resto de la imagen, es muy importante para un correcto y rápido diagnóstico.

Uno de los factores que más influyen en la segmentación de imágenes de ultrasonidos es su mala calidad. El ruido, las sombras y la atenuación hacen que la segmentación sea una tarea muy complicada [12]. Se han propuesto varios métodos para segmentar imágenes de ultrasonidos, de los que una revisión muy completa puede encontrarse en [11].

Algunos de estos métodos se basan en la umbralización [6, 7] de la imagen, con el inconveniente de que en ellos sólo tiene en cuenta la intensidad de los píxeles. Para solventar este problema, en [13] se propone un sistema difuso intervalo-valorado en el que para cada píxel se trabaja con más de un atributo.

Un método muy utilizado para la segmentación de imágenes es el clustering. El propósito de este trabajo es adaptar el conocido algoritmo Fuzzy Cluster Means (FCM) [1, 2]. Dicho algoritmo trata de clasificar un conjunto de objetos, en este caso los píxeles de la imagen, en un determinado número de clusters. Dicho número debe ser especificado a priori. La idea principal es hallar tantos centros como clusters se quieren calcular, intentando minimizar la distancia entre cada píxel y los centros de los clusters.

La clasificación de algunos píxeles en un cluster o en otro es muy complicada, incluso para un experto. Esta duda se puede reflejar mejor si utilizamos intervalos de pertenencia en lugar de pertenencias únicas. Por ello, vamos a trabajar con conjuntos intervalo-valorados difusos [16].

Como ya hemos dicho, el algoritmo que proponemos se basa en el FCM. Las aportaciones que proponemos a este conocido algoritmo son dos: el uso de conjuntos intervalo-valorados difusos y la búsqueda de la máxima equivalencia intervalar entre los píxeles de la imagen y los centros de los clusters que se obtienen. Como representamos la incertidumbre mediante intervalos, para encontrar la máxima equivalencia nos vemos obligados a definir las Funciones de Equivalencia Restringida Intervalo-Valoradas.

Este trabajo se organiza de la siguiente forma: primero explicamos algunos conceptos preliminares. En la sección 3 definimos la nueva función de Equivalencia Restringida Intervalo-Valorada. En la sección 4, explicamos la aplicación de estas funciones a la segmentación de imágenes de ultrasonidos. Finalmente, mostramos algunos resultados experimentales y conclusiones.

2 Preliminares

Sabemos que en la teoría de conjuntos difusos una función $n : [0, 1] \rightarrow [0, 1]$ con $n(0) = 1$, $n(1) = 0$ que es estrictamente decreciente y continua se llama negación estricta. Si, además, n es involutiva, entonces se dice que n es una negación fuerte.

Denotamos por $L([0, 1])$ el conjunto de todos los subintervalos cerrados del intervalo unidad; esto es,

$$L([0, 1]) = \{\mathbf{x} = [\underline{x}, \bar{x}] \mid (\underline{x}, \bar{x}) \in [0, 1]^2 \text{ y } \underline{x} \leq \bar{x}\}.$$

$L([0, 1])$ es un conjunto parcialmente ordenado con respecto a la relación $\leq_L$ que se define de la siguiente manera: dado $\mathbf{x}, \mathbf{y} \in L([0, 1])$,

$$\mathbf{x} \leq_L \mathbf{y} \text{ si y sólo si } \underline{x} \leq \underline{y} \text{ y } \bar{x} \leq \bar{y};$$

$$\mathbf{x} =_L \mathbf{y} \text{ si y sólo si } \underline{x} = \underline{y} \text{ y } \bar{x} = \bar{y};$$

La relación superior es transitiva y antisimétrica y expresa el hecho de que $\mathbf{x}$ está fuertemente relacionado con $\mathbf{y}$, por lo que $(L([0, 1]), \leq_L)$ es un retículo completo, donde el elemento más pequeño es $0_L = [0, 0]$, y el mayor $1_L = [1, 1]$.

Definición 1 Un conjunto intervalo-valorado difuso A en el universo $U \neq \emptyset$ es un mapeo $A : U \rightarrow L([0, 1])$.

Definición 2 Una negación IV es una función $N : L([0, 1]) \rightarrow L([0, 1])$ que es decreciente (con respecto a $\leq_L$) tal que $N(1_L) = 0_L$ y $N(0_L) = 1_L$. Si para todo $\mathbf{x} \in L([0, 1])$, $N(N(\mathbf{x})) = \mathbf{x}$, se dice que N es involutiva.

Teorema 1 [9, 3] Una función $N : L([0, 1]) \rightarrow L([0, 1])$ es una negación IV involutiva si y sólo si existe una negación involutiva n tal que

$$N(\mathbf{x}) = [n(\bar{x}), n(\underline{x})].$$

Definición 3 [4] Una función $REF : [0, 1]^2 \rightarrow [0, 1]$ se dice función de equivalencia restringida si satisface las siguientes condiciones:

- (1) $REF(x, y) = REF(y, x)$ para todo $x, y \in [0, 1]$;
- (2) $REF(x, y) = 1$ si y sólo si $x = y$;
- (3) $REF(x, y) = 0$ si y sólo si $x = 1, y = 0$ o $x = 0, y = 1$;
- (4) $REF(x, y) = REF(n(x), n(y))$ para todo $x, y \in [0, 1]$, siendo n una negación fuerte;
- (5) Para todo $x, y, z \in [0, 1]$, si $x \leq y \leq z$, entonces $REF(x, y) \geq REF(x, z)$ and $REF(y, z) \geq REF(x, z)$.

Definición 4 [10] Una función $F : L([0, 1])^2 \rightarrow L([0, 1])$ se dice descomponible si existen dos funciones $F_1, F_2 : [0, 1]^2 \rightarrow [0, 1]$ tal que para todo $[\underline{x}, \bar{x}], [\underline{y}, \bar{y}] \in L([0, 1])$ satisfacen que

$$F([\underline{x}, \bar{x}], [\underline{y}, \bar{y}]) = [F_1(\underline{x}, \underline{y}), F_2(\bar{x}, \bar{y})]$$

Definición 5 [8] Una función binaria de agregación se define como una función $Ag : [0, 1]^2 \rightarrow [0, 1]$ tal que:

- (1) $Ag(x_1, x_2) \leq Ag(y_1, y_2)$ siempre que $x_1 \leq y_1$ y $x_2 \leq y_2$;
- (2) $Ag(0, 0) = 0$ y $Ag(1, 1) = 1$.

3 Funciones de Equivalencia Restringida Intervalo-valoradas

Definición 6 Una Función de Equivalencia Restringida Intervalo-valorada (REF_{IV}) es una función

$$REF_{IV} : L([0, 1]) \times L([0, 1]) \rightarrow L([0, 1])$$

tal que:

- (1) $REF_{IV}(\mathbf{x}, \mathbf{y}) = REF_{IV}(\mathbf{y}, \mathbf{x})$ para todo $\mathbf{x}, \mathbf{y} \in L([0, 1])$;
- (2) $REF_{IV}(\mathbf{x}, \mathbf{y}) = 1_L$ si y sólo si $\mathbf{x} = \mathbf{y}$;
- (3) $REF_{IV}(\mathbf{x}, \mathbf{y}) = 0_L$ si y sólo si $\mathbf{x} = 1_L, \mathbf{y} = 0_L$ or $\mathbf{x} = 0_L, \mathbf{y} = 1_L$;
- (4) $REF_{IV}(\mathbf{x}, \mathbf{y}) = REF_{IV}(N(\mathbf{x}), N(\mathbf{y}))$ siendo N una negación involutiva;
- (5) Para todo $\mathbf{x}, \mathbf{y}, \mathbf{z} \in L([0, 1])$, si $\mathbf{x} \leq_L \mathbf{y} \leq_L \mathbf{z}$, entonces $REF_{IV}(\mathbf{x}, \mathbf{y}) \geq_L REF_{IV}(\mathbf{x}, \mathbf{z})$ y $REF_{IV}(\mathbf{y}, \mathbf{z}) \geq_L REF_{IV}(\mathbf{x}, \mathbf{z})$.

Teorema 2 No existe ninguna REF_{IV} descomponible.

Demostración Suponemos que existe una REF_{IV} descomponible, es decir, existen dos funciones $F_1, F_2 : [0, 1] \times [0, 1] \rightarrow [0, 1]$ tal que:

$$REF_{IV}([\underline{x}, \bar{x}], [\underline{y}, \bar{y}]) = [F_1(\underline{x}, \underline{y}), F_2(\bar{x}, \bar{y})]$$

de tal forma que REF_{IV} cumple las cinco propiedades exigidas en la definición 6. Es decir:

- (1) Como $REF_{IV}(\mathbf{x}, \mathbf{y}) = REF_{IV}(\mathbf{y}, \mathbf{x})$, entonces F_1 y F_2 deben satisfacer que $F_1(a, b) = F_1(b, a)$ y $F_2(a, b) = F_2(b, a)$ para todo $a, b \in [0, 1]$.
- (2) Como $REF_{IV}([\underline{x}, \bar{x}], [\underline{y}, \bar{y}]) = 1_L$ si y sólo si $\underline{x} = \underline{y}$ y $\bar{x} = \bar{y}$, entonces $F_1(\underline{x}, \underline{x}) = 1$ y $F_2(\bar{x}, \bar{x}) = 1$, por lo que para todo $x \in [0, 1]$ $F_1(x, x) = F_2(x, x) = 1$.

- (3) Como $REF_{IV}(\underline{x}, \bar{x}, [\underline{y}, \bar{y}]) = 0_L$ si y sólo si $\mathbf{x} = 0$ y $\mathbf{y} = 1$ o viceversa, entonces $F_1(0, 1) = 0$ y $F_2(0, 1) = 0$.
- (4) Como $REF_{IV}([\underline{x}, \bar{x}], [\underline{y}, \bar{y}]) = REF_{IV}([n(\bar{x}), n(\underline{x})], [n(\bar{y}), n(\underline{y})])$, entonces $[F_1(\underline{x}, \underline{y}), F_2(\bar{x}, \bar{y})] = [F_1(n(\bar{x}), n(\bar{y})), F_2(n(\underline{x}), n(\underline{y}))]$. Separando cada término, $F_1(\underline{x}, \underline{y}) = F_1(n(\bar{x}), n(\bar{y}))$ y $F_2(\bar{x}, \bar{y}) = F_2(n(\underline{x}), n(\underline{y}))$. En particular, si $\underline{x} = 1$ y $\underline{y} = 0$, y $\bar{x} = \bar{y} = 1$ entonces $F_2(\bar{x}, \bar{y}) = F_2(1, 1) = 1$ y $F_2(n(\underline{x}), n(\underline{y})) = F_2(0, 1) = 0$. $1 \neq 0$, por lo que no existe ninguna REF_{IV} descomponible.

Teorema 3 Sea REF una función de equivalencia restringida, y Ag_1 y Ag_2 dos funciones de agregación simétricas tal que:

- (i) $Ag_1 \leq Ag_2$;
(ii) $Ag_1(a, b) = 1$ si y sólo si $a = b = 1$;
(iii) $Ag_2(a, b) = 0$ si y sólo si $a = b = 0$.

Bajo estas condiciones, la función

$$REF_{IV} : L([0, 1]) \times L([0, 1]) \rightarrow L([0, 1])$$

dada por

$$REF_{IV}(\mathbf{x}, \mathbf{y}) = [Ag_1(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y})), Ag_2(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y}))]$$

es una REF_{IV} en el sentido de la Definición 6.

Demostración Por hipótesis, Ag_1 es más pequeña que Ag_2 , por lo que REF_{IV} es un intervalo válido.

- (1) REF , Ag_1 y Ag_2 son funciones conmutativas, por lo que REF_{IV} es también una función conmutativa.
(2) (Suficiencia) Si

$$REF_{IV}(\mathbf{x}, \mathbf{y}) = 1_L,$$

entonces $Ag_1(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y})) = 1$, por lo que por (ii), $REF(\underline{x}, \underline{y}) = REF(\bar{x}, \bar{y}) = 1$. Una REF vale 1 si y sólo si ambos argumentos son iguales, por lo que $\underline{x} = \underline{y}$ y $\bar{x} = \bar{y}$, es decir $\mathbf{x} = \mathbf{y}$.

(Necesidad) Si $\mathbf{x} = \mathbf{y}$, es decir, $\underline{x} = \underline{y}$ y $\bar{x} = \bar{y}$, entonces

$$\begin{aligned} REF_{IV}(\mathbf{x}, \mathbf{y}) &= [Ag_1(REF(\underline{x}, \underline{x}), REF(\bar{x}, \bar{y})), \\ &\quad Ag_2(REF(\underline{x}, \underline{x}), REF(\bar{x}, \bar{y}))] = \\ &= [Ag_1(1, 1), Ag_2(1, 1)] \end{aligned}$$

por la segunda propiedad de la Definición 5 es lo mismo que $[1, 1]$.

- (3) (Suficiencia) Si $REF_{IV}(\mathbf{x}, \mathbf{y}) = 0_L$, entonces $Ag_2(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y})) = 0$, lo que significa por (iii) que $REF(\underline{x}, \underline{y}) = REF(\bar{x}, \bar{y}) = 0$. Una REF vale 0 si y sólo si uno de los argumentos vale 0 y el otro 1, por lo que $\underline{x} = 1$ e $\underline{y} = 0$ o viceversa, y $\bar{x} = 1$ y $\bar{y} = 0$ o viceversa, es decir $\underline{x} = \bar{x} = 1$ y $\underline{y} = \bar{y} = 0$ o vice versa.

(Necesidad) Si $\mathbf{x} = 1_L$ y $\mathbf{y} = 0_L$, entonces $REF_{IV}(\mathbf{x}, \mathbf{y}) = [Ag_1(REF(1, 0), REF(1, 0)), Ag_2(REF(1, 0), REF(1, 0))] = [Ag_1(0, 0), Ag_2(0, 0)]$, lo que es igual a $[0, 0]$ por la segunda propiedad de la Definición 5.

- (4) $REF_{IV}(N(\mathbf{x}), N(\mathbf{y})) = [Ag_1(REF(n(\bar{x}), n(\bar{y})), REF(n(\underline{x}), n(\underline{y}))), Ag_2(REF(n(\bar{x}), n(\bar{y})), REF(n(\underline{x}), n(\underline{y})))]$. Como $REF(\underline{x}, \underline{y}) = REF(n(\underline{x}), n(\underline{y}))$ y $REF(\bar{x}, \bar{y}) = REF(n(\bar{x}), n(\bar{y}))$, entonces $REF_{IV}(N(\mathbf{x}), N(\mathbf{y})) = [Ag_1(REF(\bar{x}, \bar{y}), REF(\underline{x}, \underline{y})), Ag_2(REF(\bar{x}, \bar{y}), REF(\underline{x}, \underline{y}))]$. Como Ag_1 y Ag_2 son funciones conmutativas, esta expresión es igual a $[Ag_1(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y})), Ag_2(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y}))] = REF_{IV}(\mathbf{x}, \mathbf{y})$.

- (5) Si $\mathbf{x} \leq_L \mathbf{y} \leq_L \mathbf{z}$, entonces $\underline{x} \leq \underline{y} \leq \underline{z}$ y $\bar{x} \leq \bar{y} \leq \bar{z}$. Por (5) de la Definición 3, $REF(\underline{x}, \underline{y}) \geq REF(\underline{x}, \underline{z})$ y $REF(\bar{x}, \bar{y}) \geq REF(\bar{x}, \bar{z})$. Por (1) de la Definición 5 $Ag_1(REF(\underline{x}, \underline{z}), REF(\bar{x}, \bar{z})) \leq Ag_1(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y}))$ y $Ag_2(REF(\underline{x}, \underline{z}), REF(\bar{x}, \bar{z})) \leq Ag_2(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y}))$, por lo que $REF_{IV}(\mathbf{x}, \mathbf{z}) \leq REF_{IV}(\mathbf{x}, \mathbf{y})$. El razonamiento para $REF_{IV}(\mathbf{x}, \mathbf{z}) \leq REF_{IV}(\mathbf{y}, \mathbf{z})$ se hace análogamente.

Corolario 1 Dada una función de equivalencia restringida REF , y dadas T y S una t-norma y una t-conorma,

$$REF_{IV}(\mathbf{x}, \mathbf{y}) = [T(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y})), S(REF(\underline{x}, \underline{y}), REF(\bar{x}, \bar{y}))]$$

es una REF_{IV} .

Demostración Este corolario es equivalente al Teorema 3. En este sentido, cualquier t-norma cumple las propiedades exigidas a Ag_1 y cualquier t-conorma cumple las propiedades exigidas a Ag_2 .

Es necesario que $Ag_1(a, b) = 1$ si y sólo si $a = b = 1$. (Suficiencia) Si la t-norma es el mínimo, si $\min(a, b) = 1$, entonces $a = b = 1$. Como el mínimo es la mayor de todas las t-normas, para que valga 1 los argumentos de cualquier t-norma deben ser mayores o iguales que 1. Como las t-normas trabajan sobre el intervalo unidad, es necesario que ambos argumentos sean 1.

(Necesidad) Sabemos que $T(1, x) = x$ para cualquier $x \in [0, 1]$. Si $x = y = 1$, entonces $T(1, 1) = 1$.

Es necesario que $Ag_2(a, b) = 0$ si y sólo si $a = b = 0$. (Suficiencia) Si la t-conorma es el máximo, si $max(a, b) = 0$, entonces $a = b = 0$. Como el máximo es la menor de todas las t-conormas, para que cualquier t-conorma valga 0 sus argumentos deben ser menores o iguales a 0. Como las t-conormas trabajan sobre el intervalo unidad, es necesario que ambos argumentos valgan 0.

(Necesidad) Sabemos que $S(x, 0) = x$ para cualquier $x \in [0, 1]$. Si $x = y = 0$, entonces $S(0, 0) = 0$.

Ejemplo 1 Si elegimos $T = \min$, $S = \max$ y $REF(x, y) = 1 - |x - y|$, tenemos que:

$$\begin{aligned} REF_{IV}(\underline{x}, \underline{y}) &= [\min(1 - |\underline{x} - \underline{y}|, 1 - |\bar{x} - \bar{y}|), \\ &\quad \max(1 - |\underline{x} - \underline{y}|, 1 - |\bar{x} - \bar{y}|)] = \\ [1 - \max(|\underline{x} - \underline{y}|, |\bar{x} - \bar{y}|), 1 - \min(|\underline{x} - \underline{y}|, |\bar{x} - \bar{y}|)] &= \\ N([\min(|\underline{x} - \underline{y}|, |\bar{x} - \bar{y}|), \max(|\underline{x} - \underline{y}|, |\bar{x} - \bar{y}|)]) &= \end{aligned}$$

4 Aplicación de REF_{IV} al algoritmo de clustering

El algoritmo propuesto en este artículo es una modificación del algoritmo Fuzzy Cluster Means (FCM) [1, 2]. Este último trata de encontrar el punto más característico de cada cluster, que pasa a ser considerado su centroide, y el grado de pertenencia de cada objeto a cada cluster. Esto se logra minimizando la función objetivo:

$$J_m(U, V) = \sum_{j=1}^n \sum_{i=1}^K w_{ij}^m \|x_j - C_i\|^2$$

En el nuevo algoritmo que proponemos, la imagen se segmenta en K áreas, representadas por K centroides intervalares, maximizando la equivalencia en cada área, usando las funciones de equivalencia restringida intervalo-valoradas que hemos definido previamente.

El algoritmo comienza con la inicialización de todos los parámetros requeridos. En primer lugar es necesario construir el conjunto intervalo-valorado difuso asociado a la imagen. Para ello, a cada píxel se le asigna un intervalo basándose en una malla centrada en sí mismo. El resto de parámetros que se deben inicializar son el número de áreas que se quieren segmentar, el grado de fuzzificación, la matriz de pesos, el número máximo de iteraciones y el umbral de finalización.

Una vez completada la inicialización, el algoritmo entra en un bucle. En él, se calcula el centroide de cada cluster y la matriz de pesos, hasta que la mejora sea no significativa. Para actualizar la matriz de pesos usamos las funciones de equivalencia restringidas intervalo-valoradas.

Cuando el bucle ha terminado, el algoritmo etiqueta cada píxel para mostrar los resultados. Para ello, cada píxel se asocia al cluster con el mayor peso en la matriz de pesos.

4.1 Estructura del algoritmo

1. Dada una imagen con n píxeles, construir el conjunto intervalo-valorado difuso asociado a ella $A = \{[\underline{x}_i, \bar{x}_i] | x_i \in X\}$ con $i = 1 \dots n$.
2. Inicializar
 - 2.1 K = número de áreas a segmentar.
 - 2.2 T = número máximo de iteraciones.
 - 2.3 m = grado de fuzzificación, $1 \leq m \leq \infty$.
 - 2.4 ε = umbral de finalización, $\varepsilon > 0$.
 - 2.5 W = matriz de pesos, $0 \leq w_{ij} \leq 1$ con $i = 1 \dots n$ y $j = 1 \dots K$.
 - 2.6 t = iteración actual = 0.
3. REPETIR

3.1 Incrementar la iteración actual

3.2 Calcular los centroides

$$C_j = \frac{\sum_{i=1}^n w_{ij}^m [\underline{x}_i, \bar{x}_i]}{\sum_{i=1}^n w_{ij}^m} \quad (1)$$

3.3 Actualizar la matriz de pesos $w_{ij} =$

$$\left(\sum_{k=1}^K \left(\frac{1 - (Ag(REF_{IV}(\underline{x}_i, C_j)))}{1 - (Ag(REF_{IV}(\underline{x}_i, C_k)))} \right)^{\frac{1}{m-1}} \right)^{-1}$$

where $\underline{x}_i = [\underline{x}_i, \bar{x}_i]$, $C_j = [C_j, \bar{C}_j]$, $C_k = [C_k, \bar{C}_k]$ and

$$Ag(REF_{IV}(\underline{x}, \underline{y})) = \frac{REF_{IV}(\underline{x}, \underline{y}) + \overline{REF_{IV}}(\underline{x}, \underline{y})}{2}.$$

3.4 Calcular el error.

HASTA $t = T$ o $error \leq \varepsilon$.

4. Etiquetar la imagen.

4.2 Preprocesamiento

Para probar la efectividad del nuevo algoritmo, vamos a segmentar algunas imágenes de ultrasonidos. En este caso trabajamos con imágenes de ultrasonidos de mamás, en las que el objetivo es clasificar cada píxel de la imagen. Es decir, decidir si pertenece a la zona que queremos segmentar (tumor) o al fondo.

Dadas las características de este tipo de imágenes, donde aparecen varias posibles zonas a analizar, es necesario elegir el píxel central de la zona deseada. Como el usuario debe contribuir con cierta información, decimos que el sistema propuesto es parcialmente supervisado.

Para mejorar los resultados preprocesamos las imágenes. Para ello, creamos dos nuevas imágenes partiendo de cada

una de las originales. La primera de ellas es una imagen de bordes difusos mientras que la segunda es una imagen realzada. Un ejemplo de estas nuevas imágenes se puede ver en la Fig. 1.

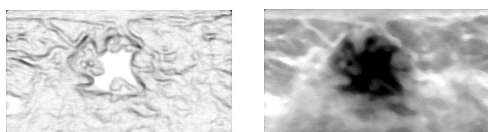


Figura 1: Imágenes de bordes difusos y realzada.

El algoritmo explicado anteriormente se va a ejecutar sobre estas dos imágenes, asociando a cada píxel un vector de dos intervalos. De esta forma, los centroides de cada cluster se representan por vectores de dos intervalos y cada píxel tiene un único valor de pertenencia a cada cluster.

4.2.1 Imagen realzada

Para crear la imagen realzada utilizamos el algoritmo de Sabha et. al. [14] para imágenes de ultrasonidos. La idea principal es borrar el ruido de la imagen y realzar los niveles de gris de la zona seleccionada. En primer lugar el ruido se borra utilizando el filtro de la mediana (7x7 o 9x9). Después, cada píxel se fuzzifica dependiendo de su nivel de gris usando una función de pertenencia que tiene en cuenta el nivel de gris de la vecindad del píxel.

4.2.2 Imagen de bordes difusos

La imagen de bordes difusos representa, para cada píxel, en qué medida ese píxel forma parte del borde. En [5] se presenta un método para obtener bordes difusos a partir de t-normas y t-conormas. Para cada píxel, el primer paso es construir una submatriz (3x3, 5x5, etc.). Aplicando una t-norma a los elementos de la submatriz se obtiene el extremo inferior del intervalo. Análogamente, el extremo superior se obtiene aplicando una t-conorma a la misma submatriz. La amplitud del intervalo, es decir, el extremo superior del intervalo menos el extremo inferior, se llama borde difuso. Si la t-norma y la t-conorma elegidas son el mínimo y el máximo respectivamente, el borde difuso consiste básicamente en la resta del máximo de los elementos de la submatriz menos el mínimo de los elementos.

4.3 Experimento

Las pruebas que mostramos a continuación se basan en las 8 imágenes de ultrasonidos que se muestran en la Fig. 2. Son imágenes de mamás en las que existen zonas de posibles tumores. A la hora de crear el intervalo asociado a cada imagen nos basamos en una malla 5x5 centrada en cada píxel. Para el extremo inferior tomamos el mínimo de dichos valores y para el extremo superior el máximo. Como expresión de REF_{IV} utilizamos la que se muestra en el ejemplo 1.

Para calcular la efectividad del método de segmentación que proponemos, usamos la medida de solapamiento de áreas entre la imagen calculada y la imagen objetivo. Esta última ha sido creada por un radiólogo experto.

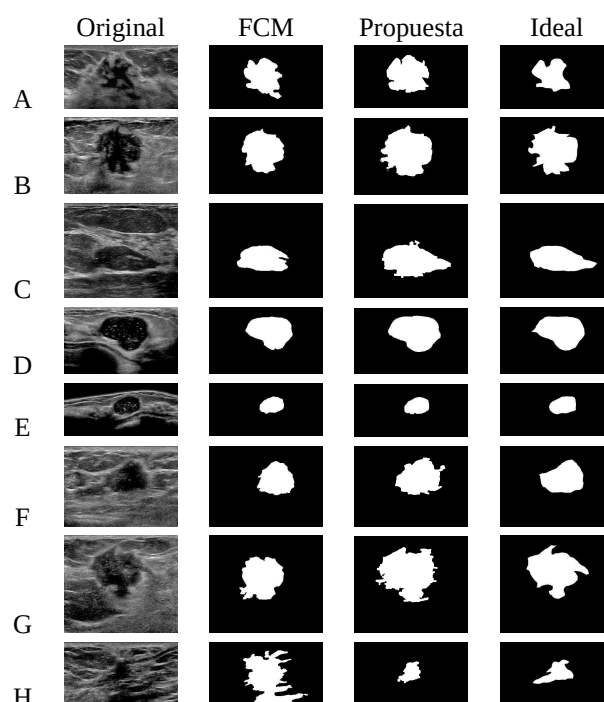


Figura 2: Resultados obtenidos.

Vamos a comparar los resultados obtenidos por nuestro algoritmo con los obtenidos mediante el algoritmo FCM, basándonos en su parecido con las imágenes objetivo. En la Fig. 2 se muestra una comparación gráfica de los resultados obtenidos. La primera columna muestra la imagen original, la segunda y la tercera se corresponden con las segmentaciones calculadas por el FCM y el algoritmo que proponemos respectivamente. Finalmente, la cuarta columna muestra la segmentación ideal.

Como se puede ver, los resultados obtenidos por el algoritmo propuesto se parecen más a la imagen ideal que los obtenidos por el FCM. Esta mejora se puede comprobar numéricamente con el porcentaje del solapamiento de áreas entre cada imagen y su ideal correspondiente (ver la tabla 1, donde A, B, ... H se refiere a cada una de las imágenes mostradas en la imagen 2).

Tabla 1: Soluciones analíticas.

	A	B	C	D
FCM	75.17	76.73	73.28	84.48
Propuesta	72.51	89.43	81.12	91.47
	E	F	G	H
FCM	74.11	72.65	57.87	27.55
Propuesta	76.77	81.72	77.54	53.49

Con estas pruebas, nuestro algoritmo propuesto mejora los resultados obtenidos por el FCM en un 10.2837%. Podemos decir que este es un buen resultado, ya que cualquier mejora en la segmentación correcta de este tipo de imágenes es un buen resultado.

El uso de los conjuntos intervalo-valorados difusos en este tipo de imágenes mejoran la precisión de la segmentación. En algunos casos, son capaces de descubrir zonas que a simple vista no son fácilmente reconocibles, como se puede ver en las imágenes mostradas.

5 Conclusiones

En este trabajo mostramos la ventaja de utilizar los conjuntos intervalo-valorados difusos en el algoritmo FCM. Para ello hemos introducido el concepto de función de equivalencia restringida intervalo-valorada. Hemos presentado un método para su construcción basado en funciones de agregación, y un caso particular de este método, basado en t-normas y t-conormas. En los resultados experimentales hemos demostrado la eficacia del método.

Como trabajo futuro, sería conveniente comparar los resultados obtenidos mediante el FCM con conjuntos intervalo-valorados difusos, con otros métodos de segmentación de imágenes de ultrasonidos, como los de umbralización [6, 7] o los sistemas difusos intervalo-valorados [13].

Agradecimientos

Este trabajo ha sido parcialmente subvencionado por el Ministerio de Educación y Ciencia con el proyecto TIN2007-65981.

Referencias

- [1] J.C. Bezdek. Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum, NY (1981).
- [2] J.C. Bezdek, J. Keller, R. Krisnapuram, N.R. Pal. Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. (Kluwer Academic Publishers) (1999).
- [3] H. Bustince, E.Barrenechea, M. Pagola. Generation of Interval-Valued Fuzzy and Atanassov's Intuitionistic Fuzzy Connectives from Fuzzy Connectives and from K_α Operators: Laws for Conjunctions and Disjunctions, Amplitude. *International Journal of Intelligent Systems*, vol. 23 (2008) 680-714.
- [4] H. Bustince, E. Barrenechea, M. Pagola. Restricted equivalence functions. *Fuzzy Sets and Systems* 157 (2006) 2333-2346.
- [5] H. Bustince, E. Barrenechea, M. Pagola, R. Orduna. Construction of interval type 2 fuzzy images to represent images in grayscale. False edges. *FUZZY IEEE* 2007. London, U.K.
- [6] H. Bustince, J. Sanz, J. Olagoitia. Umbralización de imágenes utilizando conjuntos intervalo-valorados difusos. Importancia del algoritmo de Tizhoosh. *Proceedings of ESTYLF'08*.
- [7] H. Bustince, et al. Ignorance functions. An application to the calculation of the threshold in prostate ultrasound images. *Fuzzy Sets and Systems*(2009),doi: 10.1016/j.fss.2009.03.005
- [8] T. Calvo, A. Kolesarova, M.Komornikova, R. Mesiar. Aggregation operators: properties, classes and construction methods. In T. Calvo, G. Mayor and R. Mesiar (eds): *Aggregation Operators New Trends and Applications* (Physica-Verlag, Heidelberg, 2002) 3-104.
- [9] G. Deschrijver, C. Cornelis, EE.Kerre. On the representation of intuitionistic fuzzy T-norms and T-conorms. *IEEE Trans Fuzzy Syst.* 12(1) (2004) 45-61.
- [10] P. Drygas. The problem of distributivity between binary operations in bi-fuzzy set theory. *Proceedings of IPMU'08*. L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (eds). (2008) 1648-1653.
- [11] J.A. Noble, D. BouKerroui. Ultrasound image segmentation: A survey. *IEEE Transactions on Medical Imaging* 25 (2006) 987-1010.
- [12] K. Ogawa, M. Fukushima, K. Kubota, N. Hisa. Computer-aided diagnosis system for diffuse liver-diseases with ultrasonography by neural networks. *IEEE Transactions on Nuclear Science*. 45(6) (1998) 3069-3074.
- [13] M. Pagola, E. Barrenechea, A. Jurio, M. Galar, P. Couto, F. Sahba, H. Tizhoosh. Interval-Valued Fuzzy System for Segmentation of Prostate Ultrasound Images. *Proceedings of IFSA/EUSFLAT 2009*.
- [14] F. Sahba, H.R. Tizhoosh, M.M. Salama. A coarse-to-fine approach to prostate boundary segmentation in ultrasound images. *BioMedical Engineering OnLine*. (2005) 4-58.
- [15] M.H. Yap, E.A. Edirisinghe, H.E. Bez. A novel algorithm for initial lesion detection in ultrasound breast images. *Journal of applied clinical medical physics* (Fall 2008). Vol. 9, number 4.
- [16] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning-I. *Information Science*. 8 (1975) 199-249.

Reducción de imagen utilizando operadores OWA de dimensión dos

Daniel Paternain¹ Humberto Bustince¹ Gleb Beliakov² Mikel Galar¹

¹ Departamento de Automática y Computación, Universidad Pública de Navarra, Pamplona, España
{daniel.paternain,humberto.bustince,mikel.galar}@unavarra.es

² School of Engineering and Information Technology, Deakin University, Burwood, Australia gleb@deakin.edu.au

Resumen

En este trabajo vamos a aplicar la relación entre el operador de Atanassov y los operadores OWA de dimensión dos en la reducción de imágenes. Veremos que, utilizando nuestro algoritmo de reducción y estos operadores, calculamos diferentes coeficientes que nos permiten obtener distintas reducciones de una imagen.

Palabras Clave: Reducción de imagen, Operadores OWA, Operador de Atanassov, Conjuntos intervalo-valorados difusos.

Además, veremos que bajo las propiedades de distintas familias de operadores OWA y su relación con el operador de Atanassov, obtendremos distintas reducciones de una misma imagen.

Este trabajo está organizado de la siguiente forma. Primero, comenzamos recordando los conceptos de conjunto intervalo-valorado difuso y de operador de Atanassov. En la Sección 3 mostramos la relación entre operadores OWA y operadores de Atanassov generalizados. En la Sección 4 presentamos varios métodos para el cálculo del coeficiente α basándonos en distintas familias de operadores OWA. En la Sección 5 proponemos un algoritmo de reducción de imágenes y en la Sección 6 mostramos los resultados experimentales. Finalmente terminamos con algunas conclusiones y líneas futuras.

1 Introducción

El objetivo de los algoritmos de reducción de imagen consisten en disminuir el tamaño o la resolución de la imagen con la menor pérdida de calidad posible ([12, 13, 20]). Para ello, el algoritmo deberá mantener la mayor cantidad de información relevante respecto a la imagen original.

En 1983, Atanassov presentó un operador ([1, 2]) que asocia un conjunto difuso a cada conjunto intuicionista de Atanassov o conjunto intervalo-valorado difuso ([11, 14, 16]). Posteriormente, en 1988 ([18]), Yager presentó los operadores *Ordered Weighted Averaging* (OWA). En este trabajo nos basamos en la idea de que bajo ciertas condiciones, el resultado numérico de ambos operadores coincide. Para ello definimos un operador que generaliza los operadores de Atanassov y que es un operador OWA de dimensión 2 ([3, 4, 5]).

En este trabajo vamos a utilizar ambos operadores en la reducción de imágenes. Dividimos la imagen en bloques. A cada bloque le asociamos un intervalo y de esta forma construimos un conjunto intervalo-valorado difuso asociado a la imagen. Veremos que mediante los operadores antes mencionados obtenemos un nuevo conjunto (difuso) que representa la imagen reducida.

2 Definiciones preliminares

Denotamos por $L([0, 1])$ el conjunto de todos los subintervalos cerrados en el intervalo cerrado $[0, 1]$

$$L([0, 1]) = \{x = [\underline{x}, \bar{x}] | (\underline{x}, \bar{x}) \in [0, 1]^2 \text{ y } \underline{x} \leq \bar{x}\}$$

$L([0, 1])$ es un conjunto parcialmente ordenado respecto a la relación de orden $\leq_L$ definida de la siguiente forma: dados $x, y \in L([0, 1])$

$$x \leq_L y \text{ si y sólo si } \underline{x} \leq \underline{y} \text{ y } \bar{x} \leq \bar{y}.$$

Con esta relación de orden, $(L([0, 1]), \leq_L)$ es un retículo completo, cuyo menor y mayor elemento son $0_L = [0, 0]$ y $1_L = [1, 1]$ respectivamente ([3, 4, 9, 10]).

Definición 1. Un conjunto intervalo-valorado difuso (IVFS) A definido sobre el universo $U \neq \emptyset$ es una aplicación $A : U \rightarrow L([0, 1])$, tal que el grado de pertenencia de $u \in U$ viene dado por $A(u) = [\underline{A}(u), \bar{A}(u)] \in L([0, 1])$, donde $\underline{A} : U \rightarrow [0, 1]$ y $\bar{A} : U \rightarrow [0, 1]$ son aplicaciones que definen el extremo inferior y superior del intervalo de pertenencia de $A(u)$ respectivamente.

En 1983, Atanassov propuso un operador para asociar un conjunto difuso a cada conjunto intervalo-valorado difuso

$([1, 2])$. Este operador, llamado K_α , toma un intervalo y asocia un punto interior del mismo a cada elemento del conjunto.

Definición 2. Sea $\alpha \in [0, 1]$. El operador de Atanassov K_α es una función $K_\alpha : L([0, 1]) \rightarrow [0, 1]$ definida por

1. $K_0(\mathbf{x}) = \underline{x}$ para todo $\mathbf{x} \in L([0, 1])$
2. $K_1(\mathbf{x}) = \bar{x}$ para todo $\mathbf{x} \in L([0, 1])$
3. $K_\alpha(\mathbf{x}) = K_\alpha([K_0(\mathbf{x}), K_1(\mathbf{x})]) = K_0(\mathbf{x}) + \alpha(K_1(\mathbf{x}) - K_0(\mathbf{x})) = \underline{x} + \alpha(\bar{x} - \underline{x})$ para todo $\mathbf{x} \in L([0, 1])$

En [4] se propone una generalización de este operador

Definición 3. Sea $\alpha \in [0, 1]$. Un operador D_α es una función $D_\alpha : L([0, 1]) \rightarrow [0, 1]$ tal que satisface las siguientes condiciones:

1. Si $\underline{x} = \bar{x}$, entonces $D_\alpha(\mathbf{x}) = \underline{x}$
2. $D_0(\mathbf{x}) = \underline{x}$, $D_1(\mathbf{x}) = \bar{x}$ para todo $\mathbf{x} \in L([0, 1])$
3. Si $\mathbf{x} \leq_L \mathbf{y}$ con $\mathbf{x}, \mathbf{y} \in L([0, 1])$, entonces $D_\alpha(\mathbf{x}) \leq D_\alpha(\mathbf{y})$
4. $D_\alpha([0, 1]) = \alpha$ para todo $\alpha \in [0, 1]$

Proposición 1. Para todo $\alpha \in [0, 1]$, el operador K_α es un operador D_α .

Nota: En este trabajo vamos a utilizar la siguiente expresión del operador D_α :

$$D_\alpha([x, \bar{x}]) = \underline{x} + \alpha(\bar{x} - \underline{x}) = K_\alpha([x, \bar{x}]) \quad (1)$$

3 Operador K_α y operadores OWA

En [18], Yaguer propuso los operadores OWA de la siguiente forma:

Definición 4. Una función $F : [0, 1]^n \rightarrow [0, 1]$ es un operador OWA de dimensión n si existe un vector de pesos $W, W = (w_1, w_2, \dots, w_n) \in [0, 1]^n$ con $\sum_i w_i = 1$ y tal que

$$F(a_1, a_2, \dots, a_n) = \sum_{j=1}^n w_j b_j$$

con b_j el j -ésimo mayor elemento de a_i .

En su definición original, los operadores OWA realizan un mapeo completo de $\mathbb{R}^n$ en $\mathbb{R}$. Sin embargo, en este trabajo vamos a reducir el dominio a $[0, 1]^2$ en $[0, 1]$.

Así, si nos centramos en operadores OWA de dimensión dos con vector de pesos $W = (\alpha, 1 - \alpha)$, podemos pensar en aplicarlos a los extremos de un intervalo. En este caso, el resultado numérico de la aplicación del operador OWA y del operador D_α sobre el intervalo coinciden. Sin embargo, los dos operadores son conceptualmente diferentes. Mientras el operador D_α actúa sobre elementos de $L([0, 1])$, el dominio de los operadores OWA es $[0, 1]^2$. De esta forma, los operadores OWA necesitan una operación

previa de orden para asegurar que los elementos sean extremos de un intervalo definido en $L([0, 1])$. Por esta razón damos el siguiente teorema que estudia la relación entre ambos operadores.

Definimos un nuevo operador $\mathbb{D}_\alpha$ que consiste en la composición del operador D_α con la función

$$\begin{aligned} i : [0, 1]^2 &\rightarrow L([0, 1]) \\ (x, y) &\rightarrow [\min(x, y), \max(x, y)] \end{aligned} \quad (2)$$

Teorema 1. 1. Sea $\alpha \in [0, 1]$ y $\mathbb{D} = D_\alpha \circ i$ donde D_α es el operador dado en la Definición 3. Entonces, si F es el operador OWA (de dimensión 2) definido por el vector de pesos $W = (\alpha, 1 - \alpha)$, tenemos que

$$\mathbb{D}_\alpha(x, y) = F(x, y) \text{ para todo } x, y \in [0, 1].$$

2. Sea F un operador OWA (de dimensión 2) con vector de pesos $W = (w_1, w_2)$. Entonces, para cada $(x, y) \in [0, 1]^2$ tenemos que

$$F(x, y) = \mathbb{D}_\alpha(x, y), \text{ con } \alpha = w_1.$$

4 Métodos de cálculo del coeficiente α

Antes de mostrar los distintos métodos de cálculo del coeficiente α , estudiamos dos medidas definidas en [19] asociadas a un operador OWA: el orness y la dispersión.

Definición 5. Sea F un operador OWA y W su vector de pesos. El orness viene definido por

$$\text{orness}(F) = \frac{1}{(n-1)} \sum_{i=1}^n (n-i)w_i \quad (3)$$

Proposición 2. $\text{orness}(\mathbb{D}_\alpha) = \alpha$

Definición 6. Sea F un operador OWA y W su vector de pesos. La dispersión viene definida por

$$\text{Disp}(F) = - \sum_{i=1}^n w_i \ln w_i \quad (4)$$

Proposición 3. $\text{Disp}(\mathbb{D}_\alpha) = \alpha \ln(\frac{1-\alpha}{\alpha}) + \ln(1-\alpha)$

A continuación mostramos diferentes métodos para obtener el vector de pesos de un operador $\mathbb{D}_\alpha$ en base a algunas familias de operadores OWA.

Definición 7. F es un operador ME-OWA si F es un operador OWA tal que dado un valor de orness β , F maximiza su dispersión. En concreto, resolvemos la siguiente ecuación:

$$\begin{aligned} \text{Max} \quad & \sum_{i=1}^n w_i \ln w_i \\ \text{con} \quad & \beta = 1/(n-1) \sum_{i=1}^n (n-i)w_i \\ \text{donde} \quad & \sum_{i=1}^n w_i = 1, w_i \in [0, 1] \end{aligned}$$

Proposición 4. El operador $\mathbb{D}_\alpha$ es un operador ME-OWA para todo $\alpha \in [0, 1]$ con $\text{orness}(\mathbb{D}_\alpha) = \alpha$.

Definición 8. F es un operador S-OWA generalizado si F es un operador OWA donde

$$\begin{aligned} w_1 &= a + \frac{1}{n}(1 - (a + b)) \\ w_i &= \frac{1}{n}(1 - (a + b)), \quad \text{con } i = 2, \dots, n-1, \\ w_n &= b + \frac{1}{n}(1 - (a + b)). \end{aligned}$$

siendo $a, b \in [0, 1]$ y $a + b \leq 1$.

Proposición 5. Sean $a, b \in [0, 1]$ tales que $a + b \leq 1$. Si tomamos

$$\alpha = \frac{1 + a - b}{2} \quad (5)$$

entonces $\mathbb{D}_\alpha$ es un operador S-OWA generalizado.

Definición 9. F es un operador BADD-OWA si F es un operador OWA donde

$$w_i = \frac{b_i^\beta}{\sum_{j=1}^n b_j^\beta}$$

con $\beta \geq 0$ y siendo b_i como en la Definición 4.

Proposición 6. Sea $\beta \geq 0$. Si tomamos

$$\alpha = \frac{(\mathbb{D}_1(x, y))^\beta}{(\mathbb{D}_0(x, y))^\beta + (\mathbb{D}_1(x, y))^\beta} \quad (6)$$

con $x, y \in [0, 1]$, entonces $\mathbb{D}_\alpha$ es un operador BADD-OWA.

Definición 10. Sea $\beta \geq 0$. F es un operador BADD-OWA modificado si F es un operador OWA donde

1. $w_i = \frac{(1/b_i)^\beta}{\sum_{j=1}^n (1/b_j)^\beta}$ o bien
2. $w_i = \frac{(1-b_i)^\beta}{\sum_{j=1}^n (1-b_j)^\beta}$ o bien
3. $w_i = \frac{1}{n-1} \left(1 - \frac{b_i^\beta}{\sum_{j=1}^n b_j^\beta}\right)$ o bien
4. $w_i = \frac{(b_{n-i+1})^\beta}{\sum_{j=1}^n b_j^\beta}$

siendo b_i como en la Definición 4.

Proposición 7. Sea $\beta \geq 0$. Los siguientes ítems se cumplen

1. Si tomamos

$$\begin{aligned} \alpha &= \frac{(1/\mathbb{D}_1(x, y))^\beta}{(1/\mathbb{D}_1(x, y))^\beta + (1/\mathbb{D}_0(x, y))^\beta} = \\ &= 1 - \frac{(\mathbb{D}_1(x, y))^\beta}{(\mathbb{D}_1(x, y))^\beta + (\mathbb{D}_0(x, y))^\beta} = \frac{(\mathbb{D}_0(x, y))^\beta}{(\mathbb{D}_1(x, y))^\beta + (\mathbb{D}_0(x, y))^\beta} \end{aligned} \quad (7)$$

entonces $\mathbb{D}_\alpha$ es un operador BADD-OWA modificado en el sentido de los ítems 1, 3 y 4 de la Definición 10.

2. Si tomamos

$$\alpha = \frac{(1 - \mathbb{D}_1(x, y))^\beta}{(1 - \mathbb{D}_1(x, y))^\beta + (1 - \mathbb{D}_0(x, y))^\beta} \quad (8)$$

entonces $\mathbb{D}_\alpha$ es un operador BADD-OWA modificado en el sentido del ítem 2 de la Definición 10.

5 Algoritmo de reducción de imagen

En este trabajo vamos a considerar una imagen Q de $N \times M$ píxeles como una relación difusa R de $N \times M$ elementos. Cada elemento de la relación tiene como valor de pertenencia la intensidad normalizada del píxel correspondiente a la misma posición. Normalizamos las intensidades dividiéndolas por $L-1$ (siendo L el número de niveles de gris empezando por 0). Así, el píxel localizado en la posición (i, j) tiene asociado un valor de pertenencia que denotamos por q_{ij} .

En nuestra propuesta de reducción de imágenes usamos los conjuntos intervalo-valorados difusos y el operador D_α . Como podemos ver en [6, 7, 8, 15, 17], la aplicación de conjuntos intervalo-valorados difusos en procesamiento de imagen permite el desarrollo de algoritmos con muy buenos resultados en distintas áreas, como la detección de bordes, el contraste o la umbralización. El algoritmo que proponemos es el siguiente:

1. Dividir la relación R en bloques de tamaño $n \times n$. Si M o N no son múltiplos de n , eliminaremos el mínimo número de filas y columnas en el borde de la matriz hasta obtener un nuevo tamaño que lo cumpla.
2. Asociar a cada bloque un intervalo de la siguiente forma: el extremo inferior viene dado por el mínimo de las intensidades del bloque y el extremo superior por el máximo de las intensidades (normalizadas).
3. Elegir el parámetro α del operador D_α .
4. Asociar a cada intervalo el número obtenido al aplicar el operador D_α .

Ejemplo: Sea R una relación de dimension 6×6 y sea $n = 3$

$$\begin{pmatrix} q_{1,1} & q_{1,2} & q_{1,3} & q_{1,4} & q_{1,5} & q_{1,6} \\ q_{2,1} & q_{2,2} & q_{2,3} & q_{2,4} & q_{2,5} & q_{2,6} \\ q_{3,1} & q_{3,2} & q_{3,3} & q_{3,4} & q_{3,5} & q_{3,6} \\ q_{4,1} & q_{4,2} & q_{4,3} & q_{4,4} & q_{4,5} & q_{4,6} \\ q_{5,1} & q_{5,2} & q_{5,3} & q_{5,4} & q_{5,5} & q_{5,6} \\ q_{6,1} & q_{6,2} & q_{6,3} & q_{6,4} & q_{6,5} & q_{6,6} \end{pmatrix}$$

Así, el conjunto intervalo-valorado difuso asociado a R está formado por 4 elementos:

$$\left(\left[\bigwedge_{\substack{i=1,2,3 \\ j=1,2,3}} q_{i,j} \quad \bigvee_{\substack{i=1,2,3 \\ j=1,2,3}} q_{i,j} \right] \quad \left[\bigwedge_{\substack{i=1,2,3 \\ j=4,5,6}} q_{i,j} \quad \bigvee_{\substack{i=1,2,3 \\ j=4,5,6}} q_{i,j} \right] \right)$$

Una vez obtenido el conjunto intervalo-valorado difuso para la imagen, construimos un conjunto difuso. Mediante el operador D_α aplicado a cada intervalo obtenemos una imagen reducida de tamaño 2×2 .

6 Resultados experimentales

En esta sección vamos a utilizar el algoritmo de reducción propuesto en la Sección 5 utilizando el operador D_α y las propiedades de las familias de operadores OWA de la Sección 4. En función del operador OWA elegido, obtenemos dos situaciones bien diferenciadas. En la primera, el valor de α es constante para toda la imagen (aplicamos el mismo valor a cada intervalo). Esto ocurre con los operadores ME-OWA y S-OWA generalizados debido a su propia definición. En la segunda, el valor de α depende de los extremos del intervalo, por lo que varía según el intervalo en el que trabajamos. Esto ocurre con los operadores BADD-OWA y BADD-OWA modificados.

Todas las pruebas de esta sección han sido realizadas con las imágenes Lena y Camera (Figura 1). El tamaño del bloque de reducción es $n = 3$ (submatrices de tamaño 3×3). De esta forma, el tamaño de la imagen reducida es 9 veces más pequeña que la original.



Figura 1: Imágenes Lena y Camera originales

6.1 Cálculo de α constante para toda la imagen

En esta sección el valor del parámetro α (constante) se va a calcular utilizando los operadores ME-OWA y S-OWA generalizados.

6.1.1 Reducción mediante operadores ME-OWA

Como hemos visto en la Sección 4, la construcción de un operador ME-OWA es directa. Por este motivo, vamos a analizar tres casos diferentes: $\alpha = 0$, $\alpha = 0.5$ y $\alpha = 1$.

Con $\alpha = 0$, asociamos el extremo inferior del intervalo a cada bloque. Con $\alpha = 0.5$, asociamos el punto medio del intervalo. Por último, con $\alpha = 1$, asociamos el extremo superior del intervalo.

Evidentemente, al aumentar el valor de α , aumenta el grado de pertenencia y por tanto la intensidad de la imagen. Así, la imagen es más oscura para $\alpha = 0$ que para $\alpha = 0.5$, que a su vez es más oscura que para $\alpha = 1$, tal como vemos en la Figura 2



Figura 2: Reducción mediante operadores ME-OWA

6.1.2 Reducción mediante operadores S-OWA generalizados

Para construir un operador S-OWA generalizado es necesario, como hemos visto en la Definición 8, la elección de dos parámetros $a, b \in [0, 1]$ tales que $a + b \leq 1$. Si estudiamos como contribuyen los parámetros en el operador, obtenemos los siguientes casos:

1. Si $a = b$, tenemos que $\alpha = 0.5$, obteniendo así la media de los extremos del intervalo.
2. Si $a = 1$ y $b = 0$, tenemos que $\alpha = 1$.
3. Si $a = 0$ y $b = 1$, tenemos que $\alpha = 0$.
4. Si $a + b < 1$ y $a \neq b$, cuando $a > b$, el valor de α tiende a 1, por lo que el resultado tiende hacia el extremo superior del intervalo. Si $a < b$, el valor de α es más cercano a 0, por lo que el resultado se aproxima al extremo inferior.

Los casos 1, 2 y 3 ya han sido analizados en la Subsección 6.1.2. Vamos a estudiar el caso 4. Para ello hemos considerado los siguientes casos: para $a = 0.5, b = 0.25$, al ser $a > b$, damos más importancia al extremo superior del intervalo, obteniendo así un valor de $\alpha = 0.625$. Al aumentar el valor de a , entonces $\alpha \rightarrow 1$. Por ejemplo, si $a = 0.9, b = 0$, entonces $\alpha = 0.95$. Con este valor obtenemos una imagen muy parecida a las obtenidas en la última columna de la Figura 2. Si elegimos $a = 0.25, b = 0.5$, al dominar el extremo inferior, obtenemos un valor de $\alpha = 0.375$. Al aumentar el valor de b , $\alpha \rightarrow 0$, como ocurre si tomamos $a = 0, b = 0.9$ (el valor en este caso es de $\alpha = 0.05$). Nótese en la Figura 3 que al aumentar el valor de α la imagen se aclara.



Figura 3: Reducción mediante operadores S-OWA

6.2 Cálculo de α variable

En esta sección el valor del parametro α (variable) se va a calcular utilizando los operadores BADD-OWA y BADD-OWA modificados.

6.2.1 Reducción mediante operadores BADD-OWA

Como hemos visto en la Definición 9, es necesaria la elección de un parametro $\beta \geq 0$ para la construcción de un operador BADD-OWA. Sabemos que si tomamos $\beta = 0$, entonces el valor de $\alpha = 0.5$, caso estudiado en la segunda columna de la Figura 2. En otro caso, si tomamos $\beta = 1$, el valor de α lo calculamos de la siguiente forma:

$$\alpha = \frac{\bar{x}}{\bar{x} + x} \quad (9)$$

siendo $[x, \bar{x}] \in L([0, 1])$ el intervalo que representa al bloque. De esta forma sabemos que $\alpha \geq 0.5$ y que al aumentar el valor de β , $\alpha \rightarrow 1$, es decir, el resultado tiende al extremo superior del intervalo. Así, con un valor de β alto, obtenemos una imagen muy similar a las de la columna 3 de la Figura 2, es decir, aclaramos la imagen (Figure 4).



Figura 4: Reducción mediante operadores BADD-OWA

6.3 Reducción mediante operadores BADD-OWA modificados

Si utilizamos operadores BADD-OWA modificados, podemos realizar dos calculos de α basándonos en los ítems 1 y 2 de la Proposición 7. Al igual que para los operadores BADD-OWA, el valor de α depende del intervalo en el que trabajamos y del parametro β . En ambos casos, ocurre que

si $\beta = 0$, el valor de $\alpha = 0.5$ (de forma similar a los operadores BADD-OWA).

Bajo las condiciones del ítem 1 de la Proposición 7, y tomando $\beta = 1$, el valor de α viene dado de la siguiente forma:

$$\alpha = \frac{x}{x + \bar{x}} \quad (10)$$

siendo $[x, \bar{x}] \in L([0, 1])$ el intervalo que representa al bloque. En este caso, $\alpha \leq 0.5$ y al aumentar β , $\alpha \rightarrow 0$ cuando $\beta \rightarrow \infty$ (Nótese en Figura 5 que la imagen va oscureciendo al aumentar β y disminuir α). Si nos basamos



Figura 5: Reducción mediante operadores BADD-OWA modificados segun ítem 1

en el ítem 2 de la Proposición 7, para un valor de $\beta = 1$, α viene dado por la siguiente expresion:

$$\alpha = \frac{1 - \bar{x}}{(1 - \bar{x}) + (1 - x)} \quad (11)$$

siendo $[x, \bar{x}] \in L([0, 1])$ el intervalo que representa al bloque. Al igual que ocurre con el cálculo basado en el ítem 1, el valor de α disminuye al aumentar β (Nótese en Figura 6 que, de la misma forma que con el ítem 1, la imagen se oscurece al aumentar β y disminuir α).



Figura 6: Reducción mediante operadores BADD-OWA modificados segun ítem 2

7 Conclusiones y líneas futuras

En este trabajo hemos demostrado que podemos obtener operadores OWA de dimensión 2 a partir de operadores de Atanassov y aplicarlos a la reducción de imágenes. Además, hemos utilizado diferentes métodos de la literatura en el cálculo de los pesos de operadores OWA. Este

hecho nos ha permitido obtener distintas familias de imágenes reducidas a partir de una misma imagen.

En un futuro queremos analizar las imágenes obtenidas más profundamente. En primer lugar, queremos estudiar no sólo como afectan los operadores utilizados a la intensidad de la imagen obtenida sino a la capacidad para mantener características de la imagen como bordes, contraste, etc. Por otro lado queremos utilizar distintos métodos de reconstrucción de la literatura para comprobar qué reducción es la más adecuada de las que proponemos.

Agradecimientos

Este trabajo ha sido parcialmente subvencionado por el Ministerio de Educación y Ciencia con el proyecto TIN2007-65981

Referencias

- [1] K. Atanassov, Intuitionistic fuzzy sets, In: VIIth ITKR Session, Deposited in the Central Science and Technology Library of the Bulgarian Academy of Sciences, Sofia, Bulgaria, 1983, pp. 1684-1697.
- [2] K. Atanassov, Intuitionistic fuzzy sets, *Fuzzy Sets and Systems* 20 (1986) 87-96.
- [3] H. Bustince, E. Barrenechea, M. Pagola, Generation Of Interval-Valued Fuzzy And Atanassov's Intuitionistic Fuzzy Connectives From Fuzzy Connectives And From $K(\alpha)$ Operators. Laws For Conjunctions And Disjunctions. Amplitude, International Journal of Intelligent Systems (2008) In Press
- [4] H. Bustince, J. Montero, M. Pagola, E. Barrenechea, D. Gmez: A Survey On Interval-Valued Fuzzy Sets. In W. Pedrycz, A. Skowron, V. Kreinovich (Eds.): *Handbook of Granular Computing* John Wiley and sons, New York, (2008), Chapter 22.
- [5] H. Bustince, J. Montero, E. Barrenechea, M. Pagola, Laws of Conjunctions and Disjunctions in Interval type 2 fuzzy sets. In: Proc. IEEE World Congress on Computational Intelligence, WCCI2008, Hong Kong, 2008.
- [6] H. Bustince, E. Barrenechea, M. Pagola, J. Fernandez, Interval-valued fuzzy sets constructed from matrices: Application to edge detection, *Fuzzy Sets and Systems* 160 (2009) 1819-1840
- [7] H. Bustince, M. Pagola, E. Barrenechea, J. Fernandez, P. Melo-Pinto, H.R. Tizhoosh, J. Montero, *Fuzzy Sets and Systems*, In Press.
- [8] T. Chaira, A.K. Ray, A new measure using intuitionistic fuzzy set theory and its application to edge detection, *Applied Soft Computing* 8(2) (2008) 919-927.
- [9] C. Cornelis, G. Deschrijver, E.E. Kerre, Advances and challenges in interval-valued fuzzy logic, *Fuzzy Sets and Systems* 157 (2006) 622-627.
- [10] G. Deschrijver, C. Cornelis, E.E. Kerre, On the representation of intuitionistic fuzzy T-norms and T-conorms, *IEEE Transactions on Fuzzy Systems* 12(1) (2004) 45-61.
- [11] M.B. Gorzalczyk, A method of inference in approximate reasoning based on interval-valued fuzzy sets, *Fuzzy Sets and Systems* 21 (1987) 1-17.
- [12] E. Karabassiss, M. E. Speksakis, An analysis of image interpolation, differentiation and reduction using local polynomial fits, *Graphical models and image processing*, 57 (3), (1995) 183-195.
- [13] H. Nobuhara, W. Pedrycz, S. Sessa, K. Hirota, A motion compression/reconstruction method based on max t-norm composite fuzzy relational equations, *Information Sciences* 176 (2006) 2526-2552.
- [14] R. Sambuc, *Function Φ -Flous, Application a l'aide au Diagnostic en Pathologie Thyroïdienne*, These de Doctorat en Medicine, University of Marseille (1975).
- [15] H.R. Tizhoosh, Image thresholding using type-2 fuzzy sets, *Pattern Recognition* 38 (2005) 2363-2372.
- [16] I. B. Turksen, Interval valued fuzzy sets based on normal forms, *Fuzzy Sets and Systems* 20(2) (1986) 191-210.
- [17] I.K. Vlachos, G.D. Sergiadis, Intuitionistic fuzzy information - Applications to pattern recognition, *Pattern Recognition Letters* 28(2) (2007) 197-206.
- [18] R.R. Yager, On ordered weighted averaging aggregation operators in multicriteria decisionmaking, *IEEE Trans. Syst. Man Cybern*, 18 (1988) 183-190.
- [19] R.R. Yager, Families of OWA operators, *Fuzzy Sets and Systems*, 59 (1993) 125-148.
- [20] G. Y. Yang, H.Z. Shu, C. Toumoulin, G.N. Han, L.M. Luo, Efficient Legendre moment computation for grey level images, *Pattern Recognition*, 39 (2006) 74-80.

COARSENESS FUZZY PARTITIONS APPLIED TO TEXTURE IMAGE RETRIEVAL

J. Chamorro-Martínez¹ P. Martínez-Jiménez¹ J.M. Soto-Hidalgo²

¹ Department of Computer Science and Artificial Intelligence, University of Granada, {jesus,pedromartinez}@decsai.ugr.es

² Department of Computer Architecture, Electronics and Electronic Technology, University of Cordoba, jmsoto@uco.es

Abstract

In this paper, the texture feature "coarseness" is modelled by means of a fuzzy partition on the domain of coarseness measures. The number of linguistic labels to be used, and the parameters of the membership functions associated to each fuzzy set are calculated relating representative coarseness measures (our reference set) with the human perception of this texture property. A wide variety of measures is studied, analyzing its capability to discriminate different coarseness categories. Data about the human perception of coarseness is collected by means of a pool and it is used to obtain a fuzzy partition adapted to the human perception of coarseness. This fuzzy partition is applied to texture image retrieval.

Keywords: Coarseness, fineness, fuzzy partition, fuzzy texture, image features, texture features, image retrieval.

1 INTRODUCTION

For analyzing an image several kind of features can be used. From all of them, texture is one of the most popular and, in addition, one of the most difficult to characterize due to its imprecision. For describing texture, humans use vague textural properties like *coarseness/fineness*, *orientation* or *regularity* [1, 2]. From all of them, the *coarseness/fineness* is the most common one, being usual to associate the presence of fineness with the presence of texture. In this framework, a fine texture corresponds to small texture primitives (e.g. the image in figure 1(A)), whereas a coarse texture corresponds to bigger primitives (e.g. the image in figure 1(I)).

There are many measures in the literature that, given an image, capture the fineness (or coarseness) presence in the sense that the greater the value given by the measure, the greater the perception of texture [3]. However, given a certain measure value, there is not an immediate way to decide whether there is a fine texture, a coarse texture or something intermediate; in other words, there is not a textural interpretation.

To face this problem, fuzzy logic has been recently employed for representing the imprecision related to texture. In many of these approaches, fuzzy logic is usually applied just during the process, being the output a crisp result [4, 5]. Other approaches try to model the texture and its semantic by means of fuzzy sets defined on the domain of a given texture measure. In this last framework, some proposals model the texture property by means of an unique fuzzy set [6], and other approaches define fuzzy partitions providing a set of linguistic terms [7, 8].

Focusing our study in the last type of approaches, two questions need to be faced for defining properly a fuzzy partition: (i) the number of linguistic labels to be used, and (ii) the parameters of the membership functions associated to each fuzzy set (and, consequently, the kernel localization). However, these questions are not treated properly in the literature. Firstly, the number of fuzzy sets are often chosen arbitrarily, without take into account the capability of each measure to discriminate between different categories. Secondly, in many of the approaches, just an uniform distribution of the fuzzy sets is performed on the domain of the measures, although is well known that measure values corresponding to representative labels are not distributed uniformly. In addition, from our knowledge, none of the fuzzy approaches in the literature considers the relationship between the computational feature and the human perception of texture, so the labels and the membership degrees do not necessarily will match with the human assessments.



Figure 1: Some examples of images with different degrees of coarseness

In this paper, we propose a fuzzy partition taking into account the previous questions. Firstly, in order to select the number of linguistic labels, we analyze the ability of each measure to discriminate different coarseness categories. For this purpose, data about the human perception of coarseness is collected by means of a pool. This information is also used to localize the position and size of the kernel of each fuzzy set, obtaining a fuzzy partition adapted to the human perception of coarseness-coarseness.

Moreover, we propose to apply the obtained fuzzy partition to texture image retrieval. The current image retrieval systems are based on features, such as color, texture or shape, which are automatically extracted from images. In this framework, a very important point to take into account is the imprecision in the feature descriptions, as well as the store and retrieval of that imprecise data. To deal with this vagueness, some interesting approaches introduce the use of fuzzy logic in the feature representation and in the retrieval process [9, 10]. These fuzzy approaches also allow to perform queries on the basis of linguistic terms, avoiding one of the drawbacks of the classical image retrieval systems, where the queries have to be defined on the basis of images or sketches similar to the one we are searching for. This way, the proposed fuzzy partition will be used to describe images in terms of their texture coarseness and the queries will be performed by using linguistic labels.

The rest of the paper is organized as follows. In sec-

tion 2 we present our methodology to obtain the fuzzy partition and its application to image retrieval. Results are shown in section 3, and the main conclusions and future work are summarized in section 4.

2 Fuzzy Partitions for Coarseness

As it was pointed, there is not a clear perceptual interpretation of the value given by a coarseness measure. To face this problem, we propose to define a fuzzy partition on the domain of a given coarseness measure. For this purpose, several questions will be faced: (i) what reference set should be used for the fuzzy partition, (ii) how many fuzzy sets will compound the partition, and (iii) how to obtain the membership functions for each fuzzy set.

Concern to the reference set, we will define the partition on the domain of a given coarseness-coarseness measure. From now on, we will note $\mathcal{P} = \{P_1, \dots, P_K\}$ the set of K measures analyzed in this paper, Π_k the partition defined on the domain of $\mathcal{P}_k$, N_k the number of fuzzy sets which compounds the partition Π_k , and $\mathcal{T}_k^i$ the i -th fuzzy set in Π_k . In this paper, the set $\mathcal{P} = \{P_1, \dots, P_K\}$ is formed by the $K = 17$ measures shown in the first column of table 1. It includes classical statistical measures, frequency domain approaches, fractal dimension analysis, etc. All of them are automatically computed from the texture image.

With regard to the number of fuzzy sets which compounds the partition, we will analyze the ability of each measure to distinguish between different degrees of coarseness. This analysis will be based on how the human perceives the coarseness-coarseness. To get information about human perception of coarseness, a set of images covering different degrees of coarseness will be gathered. These images will be used to collect, by means of a pool, human assessments about the perceived coarseness. From now on, let $\mathcal{I} = \{I_1, \dots, I_N\}$ be the set of N images representing coarseness-coarseness examples, and let $\Gamma = \{v^1, \dots, v^N\}$ be the set of perceived coarseness values associated to $\mathcal{I}$, with v^i being the value representing the degree of coarseness perceived by humans in the image $I_i \in \mathcal{I}$. We will use the texture image set and the way to obtain Γ described in [11].

Using the data about human perception, and the measure values obtained for each image $I_i \in \mathcal{I}$, we will apply a set of multiple comparison tests in order to obtain the number of coarseness degrees that each measure can discriminate (section 2.1). In addition, with the information given by the tests, we will define the fuzzy sets which will compound the partition (2.2).

Table 1: Result obtained by applying the algorithm proposed in [11]

Measure	N_k	Classes	$\bar{c}_5 \pm KW_5/2$	$\bar{c}_4 \pm KW_4/2$	$\bar{c}_3 \pm KW_3/2$	$\bar{c}_2 \pm KW_2/2$	$\bar{c}_1 \pm KW_1/2$
Correlation [3]	5	{1,2-4,5-6,7-8,9}	0.122±0.038	0.403±0.0272	0.495±0.0225	0.607±0.0133	0.769±0.0210
ED [12]	5	{1,2,3-5,6-8,9}	0.348±0.0086	0.282±0.0064	0.261±0.0063	0.238±0.0066	0.165±0.0061
Abbadeni [13]	4	{1,2-6,7-8,9}	-	5.672±0.2738	9.208±0.4247	11.12±0.2916	25.23±1.961
Amadasun [1]	4	{1,2-6,7-8,9}	-	4.864±0.271	7.645±0.413	9.815±0.230	19.62±1.446
Contrast [3]	4	{1,2-5,6-8,9}	-	3312±265.5	2529±295.5	1863±94.84	790.8±129.4
FD [14]	4	{1,2,3-8,9}	-	3.383±0.0355	3.174±0.0282	2.991±0.0529	2.559±0.0408
Tamura [2]	4	{1,2-6,7-8,9}	-	1.540±0.0634	1.864±0.0722	2.125±0.0420	3.045±0.0766
Weszka [15]	4	{1,2-6,7-8,9}	-	0.153±0.0064	0.113±0.0093	0.099±0.0036	0.051±0.0041
DGD [16]	3	{1,2-8,9}	-	-	0.020±0.0010	0.038±0.0017	0.091±0.0070
FMPS [17]	3	{1,2-8,9}	-	-	0.256±0.0477	0.138±0.0122	0.0734±0.0217
LH [3]	3	{1,2-8,9}	-	-	0.023±0.0010	0.052±0.0025	0.127±0.0096
Newsam [18]	3	{1,2-6,7-9}	-	-	0.1517±0.0425	0.2654±0.0466	0.4173±0.0497
SNE [19]	3	{1,2-8,9}	-	-	0.879±0.0182	0.775±0.0087	0.570±0.0232
SRE [20]	3	{1,2-8,9}	-	-	0.995±0.00026	0.987±0.00066	0.966±0.0030
Entropy [3]	2	{1,2-9}	-	-	-	9.360±0.124	8.656±0.301
Uniformity[3]	2	{1,2-9}	-	-	-	$1.3E^{-4} \pm 2.6E^{-5}$	$3.9E^{-4} \pm 1.9E^{-4}$
Variance[3]	1	-	-	-	-	-	-

2.1 DISTINGUISHABILITY ANALYSIS OF THE FINENESS MEASURES

As it was expected, some measures have better ability to represent fineness-coarseness than the others. To study the ability of each measure to discriminate different degrees of fineness-coarseness (i.e. how many classes can P_k actually discriminate), we propose to analyze each $P_k \in \mathcal{P}$ by applying a set of multiple comparison tests following the algorithm shown in [11]. This algorithm starts with an initial partition¹ and iteratively joins clusters until a partition in which all classes are distinguishable is achieved. In our proposal, the initial partition will be formed by the 9 classes used in our poll (where each class will contain the images assigned to it by the majority of the subjects), as δ the Euclidean distance between the centroids of the involved classes will be used, as ϕ a set of 5 multiple comparison tests will be considered (concretely, the tests of Scheffé, Bonferroni, Duncan, Tukey's least significant difference, and Tukey's honestly significant difference [21]), and finally the number of positive tests to accept distinguishability will be fixed to $NT = 3$.

From now on, we shall note as $\Upsilon_k = C_1^k, C_2^k, \dots, C_{N_k}^k$ the N_k classes that can be discriminated by P_k . For each C_r^k , we will note as $\bar{c}_r^k$ the class representative value. In this paper, we propose to compute $\bar{c}_r^k$ as the mean of the measure values in the class C_r^k .

Table 1 shows the parameters obtained by applying the proposed algorithm with the different measures

¹Let us remark that this partition is not the "fuzzy partition". In this case, the elements are measure values and the initial clusters the ones given by the pool

considered in this paper. The second column of this table shows the N_k classes that can discriminate each measure and the third column shows how the initial classes have been grouped. The columns from fourth to eighth show the representative values $\bar{c}_r^k$ associated to each cluster.

2.2 THE FUZZY PARTITIONS

In this section we will deal with the problem of defining the membership function for each fuzzy set compounding the partition. As it was explained, the number of fuzzy sets will be given by the number of categories that each measure can discriminate.

In this paper, trapezoidal functions are used for defining the membership functions. In addition, a fuzzy partition in the sense of Ruspini is proposed. Figure 2 shows some examples of the type of fuzzy partition used. To establish the localization of each kernel, the representative value $\bar{c}_r^k$ will be used (in our case, the mean). Concretely, this value will be localized at the center position of the kernel.

To establish the size of the kernel, we propose a solution based on the multiple comparison tests used in section 2.1. As it is known, in these tests confidence intervals around the representative value of each class are calculated (being accomplished that these intervals do not overlap for distinguishable classes). All values in the interval are considered plausible values for the estimated mean. Based on this idea, we propose to set the kernel size as the size of the confidence interval.

The confidence interval CI_r^k for the class C_r^k is defined as

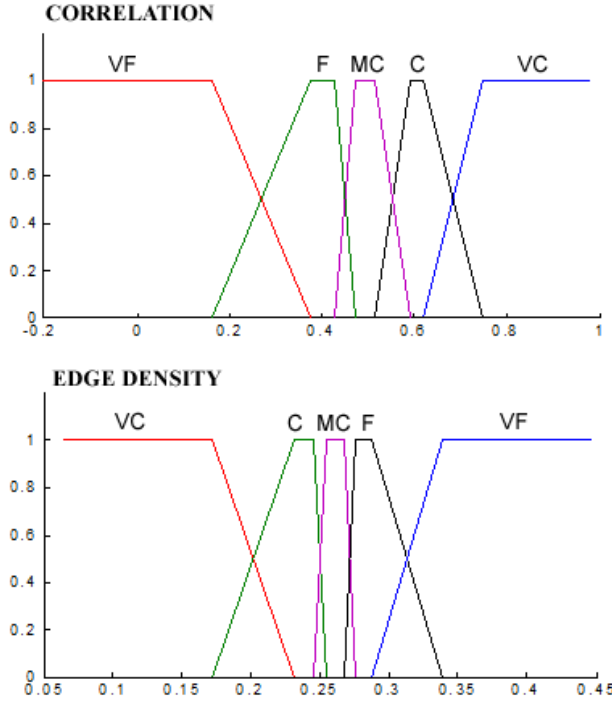


Figure 2: Fuzzy partitions for the measures Correlation and Edge Density. The linguistic labels are VC = very coarse, C = coarse, MC = medium coarse, F = fine, VF = very fine

$$CI_r^k = \bar{c}_r^k \pm 1.96 \frac{\bar{\sigma}_r^k}{\sqrt{\|C_r^k\|}} \quad (1)$$

where $\bar{c}_r^k$ is the class representative value, and $\bar{\sigma}_r^k$ is the estimated standard deviation for the class. Thus, the kernel size KW_r^k is

$$KW_r^k = 3.92 \frac{\bar{\sigma}_r^k}{\sqrt{\|C_r^k\|}} \quad (2)$$

and the endpoints of the kernel will be given by $\bar{c}_r^k \pm KW_r^k/2$. Table 1 shows these values for each measure and each class.

Figure 2 shows the fuzzy partitions for the measures of correlation and ED (the ones with higher capacity to discriminate coarseness classes).

2.3 APPLICATION TO TEXTURE IMAGE RETRIEVAL

As it was pointed, we propose to apply the obtained fuzzy partition to texture image retrieval. This fuzzy partition will be used to describe the representative texture coarseness in an image. To do this, we will extract the dominant texture coarseness. Intuitively, a

texture coarseness is dominant if it appears frequently in a given image, and it depends on the percentage of pixels where the texture coarseness appears. In order to calculate this percentage, for each pixel in the original image, a window centered on this pixel has been analyzed and its membership degree to each fuzzy set has been calculated. Thus, the percentage of pixels with texture coarseness C_r^k in the image under consideration will be

$$fr(C_r^k) = \frac{sc(C_r^k)}{NP} \quad (3)$$

where $sc(C_r^k)$ is the sigma-count histogram of the class C_r^k and NP is the number of pixels in the image.

Since frequent is an imprecise concept, dominance also is. It seems natural to model the idea of frequent apparition by means of a fuzzy set over the percentages, i.e., a fuzzy subset of the real interval $[0, 1]$. Hence, we define the fuzzy subset Dominant of texture coarseness as follows:

$$Dom(C_r^k) = \begin{cases} 0 & \text{if } fr(C_r^k) \leq u_1, \\ \frac{fr(C_r^k) - u_1}{u_2 - u_1} & \text{if } u_1 \leq fr(C_r^k) \leq u_2, \\ 1 & \text{if } fr(C_r^k) \geq u_2 \end{cases} \quad (4)$$

where u_1 and u_2 are two parameters such that $0 \leq u_1 < u_2 \leq 1$.

This way, we will describe images in terms of their dominant texture coarseness and the queries will be performed by using linguistic labels. The image retrieval process based on this dominant texture coarseness will be similar to the approach proposed in [22], based on dominant color descriptors.

3 RESULTS

In this section, the fuzzy partition defined for the measure "Correlation" (showed in Figure 2) will be applied in order to analyze the performance of the proposed model.

Let's consider Figure 3(A) corresponding to a mosaic made by several images, each one with a different increasing degree of coarseness. Figure 3(B-F) shows the membership degree to the fuzzy sets "very coarse", "coarse", "medium coarse", "fine" and "very fine", respectively, using the proposed model. For each pixel in the original image, a centered window of size 32×32 has been analyzed and its membership degree to each fuzzy set has been calculated. Thus, Figure 3(B) represents the degree in which the texture is perceived as "very coarse", with a white level meaning maximum degree, and a dark one meaning zero degree. It can be

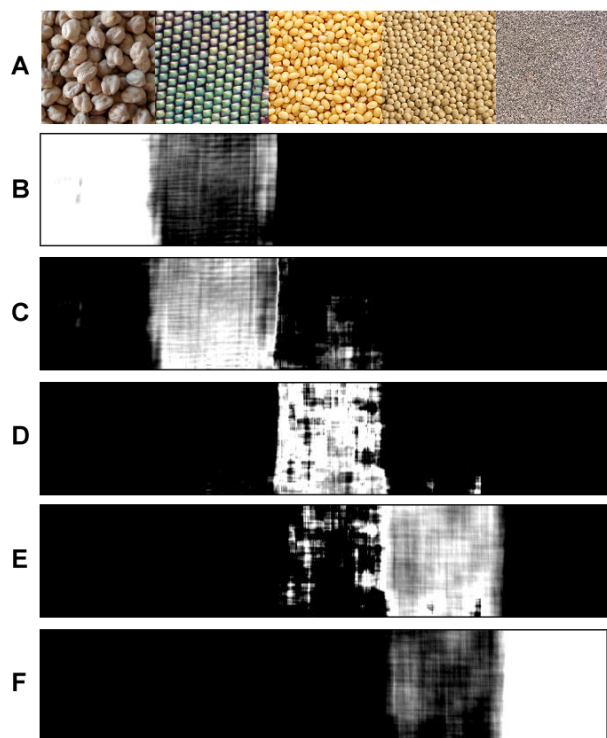


Figure 3: Results for a mosaic image. (A) Original image (B)(C)(D)(E)(F) Membership degree of each pixel to the sets "very coarse", "coarse", "medium coarse", "ne" and "very ne", respectively (the darker the pixel, the lower the membership degree)

noticed that our model captures the evolution of the perception degrees of coarseness.

Figure 4 presents an example where the proposed fuzzy partition has been employed for pattern recognition. In this case, Figure shows a microscopy image (Figure 4(A)) corresponding to the microstructure of a metal sample. The lamellae indicates islands of eutectic, which have to be separated from the uniform light regions. The brightness values in regions of the original image are not distinct, so texture information is needed for extracting the uniform areas. This fact is showed in Figure 4(B1,B2), where a thresholding on the original image is displayed (homogeneous regions cannot be separated from the textured ones as they "share" brightness values). Figure 4(C1) shows a mapping from the original image to its membership degree to the fuzzy set associated "very coarse". Thus, Figure 4(C1) represents the degree in which the texture is perceived as "very coarse" and it can be noticed that uniform regions correspond to areas with the maximum degree (bright grey levels), so if only the pixels with degree upper than 0.9 are selected, the uniform light regions emerge with ease (Figure 4(C2,C3)).

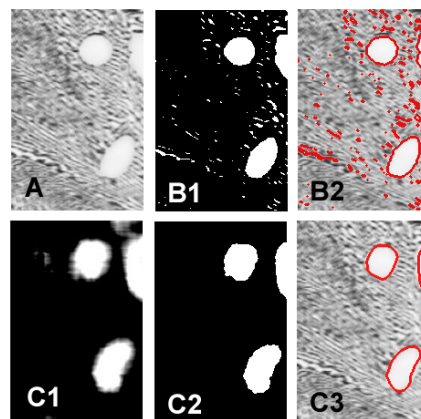


Figure 4: Example of pattern recognition (A) Original image (B1) Binary image obtained by thresholding the original one (B2) Region outlines of B1 superimposed on original image (C1) Membership degrees to the set "very coarse" obtained with our model from the original image (C2) Binary image obtained by thresholding C1 (C3) Region outlines of C2 superimposed on original image

Figure 5 presents an example of the application of the proposed fuzzy partition to image retrieval. We have used the dominant texture coarseness proposed in this paper as well as the dominant color descriptors shown in [22] in order to describe the images of a database containing about 700 color images. This example shows the results corresponding to the query *Dark orange color and very coarse texture*. It can be noticed that the images obtained match with the perception of very coarse textures.

4 CONCLUSIONS AND FUTURE WORKS

In this paper, a fuzzy partition for representing the coarseness/coarseness concept has been proposed. The number of fuzzy sets and the parameters of the membership functions have been defined relating coarseness measures with the human perception of this texture property. Pools have been used for collecting data about the human perception of coarseness, and the capability of each measure to discriminate different coarseness degrees has been analyzed. The results given by our approach show a high level of connection with the human perception of coarseness/coarseness. As future work, the performance of the fuzzy partition will be analyzed in applications like textural classification or segmentation.



Figure 5: Example of the application to image retrieval. Results corresponding to the query *Dark orange color and very coarse texture*

References

- [1] M. Amadasun and R. King. Textural features corresponding to textural properties. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):1264–1274, 1989.
- [2] H. Tamura, S. Mori, and T. Yamawaki. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, 8:460–473, 1978.
- [3] R.M. Haralick. Statistical and structural approaches to texture. *Proceedings IEEE*, 67(5):786–804, 1979.
- [4] M. Hanmandlu, V. K. Madasu, and S. Vasikarla. A fuzzy approach to texture segmentation. In *Proc. International Conference on Information Technology: Coding and Computing*, volume 1, pages 636–642, 2004.
- [5] A. Barcelo, E. Montseny, and P. Sobrevilla. Fuzzy texture unit and fuzzy texture spectrum for texture characterization. *Fuzzy Sets and Systems*, 158:239–252, 2007.
- [6] J. Chamorro-Martnez, E. Galn-Perales, J.M. Soto-Hidalgo, and B. Prados-Suarez. Using fuzzy sets for coarseness representation in texture images. In *Proceedings IFSA'07*, pages 783–792, 2007.
- [7] S. Kulkarni and B. Verma. Fuzzy logic based texture queries for cbir. In *Proc. 5th Int. Conference on Computational Intelligence and Multimedia Applications*, pages 223–228, 2003.
- [8] H.C. Lin, C.Y. Chiu, and S.N. Yang. Finding textures by textual descriptions, visual examples, and relevance feedbacks. *Pattern Recognition Letters*, 24(14):2255–2267, 2003.
- [9] C.-C. Hsu, W.W. Chu, and R.K. Taira. A knowledge-based approach for retrieving images by content. *IEEE Transactions on Knowledge and Data Engineering*, 8:522–532, 1996.
- [10] D. Sanchez, J. Chamorro-Martinez, and M.A. Vila. Modelling subjectivity in visual perception of orientation for image retrieval. *Information Processing and Management*, 39(2):251–266, 2003.
- [11] J. Chamorro-Martnez, P. Martnez-Jimnez, and J.M. Soto-Hidalgo. De ning a fuzzy partition for coarseness modelling in texture images. *Proceedings IFSA'09*, 2009.
- [12] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [13] N. Abbadeni, N. Ziou, and D.S. Wang. Autocovariance-based perceptual textural features corresponding to human visual perception. In *Proc. of 15th International Conference on Pattern Recognition*, volume 3, pages 901–904, 2000.
- [14] S. Peleg, J. Naor, R. Hartley, and D. Avnir. Multiple resolution texture analysis and classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (4):518–523, 1984.
- [15] J.S. Weszka, C.R. Dyer, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man and Cybernetics*, 6:269–285, 1976.
- [16] S.I. Kim, K.C. Choi, and D.S. Lee. Texture classification using run difference matrix. In *Proc. of IEEE 1991 Ultrasonics Symposium*, volume 2, pages 1097–1100, December 1991.
- [17] H. Yoshida, D.D. Casalino, B. Keserci, A. Coskun, O. Ozturk, and A. Savranlar. Wavelet-packet-based texture analysis for differentiation between benign and malignant liver tumours in ultrasound images. *Physics in Medicine and Biology*, 48:3735–3753, 2003.
- [18] S.D. Newsam and C. Kammath. Retrieval using texture features in high resolution multi-spectral satellite imagery. In *Data Mining and Knowledge Discovery: Theory, Tools, and Technology VI, SPIE Defense and Security*, April 2004.
- [19] C. Sun and W.G. Wee. Neighboring gray level dependence matrix for texture classification. *Computer Vision, Graphics and Image Processing*, 23:341–352, 1983.
- [20] M.M. Galloway. Texture analysis using gray level run lengths. *Computer Graphics and Image Processing*, 4:172–179, 1975.
- [21] Y. Hochberg and A.C. Tamhane. *Multiple Comparison Procedures*. Wiley, 1987.
- [22] J. Chamorro-Martnez, J.M. Medina, C. Barranco, E. Galn-Perales, and J.M. Soto-Hidalgo. Retrieving images in fuzzy object-relational databases using dominant color descriptors. *Fuzzy Sets and Systems*, 158(3):312–324, 2007.

CARACTERIZACIÓN DEL COLOR DEL CITOPLASMA EN IMÁGENES DE CITOLOGÍA CERVICO-VAGINAL

Santiago Romani¹

Pilar Sobrevilla²

Eduard Montseny²

Enrique Lerma³

¹ Universitat Rovira i Virgili, santiago.romani@urv.cat

² Universitat Politècnica de Catalunya, {pilar.sobrevilla, eduard.montseny}@upc.edu

³ Universitat Autònoma de Barcelona, elerma@hsp.santpau.es

Resumen

La afinidad cromática de los citoplasmas de las células que aparecen en un frotis cervical es uno de los principales parámetros a tener en cuenta para la detección precoz de alteraciones precancerosas del cuello uterino.

En el presente trabajo se presenta un algoritmo que permite determinar, de manera automática, los colores principales alrededor de los núcleos de células en imágenes de frotis vaginal. Ello permitirá, además de detectar los citoplasmas, evaluar su afinidad cromática, puesto que el algoritmo proporcionará el grado de similitud de cada píxel con los colores significativos de los citoplasmas.

Palabras Clave: Análisis de imágenes en color, técnicas difusas, detección de células.

1 INTRODUCCIÓN

El cáncer de cuello uterino es una enfermedad previsible cuando su diagnóstico es oportuno y se realiza un tratamiento adecuado de las lesiones pre-malignas [7].

Las alteraciones pre-malignas del cuello uterino son aquellas en las que las células tienen la potencialidad de convertirse en cancerígenas si no son tratadas de manera adecuada. El reconocimiento, en el momento oportuno de dichas alteraciones ofrece la oportunidad de prevenir el desarrollo del cáncer [9].

La prueba de Papanicolaou, o citología cérvico vaginal [5], es el primer paso en la detección del cáncer de cuello uterino. La prueba consiste en un simple frotis cervical que permite recoger muestras de células de un área específica en el cuello uterino. Las células se extienden en una lámina portaobjeto, que luego se tiñe y se fija con laca para evitar la alteración de las células.

Para poder determinar la existencia de posibles alteraciones celulares un patólogo, o citólogo entrenado, debe analizar, en el microscopio, la apariencia y características de las células del frotis cervical [8].

La prueba de Papanicolaou es considerada, generalmente, como una prueba muy específica para detección de lesiones de alto grado, o cáncer, pero con un grado moderado de sensibilidad. La alta especificidad significa que se identifica correctamente una elevada proporción de mujeres que no tienen lesiones de alto grado, o cáncer. La moderada sensibilidad significa que se identifica una proporción relativamente pequeña de mujeres que realmente tienen lesiones de alto grado, o cáncer [2].

Cuanto menor sea la sensibilidad mayor será el número de mujeres a las que no se les diagnostica la existencia de alteraciones pre-malignas o malignas (falso negativo). Por otra parte, a menor especificidad mayor será el número de diagnósticos falsos de la enfermedad (falsos positivos).

Dos son los factores que influyen en la obtención de “falsos negativos”: errores en la toma y procesamiento de la muestra; y errores en la búsqueda, identificación e interpretación de las células malignas.

Una de las razones por las que se puede obtener un resultado “falso positivo” es la dificultad y el carácter subjetivo e interpretativo de la evaluación citológica.

Para evitar los problemas anteriormente mencionados, y aumentar el grado de sensibilidad, sin reducir la especificidad, una solución es el desarrollo de sistemas de análisis y clasificación automática de las imágenes. Estos sistemas permitirían mejorar la productividad y la capacidad de diagnóstico de los patólogos.

La segmentación de la imagen es un proceso fundamental en cualquier sistema automático, ya que puede determinar, en última instancia, el éxito o fracaso del proceso de análisis [3]. En el caso de las imágenes de citología cérvico vaginal la segmentación de los núcleos es una tarea clásica en el análisis de imágenes de citología, y ha sido el centro de diversos trabajos de investigación [1, 6]. Por otra parte la interacción núcleo-citoplasma es de gran interés, ya que factores como: la posición del núcleo en el citoplasma, la relación de área núcleo/citoplasma, y la afinidad cromática de los citoplasmas de las células; son parámetros a tener en

cuenta para la detección precoz de alteraciones precancerosas del cuello uterino.

Siguiendo con nuestro trabajo para desarrollar un sistema automático de detección y análisis de células en imágenes de citología cérvico vaginal, que permita la interacción con el cito-patólogo [10, 11], en este trabajo se presenta un algoritmo que permite determinar, de manera automática, los colores principales alrededor de los núcleos de células en imágenes de frotis vaginal. Ello permitirá, además de detectar los citoplasmas, evaluar su afinidad cromática, puesto que el algoritmo proporcionará el grado de similitud de cada píxel con los colores significativos de los citoplasmas

2 ESTRUCTURA DEL SISTEMA

Como se ha dicho anteriormente, nuestro propósito final es el desarrollo de un sistema automático de detección y análisis de células en imágenes de citología cérvico vaginal, interactuando con el cito-patólogo. En consecuencia, nuestro objetivo es diseñar un sistema automático que permita eliminar los problemas relacionados con el procesamiento manual de las imágenes, pero considere tanto la información proporcionada por los especialistas, como el proceso seguido por los técnicos para analizar las imágenes.

Para determinar la estructura del sistema, teniendo en cuenta las restricciones anteriores, en primer lugar se realizó un análisis heurístico de diversas imágenes de citología, tomadas con lentes de 10 y 40 aumentos.

En este análisis se observaron las siguientes situaciones respecto a la distribución de las células:

- Células aisladas, cuyo citoplasma está rodeado únicamente por luz de fondo.
- Células parcialmente superpuestas, con áreas donde se mezclan dos citoplasmas.
- Células muy superpuestas, con áreas donde se mezclan tres o más citoplasmas.
- Células sin citoplasma, que sólo presentan núcleo

Dado que las regiones donde las células aparecen muy superpuestas son muy difíciles de analizar, incluso para un cito-patólogo experto (normalmente son consideradas como regiones de no interés), el sistema propuesto se ajusta a los otros tres casos.

Teniendo en cuenta las características de las imágenes, así como las restricciones anteriores, para cada núcleo detectado siguiendo el algoritmo propuesto en [11], el sistema aquí propuesto sigue los siguientes pasos: 1) *Obtención de histogramas de Tono y Saturación*, 2) *Obtención de las características de los colores relevantes del citoplasma*, teniendo en cuenta las características de color de cada región de citoplasma, y 3) *Clasificación de píxeles a los conjuntos difusos que definen los colores relevantes de cada célula*.

En el desarrollo del sistema se ha de tener además muy en cuenta que los colores del citoplasma son muy poco saturados, por lo que se pueden confundir fácilmente con el fondo o con colores de citoplasmas colindantes (ver figura 1). A continuación se resumen los puntos principales del sistema propuesto, según el algoritmo descrito en la figura 2.

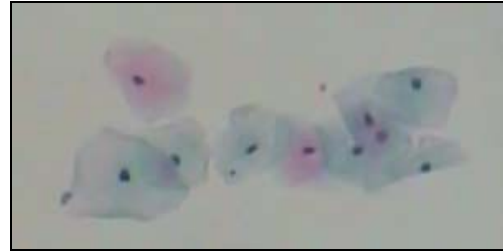


Figura 1: detalle de una imagen de citología con 40 aumentos (300x150 píxeles).

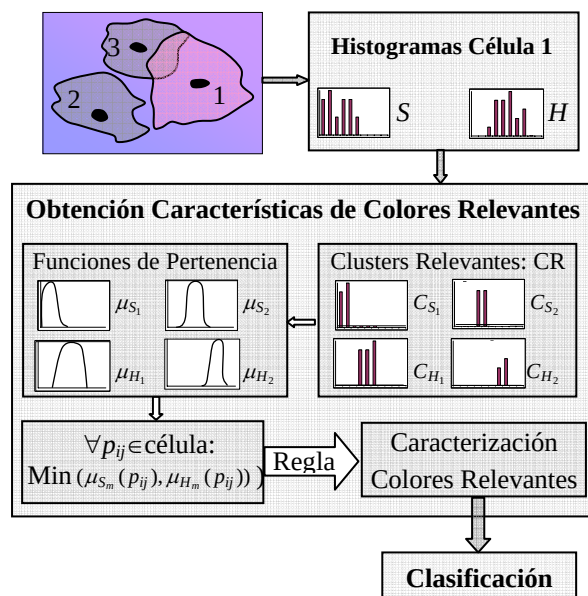


Figura 2: Esquema del sistema propuesto.

2.1 OBTENCIÓN DE LOS HISTOGRAMAS DE TONO Y SATURACIÓN

Para determinar los colores característicos de cada citoplasma se propone muestrear los valores de Tono y Saturación de algunos píxeles alrededor de cada núcleo. Para ello se sigue el siguiente proceso:

- Se introduce un radio aproximado, r , medido en píxeles, que permitirá encerrar los núcleos detectados dentro de círculos con radio igual a r . Este dato global depende de los aumentos de la imagen y del tamaño de las células, y se puede determinar heurísticamente a partir del contenido de diversas imágenes patrón.

- Para cada núcleo, N , se consideran radios orientados a 0, 45, 90, 135, 180, 225, 270 y 315 grados, situados alrededor del círculo que lo delimita.
- Para cada uno de los radios anteriores se obtiene la posición de los píxeles que están distribuidos uniformemente entre una y tres veces el radio del círculo. Los 24 píxeles considerados constituyen el conjunto de muestreo, M_N , para el núcleo N .

La figura 3 resalta los 24 puntos de muestreo (3 por radio) considerados para cada núcleo. Las etiquetas numéricas se utilizarán para referenciar las células del ejemplo.

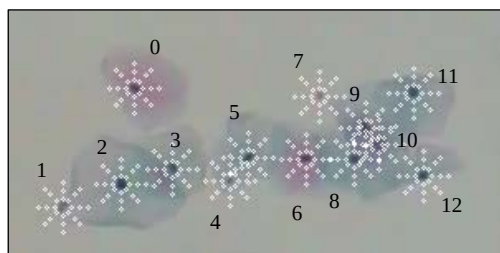


Figura 3: Posicionamiento de puntos de muestreo alrededor de los núcleos.

Dado el reducido número de muestras de que se dispone para obtener los histogramas de Tono y Saturación, el rango de ambas componentes se ha dividido en 32 particiones o “bins”, es decir, cada bin representará 8 valores del rango total.

Finalmente, para cada núcleo, se determina el bin de Tono y Saturación de cada píxel de muestreo y se obtienen los correspondientes histogramas. La figura 4 muestra el histograma de barras de los valores de Tono para la célula número 3 de la figura 3.

2.2 OBTENCIÓN DE LAS CARACTERÍSTICAS DE LOS COLORES RELEVANTES

La detección de las características de los colores relevantes está basada en la detección de los *clusters* correspondientes a los colores más relevantes del citoplasma. Para detectar dichos *clusters* se calcula lo que llamamos *Función de Distribución de Tensiones* o *Función de Carpa*. El comportamiento de la *FC* se basa en la siguiente idea:

“Si se dejara caer una cuerda sobre el histograma, cada bin ejercería una fuerza (o tensión) hacia arriba proporcional a su número de ocurrencias, y la cuerda tendería a caer hacia abajo por la fuerza de la gravedad. Además, cada punto de la cuerda recibiría y ejercería tensiones sobre sus dos puntos vecinos”.

Siguiendo la idea anterior, la posición final de los puntos de la cuerda será debida a un equilibrio de fuerzas que vendrá determinado por los siguientes parámetros:

- K_r : Resistencia de la cuerda a subir
- K_e : Resistencia de la cuerda a estirarse

Finalmente, los valles de las *FC* permitirán determinar los *clusters* significativos de los histogramas. Puesto que al variar los valores de K_r y K_e los valles se modifican, la selección de dichos valores determinará el número de *clusters*.

La figura 4 muestra cuatro versiones de la función de distribución de tensiones, una vez normalizada, para $K_r=0,5$ y un rango de valores para K_e . Como se puede observar, a medida que aumenta el valor de K_e , la cuerda se tensa más, con lo que desaparece un valle importante de la función y se abarca más posiciones de la componente Tono.

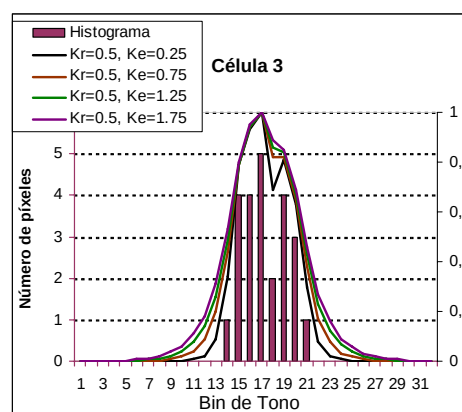


Figura 4: Histograma de Tono obtenido para la célula número 3 de la figura 3, junto con las correspondientes funciones de caracterización obtenidas para diferentes valores de los parámetros de resistencia K_r y K_e .

A continuación se explica cómo, a partir de las *FCs*, se obtienen las caracterizaciones de los colores relevantes de las zonas que rodean los núcleos.

Siguiendo el esquema de la figura 2, el primer paso consiste en obtener los *cluster* relevantes para Tono y Saturación. Dado que el proceso seguido para ambas componentes es el mismo, en la siguiente sub-sección se presenta el algoritmo general.

2.2.1. Obtención de los clusters relevantes difusos

Partiendo de la *Función de Carpa* del histograma, se propone separar sus *clusters* (picos) relevantes y, al mismo tiempo, obtener las funciones de pertenencia que definirán los *clusters* difusos. Para ello el algoritmo propuesto sigue los siguientes pasos:

- 1.- Seleccionar el bin b_m para el que *FC* toma valor 1,
- 2.- Buscar valles de *FC*: a partir de b_m avanzar a izquierda (b_{m-2}) y a derecha (b_{m+2}) mientras *FC* decrezca y no llegue a cero,
- 3.- Obtener una nueva función de carpa (*FC'*) para los bins que se encuentren en el rango seleccionado de

búsqueda (b_{m-2n}, b_{m-2n}).

- 4 – Si el número de ocurrencias dentro del rango de *bins* es significativo (en nuestro caso, mayor o igual al número de radios), eliminar las ocurrencias de los *bins* seleccionados y obtener la función de carpa para el resto de ocurrencias. Volver al paso 1.

Una vez finalizado el algoritmo se tendrá un conjunto de *Funciones de Carpa* $\{FC_{C_i}\}_{i=1}^l$ correspondientes a los l

clusters más relevantes $\{C_i\}_{i=1}^l$. Estas funciones, que siempre estarán centradas en el *bin* que presenta mayor índice de ocurrencias del *cluster*, y presentan una gradación de pertenencia decreciente hacia los *bins* vecinos de más ocurrencias del *cluster*, serán las que definan las funciones de pertenencia de los *clusters*.

La figura 5 muestra las dos *Funciones de Carpa* sobre los *clusters* detectados para el Tono de la célula etiquetada como 3 en la figura 3.

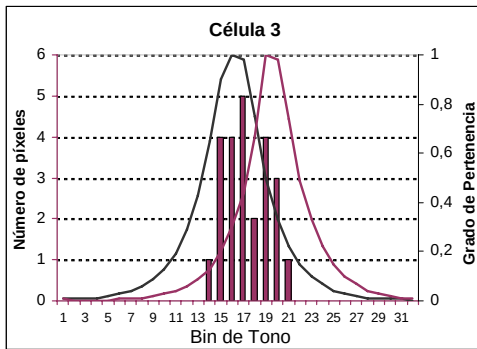


Figura 5: Ejemplo de definición de dos funciones de pertenencia para dos clusters en un histograma de Tono.

Se debe remarcar que, en la obtención de las funciones de pertenencia asociadas a cada cluster, los valores considerados para los parámetros de resistencia han sido: $K_r=0,25$ y $K_e=1,5$. De esta forma las funciones obtenidas son estrictas en el sentido de proporcionar mayor propagación del grado de pertenencia de los *bins* a los *clusters*.

2.2.2. Caracterización difusa de los colores relevantes

Siguiendo el proceso anteriormente descrito, para cada núcleo se habrán obtenido un conjunto de *clusters difusos de Tono*, $\{C_{H_i}\}_{i=1}^l$, y un conjunto de *clusters difusos de Saturación*, $\{C_{S_j}\}_{j=1}^{l'}$.

Para obtener la caracterización difusa de los colores relevantes de la región que rodea el núcleo (citoplasma), será necesario seleccionar, de entre todos posibles pares (*cluster difuso Tono*, *clusters difuso Saturación*), los que realmente correspondan a colores relevantes de las células.

A modo de ejemplo, supongamos que para un núcleo se han obtenido dos clusters difusos para *Tono*, y dos clusters difusos para *Saturación*, con funciones de pertenencia μ_{H_1} , μ_{H_2} , μ_{S_1} y μ_{S_2} . Puesto que los histogramas de *Tono* y *Saturación* se han obtenido de manera independiente, tendríamos que analizar cuál (o cuales) de los posibles pares (μ_{H_1}, μ_{S_1}) , (μ_{H_1}, μ_{S_2}) , (μ_{H_2}, μ_{S_1}) y (μ_{H_2}, μ_{S_2}) es representativo de los píxeles muestreados.

Para salvar este inconveniente, para cada núcleo N , si M_N es su conjunto de píxeles de muestreo, $\{\mu_{H_l}\}_{l=1}^{n_1}$ son las funciones de pertenencia asociadas a los *clusters difusos de Tono*, y $\{\mu_{S_k}\}_{k=1}^{n_2}$ las funciones de pertenencia asociadas a los *clusters difusos de Saturación*, dado $p_{ij} \in M_N$ se procede de la manera siguiente:

- $\forall l, k: 1 \leq l \leq n_1, 1 \leq k \leq n_2: \|(C_{H_l}, C_{S_k})\| = 0$
- $\forall l, k: 1 \leq l \leq n_1, 1 \leq k \leq n_2$ Se obtiene:

$$\mu_{lk}(p_{ij}) = \min(\mu_{H_l}(p_{ij}), \mu_{S_k}(p_{ij}))$$
- Si $\mu_{i,k_1}(p_{ij}) = \max\{\mu_{lk}(p_{ij}) : 1 \leq l \leq n_1, 1 \leq k \leq n_2\}$
 entonces $\|(C_{H_{l_1}}, C_{S_{k_1}})\| = i + 1$

Una vez que realizado el proceso anterior para todos los píxeles de M_N , se considerarán relevantes los pares de clusters (C_{H_l}, C_{S_k}) para los que $\|(C_{H_l}, C_{S_k})\| \geq h$. En el caso que aquí se presenta el umbral, h , es dos veces el número de píxeles considerados en los radios de muestreo.

2.3 CLASIFICACIÓN DE LOS PÍXELES A LOS CONJUNTOS DIFUSOS DE CADA CÉLULA

Una vez determinados los colores relevantes de cada célula, mediante los pares (H_l, S_k) correspondientes a los pares de clusters relevantes, (C_{H_l}, C_{S_k}) , el último paso del algoritmo consiste en clasificar los píxeles a los conjuntos difusos de cada célula.

Para ello, dado un píxel p_{ij} , y un color relevante, $c_{lk} = (H_l, S_k)$, cuyas funciones de pertenencia asociadas son μ_{H_l} y μ_{S_k} , el grado con el que p_{ij} es c_{lk} viene dado por:

$$\mu_{c_{lk}}(p_{ij}) = \min(\mu_{H_l}(p_{ij}), \mu_{S_k}(p_{ij}))$$

Además, podemos considerar que un píxel no se podrá clasificar a una célula que esté a más de una cierta distancia (euclídea) del centro del núcleo. Según las imágenes de ejemplo, se ha considerado que dicha distancia se puede limitar a 8 veces el radio del núcleo.

3 RESULTADOS

Para probar la eficacia del algoritmo propuesto se han obtenido los grados de pertenencia de todos los píxeles cercanos a cada núcleo de la imagen, para todos los conjuntos difusos asociados a los colores relevantes del citoplasma correspondiente.

En las siguientes imágenes se muestran los grados de pertenencia utilizando niveles de gris de manera que, cuanto más claro es el nivel de gris, mayor es el grado de pertenencia. Además, se han resaltado en blanco los bordes reales de las células para poder observar mejor cómo se clasifican los píxeles internos y externos a cada célula. Por último, se ha marcado en rojo el círculo de búsqueda de cada célula.

La figura 6 muestra la clasificación de los píxeles a la célula etiquetada 0 en la figura 3. Se ha de resaltar que, para esta célula se ha obtenido un único color característico. Como se puede apreciar, prácticamente toda el área del citoplasma tiene un buen grado de pertenencia a dicho color relevante. Por otra parte, se puede apreciar que los píxeles externos no llegan a un nivel mínimo, que hemos fijado en 0,10.

Se ha de resaltar así mismo que en los bordes del citoplasma se manifiestan ciertas divergencias, las cuales podemos considerar aceptables puesto que los colores de dichos píxeles divergen bastante del color general del citoplasma.

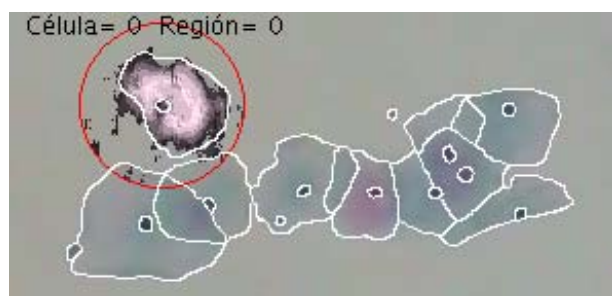


Figura 6: Clasificación de píxeles a la célula 0.

En la figura 7 se muestra el grado de pertenencia a dos zonas de color detectadas alrededor del núcleo 1. Este núcleo está pegado al citoplasma de la célula 2, y no tiene citoplasma. En la figura 7-a) se puede observar que prácticamente todos los píxeles del fondo, que es la región principal de color detectada, se clasifican bien.

En la figura 7-b) se clasifican buena parte de los píxeles del citoplasma de la célula 2, que es la segunda región de color detectada. En esta segunda clasificación, sin embargo, el área a detectar presenta más agujeros y zonas oscuras, debido principalmente a que se ha caracterizado con pocos puntos de muestreo (sólo 7 puntos).

Para la célula 2 se han detectado dos colores característicos, uno correspondiente al propio citoplasma de la célula, y el otro al área de solapamiento con el citoplasma de la célula 3.

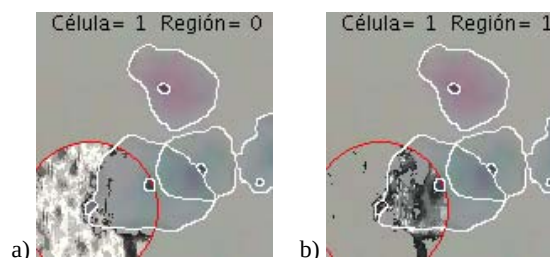


Figura 7: Clasificación de píxeles a la célula 1, región 0 (a) y región 1 (b).

Observando la figura 8-a), que muestra la clasificación al primer color, se aprecia que los resultados se ajustan bastante a lo esperado, aunque se observan regiones internas con píxeles sin clasificar debido a que el color del citoplasma en esas regiones se confunde con el color del fondo. Sin embargo, en la zona de solapamiento los grados de pertenencia son bastante oscuros, lo cual indica la buena separabilidad las dos regiones.

En la figura 8-b), correspondiente a la clasificación del color de la zona de solapamiento, se observan altos grados de pertenencia, salvo una pequeña mancha cerca del núcleo 3. Por otra parte, los píxeles de la zona no solapada muestran bajos grados de pertenencia, lo que refleja el buen comportamiento del algoritmo.

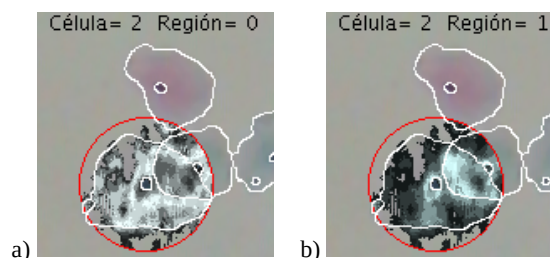


Figura 8: Clasificación de píxeles a la célula 2, región 0 (a) y región 1 (b).

Finalmente, la figura 9 corresponde a la clasificación de los píxeles respecto de los dos clusters obtenidos para la célula 3: En la figura 9-a) aparece la clasificación al citoplasma sin solapamiento, y en la figura 9-b) se representa la clasificación al citoplasma solapado con la célula 2, es decir, la región coincide con la de la figura 8-b), pero la caracterización es diferente puesto que los puntos de muestreo no son los mismos.

Para el caso del área sin solapamiento se observa que algunos segmentos cerca del borde superior derecho no se han clasificado como citoplasma, mientras que algunos píxeles, tanto de la zona de solapamiento como de los citoplasmas de las células 2 y 4, ofrecen un grado de pertenencia demasiado alto.

Sin embargo, en el caso del área con solapamiento, los resultados se ajustan más a lo esperado, definiendo dicha zona bastante mejor que la caracterización obtenida en la célula anterior.

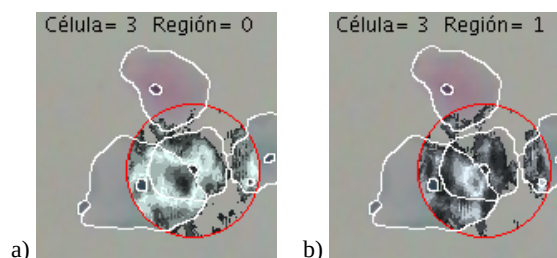


Figura 9: Clasificación de píxeles a la célula 3, región 0 (a) y región 1 (b).

En general podemos decir que las caracterizaciones representan convenientemente los colores relevantes de los citoplasmas, aunque el bajo Tono de color entre las diferentes zonas separables constituye un reto enorme para cualquier método de segmentación.

4 CONCLUSIONES

El método presentado en este artículo es capaz de determinar automáticamente los colores principales alrededor de los núcleos con el fin de detectar el citoplasma de cada célula. Además, obtiene una caracterización de dichos colores que permite una clasificación difusa de los valores de *Tono* y *Saturación* de los píxeles. Es de destacar que dicha caracterización se extrae satisfactoriamente a partir de un conjunto muy poco denso de valores de muestreo, algunos de los cuales pueden no formar parte de los patrones que se buscan.

En conclusión, el método propuesto constituye un sistema robusto de extracción de características de bajo nivel (cromacidad), que resultará de gran utilidad para otras etapas de consolidación de la detección de las células. Se ha de remarcar que la información proporcionada por el método propuesto se ha de incorporar en un sistema más general en el que se estudia y analiza, entre otras, información de la geometría de las células, como compacidad, bordes, etc.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el proyecto CICYT, TIN2007-68063.

Referencias

- [1] G. Begelman, E. Gur, E. Rivlin, M. Rudzsky, Z. Zalevsky. Cell nuclei segmentation using fuzzy logic engine. *International Conference on Image Processing*, vol. 5, Pág. 2937-2940, 2004.
- [2] F. Bray, B. Carstensen, H. Moller, et al. Incidence trends of adenocarcinoma of the cervix in 13 European countries. *Cancer Epidemiol Biomarkers Prevention*, vol. 14(9), Pág. 2191-2199, 2005.
- [3] H. D. Cheng, X. H. Jiang, Y. Sun, L. Wang. Color Image Segmentation: advances and prospects. *Pattern Recognition*, vol. 34(12), Pág. 2259-2281, 2001.
- [4] L. Na, L. J. Xin. Automatic color image segmentation based on anisotropic diffusion and the application in cancer cell segmentation. *The 1st International Conference on Bioinformatics and Biomedical Engineering*, Pág. 992-994, 2007.
- [5] G. N. Papanicolaou, N.F. Traut. *Diagnosis of uterine cancer by the vaginal smear*, New York: Commonwealth Fund; 1943.
- [6] M. E. Plissiti, A. Charchanti, O. Krikoni, D.I. Fotiadis. Automated segmentation of cell nuclei in PAP smear images. *Proc. IEEE International Special Topic Conference on Information Technology in Biomedicine*, 2006.
- [7] P. Sasieni, J. Adams, J. Cuzick. Benefit of cervical screening at different ages: evidence from the UK audit of screening histories. *Br J Cancer*, no 89, Pág. 88-93, 2003.
- [8] G. F. Sawaya, A. D. Brown, A.E. Washington, A.M. Garber. Clinical practice. Current approaches to cervical-cancer screening. *N Eng J Med*, no 344, Pág. 1603-1607, 2001.
- [9] S. M. Selvaggi. Implications of low diagnostic reproductibility of cervical and histologic diagnosis. *JAMA*, vol. 285, Pág. 1506-1507, 2001.
- [10] P. Sobrevilla, E. Montseny, E. Lerma. A Fuzzy-based Automated Cells Detection System for Color Pap Smear Tests –FACSDS–. *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models*, Springer, Berlin / Heidelberg, Pág. 657-674, 2008.
- [11] F. Vaschetto, E. Montseny, P. Sobrevilla, E. Lerma. Threecond-Fabio THREECOND: An Automated and Unsupervised Three Colour Fuzzy-Based Algorithm for Detecting Nuclei in Cervical Pap Smear Images. Aceptado para su presentación en ISDA 2010.

FUZZY STEERING CONTROL OF A HYDRAULIC TRACTOR AND LASER PERCEPTION IN OBSTACLE AVOIDANCE STRATEGIES

D. Martín, D.M. Guinea, M.C. García-Alegre, E. Villanueva, D. Guinea

Industrial Automation Institute (IAI), Spanish Council for Scientific Research (CSIC)
28500 Arganda del Rey, Madrid, Spain, {dmartin, dmguinea, maria, eugenio, domingo}@iai.csic.es

Abstract

Present work describes the implementation of a fuzzy steering controller for safe obstacle avoidance in unmanned navigation of a robot-tractor. Safety is warranted by the laser perception of the environment. The fuzzy controller integrates human knowledge on vehicle driving in the presence of unexpected objects, in a set of linguistic expressions. This steering control strategy is extremely useful for the reactive piloting of a commercial tractor retrofitted at the IAI-CSIC warehouses, to perform autonomous navigation in agricultural and open field environments. Some results are displayed on a global geo-referenced map of the IAI-CSIC campus in Arganda del Rey (Madrid), to show the real time avoidance behaviour.

Keywords: Fuzzy steering control, Robot-tractor, Hydraulic actuator, Avoid obstacle, laser perception.

1 INTRODUCTION

The development of an autonomous outdoor vehicle requires the integration of a plethora of subsystems and knowledge from multiple fields. From the mechanical design, control of the actuation systems, sensors and perception mechanisms, to the cognitive architectures, required to navigate in dynamic environments.

The automation of tractor navigation is still a challenge for many tasks, being one of them the selective distribution of pesticides, as it greatly reduces environment and human risks [4], [16]. The integration of sensors, actuators, processors and wireless communication systems would shift repetitive and sometimes risky low speed manual operations, to teleoperation and autonomous navigation of rural vehicles, such as tractors [14]. In addition, real-time perception of the environment is necessary to ensure safety in

environments where operators and vehicles work together.

On the other hand, fuzzy controllers have been widely applied for industrial processes due to their simplicity and effectiveness, as non-analytical model that embed human expert knowledge [1], [2], [10], [12]. They have been successfully used for a wide number of applications dealing with complex and non-linear processes, wherein they are proved to be robust and less sensitive to parametric variations and noise than conventional controllers [3], [18].

Finally, laser range finders are fundamental to get a safe navigation in outdoor environments [6], [17], [7], [19]. Safe and efficient obstacle avoidance strategies would be used in multi-objective autonomous navigation. To this aim, a fuzzy steering controller and a laser perception system have been developed and tested in a commercial hydraulic tractor.

2 EXPERIMENTAL SYSTEM

2.1. THE ROBOT-TRACTOR DEDALO

The vehicle used, as experimental platform, is a commercial tractor (AGRIA HISPANIA, S.A.) adequately retrofitted to get two navigation modes: teleoperated and autonomous GPS guided, in addition to the manual drive, Figure 1.



Figure 1: Hydraulic tractor, namely DEDALO.

The steering, the brake and the clutch of the tractor are activated through a multifunctional A/D PCI card (Advantech PCI-1710 series), plugged in the onboard industrial PC and allows the powering of the electro-valves. The fuzzy steering control strategy has been implemented with Matlab Data Acquisition Toolbox [11], which permits accurate communication between the onboard PC and the A/D card to manage both digital and analog signals.

The low-level control of the tractor is accomplished, managing the hydraulic actuation system based on the tractor pump, the hydraulic cylinders and the electro-valves. This hydraulic system gives pressure to the brake and clutch cylinders to shift brake and clutch actuators. The steering has been automated with an electro-valve since the tractor was steering hydraulic power-assisted. The whole hydraulic actuation system is managed by: (i) automatic/manual trigger, (ii) hydraulic pressure control, (iii) clutch electro-valve signal, (iv) brake electro-valve signal, (v) steering electro-valve control (left and right); to convert the electrical signals in mechanical displacement of the cylinder piston rods.

The control of the position of the brake and clutch is on/off, whereas the control of the position of the steering is gradual and measured by means of a potentiometer sensor coupled to the steering cylinder. The analog voltage of the potentiometer is read by the A/D PCI card. The conversion from analog voltages to precise steering angles is performed through a calibration process, by measuring several steering angles with its corresponding voltages. The results are shown in Table 1.

Table 1: Wheel angle and potentiometer measurements.

Wheel steering angle (degrees)	Potentiometer Voltages [V]
-35° (maximum left steering)	4.57 V
0° (no steering)	3.42 V
+35° (maximum right steering)	2.41 V

According to these measurements, a quadratic calibration function (1), to calculate the voltage at any steering angle in degrees.

$$V = 5.71e^{-5}\alpha^2 - 0.0309\alpha + 3.42 \quad (1)$$

So, the steps of the automatic steering control strategy are: (i) reading of the initial voltage from the analog port, (ii) introduction of the orientation of the target position in the Graphical User Interface (GUI), by the operator of the robot-tractor, (iii) calculation of the voltage

corresponding to the orientation of the target position, (iv) activation of the suitable electro-valve, to reach the target orientation, and (v) closing of the steering electro-valve once the target voltage is attained. As an example, lets the robot-tractor going straight ahead and the steering voltage being 3.42 volts. The tractor orientation at the target position implies a turn of 20° to the right, which corresponds to 2.82 V. Then,, the steering electro-valve is activated to shift the steering cylinder piston rod, so as the potentiometer voltage reach 2.82 V.

2.2. ONBOARD PROCESSOR

The onboard processing system is an industrial laptop, Kontron NotePAC Intel Core Duo 1.66 GHz processor, 2 GB RAM, with a PCMCIA WLAN. Initialization and visualization processes are executed on an external Tablet PC (Fujitsu-Siemens Stylistic ST5112), Intel Core Duo Processor (1.33 GHz) with 3.24 GB RAM. This Tablet PC has similar characteristic to a PDA but larger-size screen and higher processing capabilities [13], allowing a more complex user-interface, usually required in outdoor environments.

2.3. LASER RANGE FINDER

Laser scanners are increasingly used in automation and robotic applications as a depth sensing device useful for safety requirements. They have been tested in agricultural applications for measuring plant growth rate, tree volume, 3D imaging, and pattern recognition [9].

To reach safe piloting, a laser range finder has been installed in the tractor to get real-time depth images of the environment objects [15]. The 2D laser range finder emits an infrared laser beam, and receives the signal reflected by the object closer than 80 m. The laser provides polar coordinates, with 0.5 degrees of resolution in a 180° field, Figure 2.



Figure 2: Laser range finder in the tractor.

2.4 WIRELESS NETWORK FOR REMOTE CONTROL

Man-machine communication is nowadays accomplished with low cost wireless networks (WLAN), which can be easily installed and operated. In current work, communication relies on a Wireless Local Area Network (WLAN) that follows the standard IEEE-802.11b. The integration of the wireless communication enables the onboard processor to be accessed via the Intranet/Internet, from any WLAN workstation.

3. CONTROL ARCHITECTURE: AMARA

A cognitive architecture of agents of behaviour has been implemented as a suitable framework to deal with all the information acquired by the sensors, the domain knowledge, and the control processes [5]. Perceptual and motor processes are encapsulated under the framework of agents of behaviour [4]. The system offers three levels of competence with distinct system representation, and two ways to share information among processes: by message passing among agents and through shared memory. The system is implemented under the client/server paradigm, where processes running on different machines are appropriately connected by win-socket drivers and standard protocols. The main agent is the SERVER that runs on the onboard processor. This agent acts as a data server and as a client, sending sensor data upstream and accepting action commands during the run time. In addition to SERVER agent, several agents have been designed to mimic the sequence of actions performed by the human drivers. Agents can be ascribed to different competence layers. The middle layer is the piloting layer composed of three basic reactive agents of behaviour: ADVANCE, AVOID and STOP, Figure 3. They are coordinated by the adjacent upper level motor behaviour, namely WANDER. This motor agent relies on the perception cues of the working domain such as, Point, Obstacle, Local and Global maps, which are elaborated from raw sensor data, context information and Web services. Real-time interactions with the world, through the sensors, guide the activation of the reactive layer: {STOP, AVOID, ADVANCE} from a specific upper level behaviour.

3.1 AN AGENT OF BEHAVIOUR: WANDER

The module developed in current work, is a motor agent, namely WANDER. This agent implements the fusion of raw and elaborated data and representations, such as laser, GPS, objects and local and global grid map, respectively. From GPS and raw laser data, tractor position and depth of close objects are obtained by the perceptive agents, DETECT POSITION, DETECT OBSTACLE in addition to the deliberative agents DOWNLOAD GLOBAL MAP, GENERATE LOCAL MAP and VISUALIZATION.

3.2 AGENTS AND INFORMATION FLOW

In the AMARA architecture, perceptions flows bottom-up from raw data to abstract structures, and actions flow top-down from human commands to the hydraulic actuators, integrated in DEDALO tractor, Figure 3. The local grid map generated from raw laser data has a temporal persistency of about 2 seconds, and a spatial resolution of 20 cm, both much lower than the one obtained with raw laser data but with a higher confidence degree on the detected obstacles. These processes are embedded in the agent GENERATE LOCAL MAP that generates a perceptual structure, Object, defined through a set of features. The agent DOWNLOAD GLOBAL MAP is devoted to the acquisition of a georeferenced map of the working domain. This last agent is considered a deliberative one, and it delivers a data structure in the shared memory to be used by others agents, such as the agent VISUALIZATION.

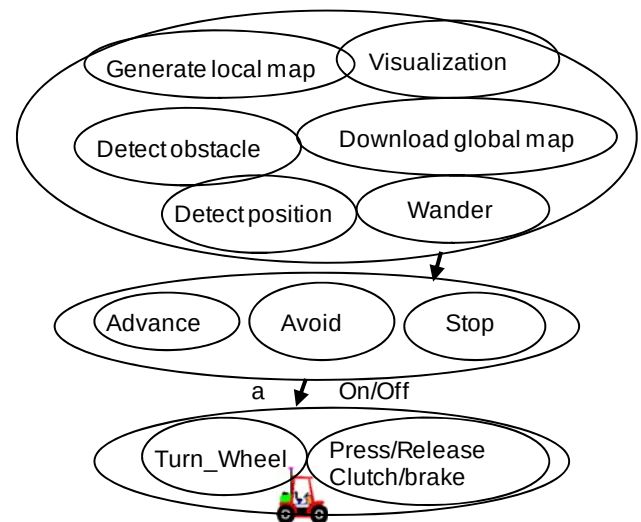


Figure 3: AMARA architecture.

4 FUZZY STEERING CONTROL IN OBSTACLE AVOIDANCE

The difficulty of deriving an analytic model of the steering system, as it is composed of several subsystems, recommend the design of a fuzzy controller, where the model is formulated by a set of linguistic rules expressed in natural language. These control rules establish the relationships between inputs and output [8], and encapsulate the non-linear dynamics of the whole actuation system to control the steering of the front wheels.

The Knowledge Base has been designed with the aid of MATLAB, to speed up the design and tuning of the fuzzy controller. The Fuzzy Logic Toolbox allows the definition of variables, linguistic labels and rules.

There are two types of variables: input and output. The first one accounts for the state of the system and the second for the control actions to be performed. The differentiation occurs implicitly, as variables defined as inputs always appear in the rule antecedent and outputs in the consequent. The implemented fuzzy-logic controller has two input variables: the “Obstacle aperture angle” and the “Obstacle distance”, and one output variable: the “Wheel Angle”.

Each variable is described by a set of linguistic terms, which are represented by means of membership functions, trapezoidal in this case. The membership functions are fixed in agreement with the expert knowledge and the empirical tests initially performed. The linguistic values of the two input variables are:

- “Obstacle aperture angle” {Small, Medium, Large}, Figure 4.
- “Obstacle distance” {Near, Far, VeryFar}, Figure 5.

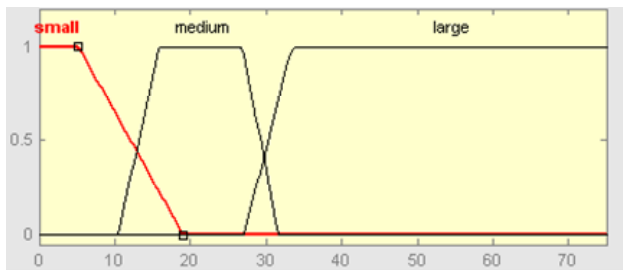


Figure 4 : Membership functions of the linguistic values of the “Obstacle aperture angle (angle)” variable.

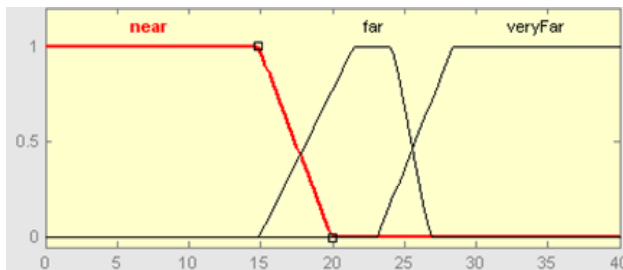


Figure 5: Membership functions of the linguistic values of the “Obstacle Distance” (distance) variable.

Five linguistic values, unevenly distributed along the universe of discourse of the variable, have been defined for the output “Wheel Angle”, Figure 6: {VeryVerySmall, VerySmall, Small, High, VeryHigh}.

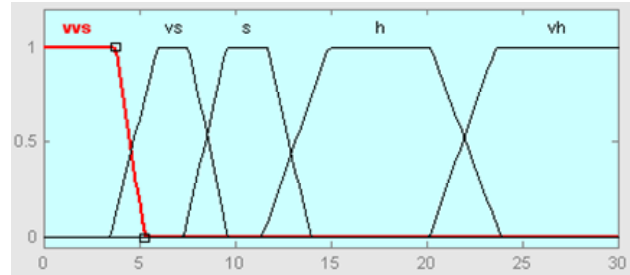


Figure 6: Membership functions of the linguistic values of the variable “Wheel Angle” (angleOut)

The Knowledge Base is composed of a set of propositions of the type: IF {(A is a) AND (B is b)} THEN {(C is c)} with a number of conjunctive constraints in the antecedent and only one constraint in the consequent. The designed fuzzy steering controller is composed of rules of the type: IF (“Obstacle aperture angle” is small) AND (“Obstacle distance” is near) THEN (“Wheel Angle” is small). All possible combinations of the variables terms, are represented in a fuzzy associative memory matrix, Table 2. The Knowledge Base of the fuzzy steering controller is displayed in Table 3.

Table 2 : Input, output matrix

Angle\Distance	Near	Far	Very far
Small	VeryHigh	High	Small
Medium	High	Small	VerySmall
Large	Small	VerySmall	VeryVerySmall

Table 3: Knowledge Base of the fuzzy steering controller

- R1. If (angle is small) and (distance is near) then (angleOut is vh) (1)
 R2. If (angle is small) and (distance is far) then (angleOut is h) (1)
 R3. If (angle is small) and (distance is veryFar) then (angleOut is s) (1)
 R4. If (angle is medium) and (distance is near) then (angleOut is h) (1)
 R5. If (angle is medium) and (distance is far) then (angleOut is s) (1)
 R6. If (angle is medium) and (distance is veryFar) then (angleOut is vs) (1)
 R7. If (angle is large) and (distance is near) then (angleOut is s) (1)
 R8. If (angle is large) and (distance is far) then (angleOut is vs) (1)
 R9. If (angle is large) and (distance is veryFar) then (angleOut is vvs) (1)

Matlab Fuzzy toolbox allows the design and visualization of the different rules and membership functions to adjust their numeric values to the restrictions of the control system. Figure 7 illustrates an example of each rule activation and output evaluation. The defuzzification algorithm used is the Centre of Mass, that weights the output of each rule in the Knowledge Base with the truth degree of its antecedent, to derive a mean crisp value. In the Figure 7, the two input values are: (i) “Obstacle aperture angle”, 30.6° , and (ii) “Obstacle distance”, 18 laser cells in the grid map. This grid map is built to represent detected objects, with a higher temporal persistency than the laser snapshot. Each cell represents an area of 400 cm^2 ($20 \text{ cm} \times 20 \text{ cm}$), that corresponds to a laser distance increment of 3.6 meters. Finally, the Centre of Mass defuzzification algorithm gives an output of 12° to the “Wheel angle”, to avoid the detected obstacle.

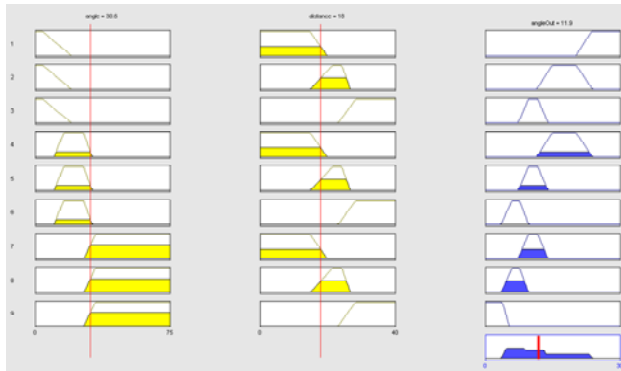


Figure 7: Rules activation and corresponding output

The control surface of the fuzzy steering controller is represented in Figure 8, with a control cycle of 200 ms, close to that of the laser snapshot.

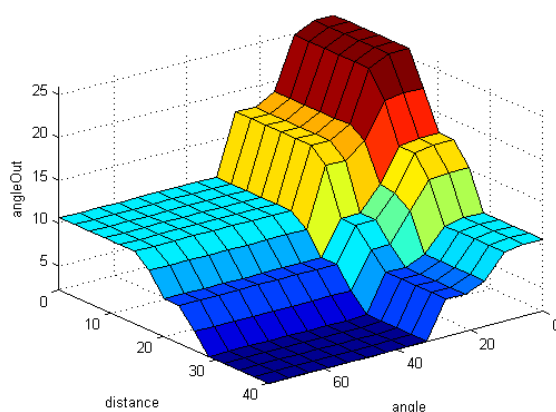


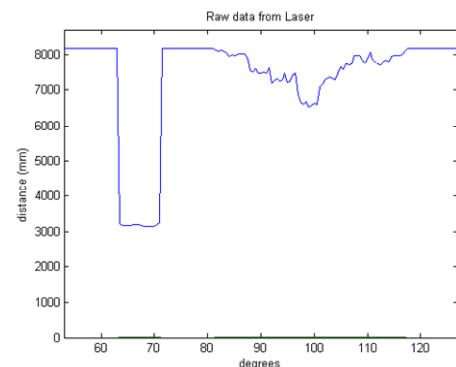
Figure 8: Control surface of the fuzzy steering controller.

5. RESULTS

The performance of the fuzzy controller has been tested in real time, under different working conditions, in the whole steering angle range $[-30^\circ \text{ to } +30^\circ]$. The Figure 9 shows a visual and laser images of a scene. The analysis of the raw laser data has been accomplished with MATLAB [11], due to the ease and fast development facilities offered by its multiple toolboxes and libraries in the initial stage of the development of a program. The original laser data, in polar coordinates, are processed to represent orientation (degrees) in the abscissa axis, and distance to the reflecting object surface in the ordinate axis.



(a)



(b)

Figure 9: (a) Colour visual image, (b) laser image.

The local grid map in Figure 10, represents the environment in front of the robot to a maximum distance of 8 meters, corresponding to the scene displayed in Figure 9. Yellow-green colour rectangles mark the points where the laser detects an obstacle in consecutive snapshots. To give more security and accuracy to the obstacle detected with the laser, an orange perimeter contour of one cell width surround every yellow obstacle in the local map. The tractor is represented by azure object at the bottom of the map.

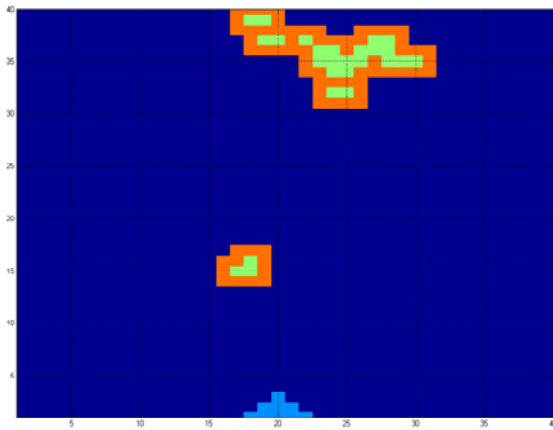


Figure 10: Local grid map representing detected obstacles in scene displayed in Figure 9.

The Figure 11 represents graphically the three regions (violet, orange and azure) for the “Obstacle distance” and three angles (red, azure and violet) for the “Obstacle aperture angle”, selected as input variables in the fuzzy controller. The regions are in the same order as they appear in the Figure 4 and Figure 5. In this way a “small” angle is represented by the “red” angle in the Figure 11. Depending on the “Obstacle aperture angle” and the “Obstacle distance” to the detected obstacle in the local grid map, the “Wheel angle” response will be different.

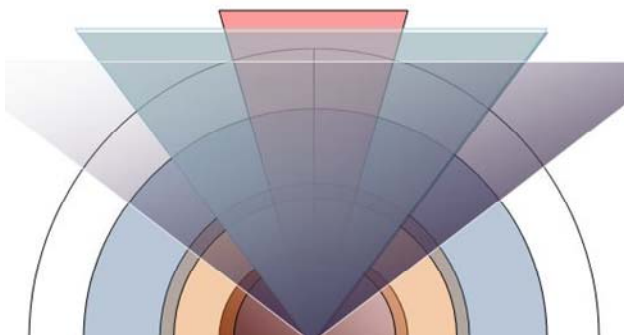


Figure 11: Obstacle detection regions.

The laser range finder has been configured to detect obstacles to a distance of 8 meters. This depth value allows the activation of the fuzzy steering controller in the AVOID agent, generating a deviation trajectory from the obstacle.

The Figure 12 displays three results of the operation of the fuzzy steering controller within the AVOID agent. The first experiment, Figure 12a, shows a person in front of the path of the robot-tractor. In this case, once the person is detected, the AVOID agent is activated and the fuzzy controller generates a deviation trajectory around the person. The Figures 12b and 12c display two experiments where the AVOID agent is activated to

deviate the tractor from the tree that appears close to its path.



(a)



(b)



(c)

Figure 12: Three scenes of activation of the AVOID agent fuzzy controller.

In these scenes, the robot-tractor was initially controlled by the ADVANCE agent, and upon detection of an obstacle closer than the safety distance, the AVOID agent is activated, by the coordination agent WANDER.

6 COMMENTS AND CONCLUSIONS

The fuzzy controller here developed is a non-analytic-model, based on a set of rules formulated by the expert, to adequately accomplish the obstacle avoidance strategies in an unmanned hydraulic tractor.

This fuzzy control system has been implemented and validated under different ground conditions, both in rough and flat terrains. In all of them, it has achieved the requirements imposed on both speed to reach a reference and low oscillation level.

The human-machine interface, developed in Matlab using a GUI, has been designed and implemented to visualize the environment while the tractor is guided by the WANDER agent of behaviour.

Obstacles depth images, derived from laser range finders, conveniently integrated with domain knowledge results in smart strategies for reactive tractor self-piloting. Intelligent and safe piloting strategies are fundamental issues in the development of unmanned navigation of vehicles in outdoor scenarios.

Acknowledgements

Authors thank ROBOCITY2030 (PRICIT-CAM- P - DPI-000176-0505) and PARSEVAL (MEC / CICYT_DPI_2006_14497) research projects for financial support in cognitive architectures for autonomous service robots.

References

- [1] C. V. Altrock, H.-O. Arend, B. Krause, C. Steffens, E. Behrens-Römmeler. Adaptive Fuzzy control applied to home heating system. *Fuzzy Sets and Systems*, 61, pp. 29 – 35, 1994.
- [2] P.J. Costa Branco, J.A. Dente. A Fuzzy relational identification algorithm and its applications to predict the behaviour of a motor drive system. *Fuzzy Sets and Systems*, 109, pp. 343 – 354, 2000.
- [3] M.C. García-Alegre. Seven years of fuzzy logic technology and industrial applications at IAI-CSIC. *5th Spanish Conf. On Technologies and Fuzzy Logic*, Murcia, pp. 25 – 34, 1995.
- [4] M.C. García-Alegre, F. Recio. Basic visual and motor agents for increasingly complex behavior generation on a mobile robot. *Autonomous Robots*, 5, pp. 19 – 28, 1998.
- [5] L. García-Pérez, M.C. García-Alegre, A. Ribeiro, D. Guinea. An agent of behaviour architecture for unmanned control of a farming vehicle. *Computers and Electronics in Agriculture*, vol. 60, pp. 39 – 48, 2008.
- [6] R. Keicher, H. Seufert. Automatic guidance for agricultural vehicles in Europe. *Computers and electronics in agriculture*, vol. 25, pp. 169 – 194, 2000.
- [7] A. Kelly, A. Stentz, O. Amidi, M. Bode, D. Bradley, A. Diaz-Calderon, M. Happold, H. Herman, R. Mandelbaum, T. Pilarski, P. Rander, S. Thayer, N. Vallidis, R. Warner. Toward Reliable Off Road Autonomous Vehicles Operating in Challenging Environments. *The International Journal of Robotics Research*, vol. 25, pp. 449 – 483, May–June 2006.
- [8] C.C. Lee. Fuzzy logic in control systems: fuzzy logic controllers - Part I and II. *IEEE Trans. System Man Cybernetics*, 20, pp. 404 – 435, 1990.
- [9] K.-H. Lee, R. Ehsani. Comparison of two 2D laser scanners for sensing object distances, shapes, and surface patterns. *Computers and Electronics in Agriculture*, vol. 60, pp. 250 – 262, 2008.
- [10] W. Li, X. Chang. Application of hybrid fuzzy logic proportional plus conventional integral-derivative controller to combustion control of stoker-fired boilers. *Fuzzy Sets and Systems*, 111, pp. 267 – 284, 2000.
- [11] Matlab. The MathWorks Inc. Natick, MA, United States, <http://www.mathworks.com/>
- [12] W. Meier, R. Weber, H.-J. Zimmerman. Fuzzy data analysis-Methods and industrial applications. *Fuzzy Sets and Systems*, 61, 19 – 28, 1994.
- [13] T. Oksanen, M. Ohman, M. Miettinen, A. Visala. Open and configurable control system for precision farming. *Automation Technology for Off-road Equipment Conference*, Kyoto, Japan, 2004.
- [14] F. Rovira-Más, Qin Zhang, John F. Reid. Stereo vision three-dimensional terrain maps for precision agriculture. *Computers and Electronics in Agriculture*, vol. 60, pp. 133 – 143, 2008.
- [15] SICK AG. Erwin-Sick-Str. 1, 79183 Waldkirch, Germany, <http://www.sick.com/>
- [16] D.C. Slaughter, D.K. Giles, D. Downey. Autonomous robotic weed control systems: A review. *Computers and Electronics in Agriculture*, vol. 61, pp. 63 – 78, 2008.
- [17] A. Stentz, C. Dima, C. Wellington, H. Herman, D. Stager. A System for Semi-Autonomous Tractor Operations. *Autonomous Robots*, vol. 13, pp. 83 – 104, 2002.
- [18] M. Sugeno. Industrial applications of fuzzy control. Amsterdam. The Netherlands: Elsevier, 1985.
- [19] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M.N. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T.M. Howard, S. Kolski, A. Kelly, M. Likhachev, M. McNaughton, N. Miller, K. Peterson, B. Pilnick, R. Rajkumar, P. Rybski, B. Salesky, Y.-W. Seo, S. Singh, J. Snider, A. Stentz, W.R. Whittaker, Z. Wolkowicki, J. Ziglar, H. Bae, T. Brown, D. Demitrish, B. Litkouhi, J. Nickolaou, V. Sadekar, W. Zhang, J. Struble, M. Taylor, M. Darms, D. Ferguson. Autonomous Driving in Urban Environments: Boss and the Urban Challenge. *Journal of Field Robotics*, vol. 25, pp. 425 – 466, 2008.

Reglas Difusas para el Control de un Vehículo en la 2009 Simulated Car Racing Competition: un Enfoque Ganador

Enrique Onieva¹ David A. Pelta² Joshué Pérez¹ Javier Alonso¹ Vicente Milanés¹

¹ Departamento de Informática Industrial. Instituto de Automática Industrial (CSIC).
{onieva, jperez, jalonso, vmilanes}@iai.csic.es

² Departamento de Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada.
dpelta@decsai.ugr.es

Resumen

El área de los videojuegos es un campo de aplicación excelente para las técnicas de Soft Computing. En particular, las carreras de coches simulados están ganando el interés de la comunidad científica por ser un excelente marco de prueba para aprendizaje, adaptación, razonamiento, evolución, etc. En esencia se trata de controlar el comportamiento de un coche en una serie de carreras donde existen otros coches.

Este trabajo presenta los resultados de nuestra contribución en la 2009 ‘*Simulated Car Racing Competition*’, en la que presentamos un coche conducido por conjunto de reglas difusas.

Palabras Clave: simulación de vehículos, carreras de coches, arquitecturas de control.

1 INTRODUCCIÓN

La Inteligencia Artificial se ha aplicado a los juegos ha desde sus inicios. Tradicionalmente, los juegos medían la habilidad del jugador para mover fichas en un tablero o jugar con cartas, todo ello, enmarcado en un conjunto de reglas que hacía que el proceso de generar un árbol de decisión fuera simplemente *cuestión de tiempo*. Desde la aparición de *Pong* en 1970 ha habido un fenomenal crecimiento en complejidad, diversidad y calidad de los videojuegos; tanto que, en muchos casos, los entornos virtuales planteados son muy similares al mundo real. Primeros ejemplos del interés de la AI por los juegos son los programas *Deep Blue* [2] para ajedrez y *Chinook* [10] para damas.

En los últimos años, el surgimiento de un gran número de interfaces de programación de aplicaciones (API) ha permitido probar técnicas de aprendizaje, planificación,

control, etc. en entornos continuos de gran similitud con el mundo real [5]. En particular una nueva generación de juegos usa técnicas de Soft Computing [3] en la creación de contenidos, adaptación de luminosidad, control de cámaras o, como ejemplo más obvio, el control automático de jugadores (Non Player Characters, NPC) [13].

Un ejemplo que nos interesa por la cercanía a nuestra investigación es la simulación de carreras de coches, ya que provee un excelente marco para probar aprendizaje, adaptación, razonamiento, evolución, etc. Por otra parte, los entornos de simulación actuales implementan con gran detalle la inercia, aerodinámica y demás aspectos físicos, dotando así a los simuladores de gran realismo, lo que permite la realización de pruebas propias de una rama de investigación tan interesante e importante como son los sistemas de transporte inteligente y el guiado automático de vehículos [7], [9].

En este trabajo se presenta nuestra contribución al 2009 *Simulated Car Racing Competition*: La creación de una arquitectura de control completa donde cada módulo será responsable de una acción considerada básica para el control de un coche en carrera: 1) Control de marchas, 2) Control de velocidad, 3) Determinación de la velocidad permitida, 4) Control del volante, 5) Gestión de oponentes y 6) Aprendizaje entre vueltas. Un elemento crítico es el control de la velocidad puesto que se debe ir lo más rápido posible, evitando salidas de pista, derrapes, etc. En nuestro caso se ha implementado un controlador difuso tipo Mamdani [6] con consecuentes de tipo singleton. El controlador de gran simplicidad dio buenos resultados.

El trabajo se estructura como sigue: la sección 2 hace una breve historia de la competición y describe el entorno de simulación TORCS. La sección 3 presenta la arquitectura de la última carrera, así como las mejoras realizadas con respecto a las dos anteriores. La sección 4 presenta los resultados en las tres carreras. Finalmente, la sección 5 presenta las conclusiones obtenidas del presente trabajo, así como líneas de investigación futuras.

2 HISTORIA DE LA COMPETICIÓN Y ENTORNO DE SIMULACIÓN TORCS

Las primeras competiciones tuvieron lugar en 2007, en los congresos del IEEE *Congress on Evolutionary Computation* (CEC) y *Computational Intelligence and Games Symposium* (CIG). Las carreras usaron un modelo computacional gráfica y mecánicamente simple, lo que propició un alto grado de participación [3]. En 2008, en el *IEEE World Congress on Computational Intelligence* (WCCI), la competición se basó en TORCS (*The Open Racing Car Simulator*). TORCS permitió simular carreras complejas tanto desde el punto de vista de la competición, ya que todos los competidores corrían en la misma pista, como físico, ya que TORCS simula con un alto grado de realismo las dinámicas de un vehículo [4]. Otra competición similar se realizó en el IEEE-CIG de 2008.

La última competición, *2009 Simulated Car Racing Competition*, ha consistido en las tres carreras que se citan a continuación y el resultado final ha sido la suma de las puntuaciones obtenidas en cada una de ellas.

- *IEEE Congress on Evolutionary Computation* (CEC09), Trondheim (Noruega), mayo de 2009
- *Genetic and Evolutionary Computation Conference* (GECCO09), Montreal (Canadá), julio de 2009
- *IEEE Congress on Computational Intelligence and Games* (CIG09), Milán (Italia), septiembre de 2009

El simulador TORCS (Figura 1) está escrito en C++ y disponible bajo licencia GPL a través de la web¹. Los aspectos que lo hacen interesante desde el punto de vista académico son:

- Es totalmente configurable y fácilmente extensible para implementar nuevas funcionalidades.
- Implementa una física motora (aerodinámica, consumo de combustible, tracción,...) altamente sofisticada, así como buenos gráficos 3D.
- Está diseñado para facilitar la implementación de controladores automáticos.

La organización puso a disposición de los participantes una arquitectura cliente-servidor que permite intercambiar información entre el controlador y el entorno. Así el controlador recibe información sensorial del entorno (Tabla 1) y envía las órdenes de control.

Una descripción detallada de la arquitectura y de la información sensorial disponible se encuentra en la web². Las variables que determinan las acciones de control son: *volante*, en $[-1,1]$, *acelerador* y *freno* en $[0,1]$ y *marcha* en -1 para marcha atrás, y $\{1, 2, 3, 4, 5, 6\}$ para el resto de marchas.

¹ <http://torcs.sourceforge.net>

² <http://cig.dei.polimi.it>



Figura 1: Entorno de Simulación TORCS.

Tabla 1: Descripción de la información sensorial.

NOMBRE	DESCRIPCIÓN
<i>Cabeceo</i> $[-\pi, \pi]$ rads	Ángulo formado por el vector director del coche y el eje central de la pista.
<i>Distancia</i> $[0, -]$ m	Distancia desde la línea de salida hasta la posición actual del vehículo.
<i>Marcha</i> $\{-1, 1, 2, 3, 4, 5, 6\}$	Marcha actual.
<i>RPM</i> $[0, 10.000]$	Revoluciones por minuto del motor del vehículo.
<i>Velocidad</i> $[0, 300]$ km/h	Velocidad actual del vehículo.
<i>Ruedas</i> $[0, -]$ rad/s	Vector de 4 elementos con la velocidad angular de cada una de las ruedas.
<i>Deriva</i>	Desviación lateral normalizada del vehículo. En $[-1, 1]$ si se encuentra dentro de la pista, otros valores si está fuera.
<i>Pista</i> $[0, 100]$ m	Vector de 19 sensores de proximidad orientados cada 10° en $[-90^\circ, 90^\circ]$. Miden la distancia al borde de la pista hasta 100m. Estos valores no son válidos si el coche se encuentra fuera de la pista.
<i>Oponentes</i> $[0, 100]$ m	Vector de 36 sensores de proximidad orientados cada 10° en $[-180^\circ, 180^\circ]$. Miden la distancia a un posible oponente hasta 100m.

Cada carrera del 2009, como las del 2008, tuvo dos fases: La clasificatoria, en la que un coche circula solo durante 200s en tres pistas, esta fase elige los ocho controladores que recorren mayor distancia. En la segunda fase los elegidos compiten entre sí en tres carreras, cada una de cinco vueltas al circuito, y repetida diez veces con diferentes parrillas iniciales. Los participantes puntúan en 10, 8, 6, 5, 4, 3, 2, y 1 respectivamente, del primero al último, sumando 2 puntos adicionales el que finalice con menos daños y otros 2 el que haga la vuelta más rápida.

3 ARQUITECTURA DE CONTROL

La figura 2 muestra un esquema global de la arquitectura de control presentada al GECCO09 y CIG09. En el CEC09 se presentó una arquitectura preliminar a la descrita en este trabajo [8].

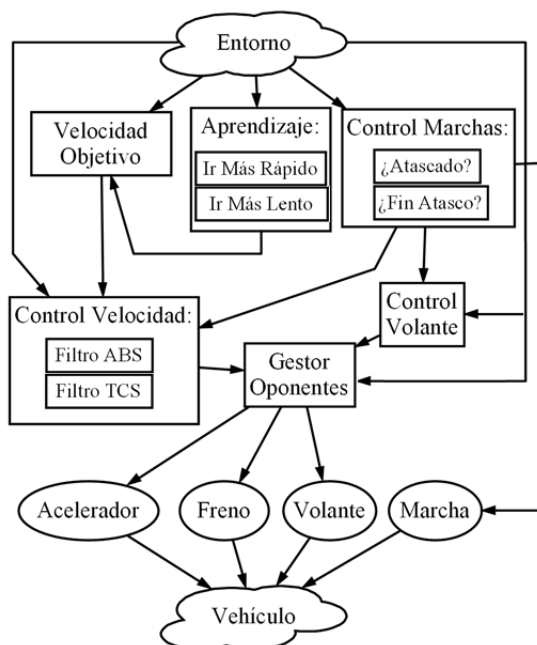


Figura 2: Arquitectura de Control.

En las siguientes secciones, damos una descripción detallada de cada uno de los módulos que la forman.

3.1. VELOCIDAD OBJETIVO

Tras las primeras pruebas, se concluyó que la clave para una conducción rápida era determinar correctamente la velocidad límite en cada uno de los tramos de pista. Esto permite no sólo recorrer la pista lo más rápido posible, sino, evitar salirse de ella, así como derrapar y perder el control del vehículo en una curva, chocar contra las barreras laterales,... El objetivo de este módulo es obtener ese valor de velocidad límite (V_{obj} , en adelante).

Es por ello que se optó por el uso de un controlador difuso para llevar a cabo esta tarea. El controlador se basó en un modelo computacional de coprocesador difuso llamado ORBEX [1] (Ordenador Borroso Experimental). ORBEX es capaz de definir sistemas difusos con funciones de pertenencia de tipo singleton. Sugeno [11] probó que el uso de singletons como consecuentes en las reglas difusas es adecuado para aplicaciones de control, donde el tiempo de respuesta es crucial. Es por ello el uso de este tipo de sistema difuso.

El controlador difuso utiliza como entradas tres de los 19 sensores de pista disponibles. Estos son, el sensor frontal, el máximo valor de los sensores que miden distancia en $\pm 10^\circ$ y el máximo valor de los sensores que miden distancia en $\pm 20^\circ$. Por ello las variables de entrada se llamarán *FRENTE*, *MAX10* y *MAX20*, respectivamente.

Se optó por el uso de estas variables como entrada del sistema ya que nos proporcionarán, de cierta manera, una forma de predecir características de la pista con una antelación de hasta 100m (máximo valor posible para estos sensores). Un valor máximo de *FRENTE* indicará vía libre para circular a la máxima velocidad, mientras que, si este valor es bajo, y *MAX10* es alto, indicará que el vehículo se está aproximando a una curva, el radio de dicha curva dependerá de los valores concretos de estas variables. El usar valores máximos entre dos sensores se debe a cuestiones de simetría, ya que la velocidad permitida debe ser similar ante curvas de igual radio, ya sean hacia la izquierda o hacia la derecha.

Para codificar las entradas se utilizaron tres funciones de pertenencia trapezoidales, llamadas *Bajo*, *Medio* y *Alto*. La figura 3 muestra una representación gráfica de dichas funciones.

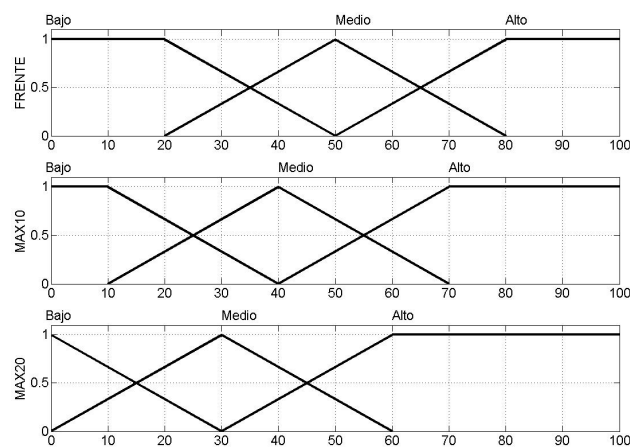


Figura 3: Funciones de pertenencia de las entradas.

Como consecuentes de las reglas se utilizaron los singletons mostrados en la figura 4.

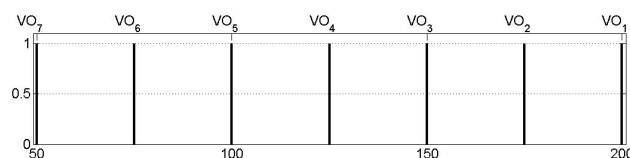


Figura 4: Singletons de salida.

La base de reglas está formada por 7 reglas difusas y una regla no difusa (8), que permite una velocidad máxima en tramos rectos; dicha regla es aplicada siempre que el módulo de aprendizaje entre vueltas no detecte riesgo de salida de la pista en el tramo.

La base de reglas utilizada está formada por el siguiente conjunto de reglas:

1. Si FRENTE es Alto Entonces V_{obj} es VO_1
2. Si FRENTE es Medio Entonces V_{obj} es VO_2
3. Si FRENTE es Bajo Y MAX10 es Alto Entonces V_{obj} es VO_3
4. Si FRENTE es Bajo Y MAX10 es Medio Entonces V_{obj} es VO_4
5. Si FRENTE es Bajo Y MAX10 es Bajo Y MAX20 es Alto Entonces V_{obj} es VO_5
6. Si FRENTE es Bajo Y MAX10 es Bajo Y MAX20 es Medio Entonces V_{obj} es VO_6
7. Si FRENTE es Bajo Y MAX10 es Bajo Y MAX20 es Bajo Entonces V_{obj} es VO_7
8. Si FRENTE=100 O MAX10=100 Entonces V_{obj} =300

El módulo de aprendizaje multiplica por un factor (1 en la primera vuelta) la salida de este módulo, en función de si la velocidad debe ser reducida o ampliada en un determinado tramo de la pista.

3.2. CONTROL DE MARCHAS

La funcionalidad de este módulo es doble. Primero, estará encargado de aumentar o disminuir la marcha actual en función del valor de *RPM*, y segundo, será el encargado de detectar si el coche requiere usar la marcha atrás (se ha quedado atascado, ha derrapado,...) así como cuándo debe dejar de usarse y volver a primera marcha para continuar la carrera.

Según la marcha actual, ésta será aumentada si *RPM* es superior a un valor entre 8000 y 9000 y será reducida si *RPM* es inferior a un valor entre 3000 y 3500. La detección de la necesidad de la marcha atrás se calculará en función del *cabeceo* y *deriva* del vehículo.

3.3. CONTROL DE VELOCIDAD

Por motivos prácticos y para evitar problemas, las salidas *acelerador* y *freno* se codifican como una única variable *pedales*, en $[-1, 1]$ representando así, acciones sobre el freno cuando *pedales*<0 y sobre el acelerador cuando *pedales*>0. El valor de dicha variable es una función exponencial de la diferencia entre *velocidad* y V_{obj} .

Los filtros ABS y TCL se añadieron tras la primera competición, al observarse que el vehículo a veces derrapaba perdiendo el control debido a que las acciones sobre los pedales eran muy bruscas, por lo que se implementaron funciones que reducían dichas acciones cuando la diferencia entre *velocidad* y las velocidades calculadas por los valores *ruedas* superaba un determinado umbral (1.5km/h). En ese caso, a la salida se disminuye un quinto de dicha diferencia.

3.4. CONTROL DEL VOLANTE

Para el control del volante, se distinguen tres casos:

1. Marcha normal, si el vehículo avanza dentro de la pista. En este caso se utiliza una media ponderada entre el sensor de *Pista* con mayor valor y sus dos adyacentes, logrando así, dirigir al coche en la dirección con mayor distancia libre.
2. Si el vehículo se encuentra fuera de la pista los sensores de pista no dan valores válidos, en esta situación se usa una función proporcional al *cabeceo* y *deriva* del vehículo.
3. Si el vehículo va marcha atrás, únicamente se utiliza una función inversa al *cabeceo*, para recuperar la orientación y poder continuar la marcha.

3.5. GESTOR DE Oponentes

Este módulo se introdujo porque TORCS retira de la carrera y pone en la última posición al vehículo cuyos daños superan un determinado umbral. Por ello conviene modificar levemente las acciones de control sobre los pedales y el volante cuando hay riesgo de colisión. Tiene tres objetivos bien diferenciados:

1. Para hacer adelantamientos. Se modifica la acción inferida por el controlador del volante cuando se detecta un oponente en un determinado radio. El valor de la modificación dependerá cuál de los sensores de oponentes ha detectado un valor inferior al umbral, y variará en ± 0.1 y ± 0.15 .
2. Para evitar colisiones. En caso de detectar un vehículo a una distancia frontal inferior a 10 m, se actúa sobre el freno (*pedales*=-0.5) hasta reducir la velocidad al 75% de V_{obj} , para minimizar los daños.
3. Se mejoró el controlador presentado al CEC09, añadiéndole una nueva función que implementa un efecto de "volantazo". Para ello, a la acción sobre los pedales, se le añade una acción sobre el volante, más brusca que la primera (± 0.25), logrando así obtener un efecto *volantazo* ante una posible colisión.

3.6. APRENDIZAJE ENTRE VUELTAS

Este módulo se añadió tras la primera competición, para evitar salidas sistemáticas en pistas con curvas muy pronunciadas donde el tiempo de reacción es mínimo. El módulo, usando *distancia*, reconoce los tramos en los que el vehículo no ha tenido tiempo de reacción y se ha salido de la pista y, antes de volver entrar en ellos (con una antelación de 200m) reducirá V_{obj} al 80%.

Además, reconoce largos tramos rectos para dar el valor máximo a V_{obj} (300km/h), sabiendo que no hay ninguna curva próxima.

4 RESULTADOS OBTENIDOS

Los resultados que se muestran han sido obtenidos en los contextos de los congresos CEC09, GECCO09 y CIG09. La arquitectura de control presentada recibe el nombre de *Onieva&Pelta*, en adelante.

En el CEC09 se presentó un prototipo preliminar de arquitectura, se presentaron 6 controladores, incluyendo al ganador de la anterior edición de la competición, llamado *Champ2008*. Las tres pistas utilizadas en esta carrera se muestran en la figura 5.

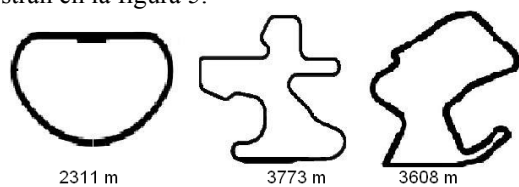


Figura 5: Pistas usadas en CEC09.

En la fase clasificatoria, nos clasificamos en el tercer, segundo y cuarto puesto, para cada una de las pistas. Los resultados de la fase final aparecen en la tabla 2, donde se muestran, para cada pista, la media de las puntuaciones obtenidas en las 10 carreras realizadas para cada uno de los competidores, así como la puntuación final.

Tabla 2: Resultados finales obtenidos en CEC09.

	Pista1	Pista2	Pista3	Total
Cobostar	12	7.5	9	28.5
Onieva&Pelta	8	9	5	22
Champ2008	5	10	5	20
Mr. Racer	3	6	10	19
Perez&Saez	6	4	6	16
RedJava	5	5	4	14

En la primera carrera obtuvimos la segunda posición. La mala puntuación en la tercera pista se debe a que el vehículo siempre se salía de la pista en tres lugares, eso motivó la incorporación del módulo de aprendizaje. Por otra parte, el exterior de la tercera pista es de arena, por lo que al salirse el coche acababa derrapando y perdiendo el control, lo que motivó el añadir filtros para suavizar las acciones sobre los pedales en dichas situaciones.

En GECCO09 se presentó la arquitectura tal y como se describe en el presente trabajo. Participamos 11 competidores, las pistas utilizadas se muestran en la figura 6.

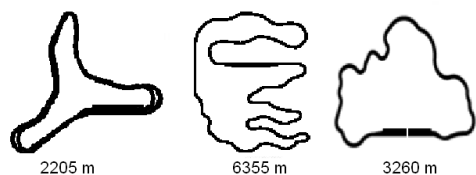


Figura 6: Pistas usadas en GECCO09.

En la fase clasificatoria obtuvimos respectivamente el primer, segundo y segundo puesto. Hay que notar que la primera de las pistas es de tierra, por lo que los filtros sobre las acciones de los pedales jugaron un importante papel en esa primera clasificación. Únicamente 8 se clasificaron para la fase final. Los resultados de esta fase se resumen en la tabla 3.

Tabla 3: Resultados finales obtenidos en GECCO09.

	Pista1	Pista2	Pista3	Total
Onieva&Pelta	12	10	10	32
Champ2008	8	7	8	23
Cobostar	3.5	5.5	7.5	16.5
Epic	4	5.5	3.5	13
Simplicity	5	3.5	4	12.5
DRT	2.5	3.5	5	11
Perez&Saez	3.5	4	3.5	11
Jorge Fuentes	3	3	2.5	8.5

En esta carrera fuimos los primeros, a notable distancia incluso de *Champ2008* y de *Cobostar*, que en la primera habían sido nuestros principales oponentes.

Al CIG09 acudimos con el mismo controlador usado en la carrera anterior. Hubo 13 participantes y 8 clasificados para la fase final. La figura 7 muestra las pistas utilizadas.



Figura 7: Pistas usadas en CIG09.

En la fase clasificatoria obtuvimos el segundo lugar en las tres pistas. Los resultados de la fase final se resumen en la tabla 4. En la carrera obtuvimos un segundo puesto, a únicamente un punto de *Cobostar*, y con una gran diferencia con el resto de competidores.

Tabla 4: Resultados finales obtenidos en CIG09.

	Pista1	Pista2	Pista3	Total
COBOSTAR	11	7	12	30
Onieva&Pelta	9	12	8	29
Wolf-Dieter	4	5.5	6	15.5
Jorge Muñoz	6	3.5	5	14.5
Champ2008	4.5	4	4	12.5
Mr. Racer	3.5	6	2.5	12
RedJava	2.5	4	4	10.5
Perez&Saez	3	2.5	4	9.5

Finalmente, en la tabla 5 se muestran los resultados finales del campeonato, siendo este la suma de las puntuaciones obtenidas a lo largo de las tres competiciones. Siendo el resultado obtenido el de campeones del 2009 *Simulated Car Racing Championship*.

Tabla 5: Resultados y clasificación final. (NP: No presentado, NC: No clasificado)

	CEC 09	GECCO 09	CIG 09	Total
Onieva&Pelta	22	32	29	83
COBOSTAR	28.5	16.5	30	75
Champ2008	20	23	12.5	55.5
Perez&Saez	16	11	12.5	36.5
Mr. Racer	19	NC	12	31
RedJava	14	NC	10.5	24.5
Jorge Muñoz	NP	8.5	14.5	23
Wolf-Dieter	NP	NP	15.5	15.5
Epic	NP	13	NC	13
Simplicity	NP	12.5	NC	12.5
DRT	NP	11	NC	11
Witold	NP	NC	NC	0
Ebner&Tiede	NP	NP	NC	0

5 CONCLUSIONES Y TRABAJO FUTURO

Este artículo presenta nuestra contribución a las carreras de coches simulados celebradas en los tres congresos internacionales que forman la *2009 Simulated Car Racing Competition*. Se ha presentado una arquitectura de control completa, cuyos módulos controlan aspectos básicos de la conducción. Debido a la importancia de determinar la velocidad permitida, se ha utilizado un controlador difuso, técnica no utilizada hasta ahora por ninguno de los participantes, ni de esta, ni de la pasada edición.

Los resultados muestran claramente la utilidad de los controladores difusos en el mundo de los videojuegos, en concreto, en las carreras de coches. A partir de ahora estamos en condiciones de mejorar y ajustar los distintos módulos con técnicas evolutivas por ejemplo, así como de aplicar técnicas de Soft Computing en el diseño automático de arquitecturas alternativas.

Por otra parte, módulos de la arquitectura son susceptibles de ser mejorados o modificados, así queda abierta la posibilidad de incluir nuevos comportamientos y estrategias de carrera, tales como razonamiento temporal para predecir situaciones futuras, o aprendizaje para mejorar a lo largo de la carrera.

Agradecimientos. Este trabajo se ha realizado gracias a los proyectos TRÁNSITO (TRA2008-06602-01, ENVITE Ministerio de Fomento (T7/ 2006), GUIADE (Exp. N° P9/08), el proyecto TIN2008/01948 del ministerio de Ciencia e Innovación y el proyecto P07-TIC02970 de la Junta de Andalucía.

Referencias

- [1] R. García, T. de Pedro. Modeling a Fuzzy coprocessor and its programming language, *Mathware and Soft Computing*, vol. 5(2-3), Pág. 167-174, 1998.
- [2] F. Hsu. Behind Deep Blue. *Princeton Univ. Press*, 2002.
- [3] J.E. Laird. Research in human-level AI using computer games. *Commun. ACM*, Vol. 3(8), Pág 32-35. 2002.
- [4] D. Loiacono, J. Togelius, P.L. Lanzi, L. Kinnaird-Hetter, S.M. Lucas, M. Simmerson, D. Pérez, R.G. Reynolds, Y. Sáez. The WCCI 2008 Simulated Car Racing Competition. *Proceedings of the IEEE Symposium on Computational Intelligence and Games*. Pág. 119-126. 2008.
- [5] S. M. Lucas. Computational Intelligence and AI in Games: a New IEEE Transactions. *IEEE Trans. on Computational Intelligence and AI in Games*. Vol 1 (1). 2009.
- [6] E. H. Mamdani, S. Assilian. Applications of fuzzy algorithms for control of simple dynamic plant, *Proc. IEEE*, vol. 121, Pág. 1585-1588. 1974.
- [7] M.C. Marques, R. Neves-Silva. Traffic simulation for intelligent transportation systems development. *Proceedings 2005 IEEE Intelligent Transportation Systems*. Pág. 320-325. 2005.
- [8] E. Onieva, D. A. Pelta, J. Alonso, V. Milanés, J. Pérez. A Modular Parametric Architecture for the TORCS Racing Engine. *Proc. IEEE Symposium on Computational Intelligence and Games*, Pág. 256-262. 2009.
- [9] L. Qi. Research on Intelligent Transportation System Technologies and Applications. *2008 Workshop on Power Electronics and Intelligent Transportation System*. Pág. 529-531. 2008.
- [10] J. Schaeffer, N. Burch, A.K.Y Björnsson, M. Mueller, R. Lake, P. Lu, S. Sutphen. Checkers is solved, *Science*, vol. 317, Pág. 1518-1522. 2007.
- [11] M. Sugeno, On Stability of Fuzzy Systems Expressed by Fuzzy Rules with Singletons Consequents. *IEEE Trans. Fuzzy Systems*. Vol. 7(2), Pág 201-224. 1999.
- [12] J. Togelius, S. Lucas, H. Duc Thang, J.M. Garibaldi, T. Nakashima, C. Hiong Tan, I. Ihanany, S. Berant, P. Hingston, R.M. MacCallum, T. Haferlach, A. Gowrisankar, P. Burrow. The IEEE CEC Simulated Car Racing Competition. *Genetic Programming and Evolvable Machines*, vol 9, Pág. 295-329, 2008.
- [13] G. Yannakakis and J. Hallam, "Evolving opponents for interesting interactive computer games," From Animals to Animats, vol. 8, pp. 499-508, 2004.

Sistema basado en reglas difusas para el mapeo de ontologías

Susel Fernández

Juan R. Velasco

Miguel A. López-Carmona

Departamento de Automática, Universidad de Alcalá
 Edificio Politécnico, Ctra. N-II Km. 31,600
 28871 Alcalá de Henares, Madrid, España
 {susel,juanra,miguellop}@aut.uah.es

Resumen

Las ontologías son una herramienta crucial para especificar formalmente el vocabulario y la relación de conceptos en las plataformas de agentes, por lo que, para compartir información, agentes que utilizan diferentes vocabularios deben poder traducir los datos de un marco ontológico a otro. El tratamiento de la incertidumbre juega un papel fundamental en el mapeo de ontologías, puesto que el grado de solapamiento entre conceptos es difícil de representar lógicamente, y por lo general las aplicaciones no cuentan con la suficiente información al respecto. Este artículo está enfocado a ofrecer mecanismos de ayuda a los expertos en la primera fase del mapeo de ontologías, utilizando técnicas de lógica difusa para determinar el grado de similitud entre conceptos de ontologías diferentes. Para cada par de conceptos se calculan dos tipos de similitud: la primera utilizando el coeficiente de Jaccard sobre muestras de documentos relevantes tomados de la web, y la segunda basada en la relación lingüística de los conceptos. Finalmente se calcula la similitud total a través de un sistema basado en reglas difusas. Las ideas presentadas en este trabajo fueron validadas utilizando varias ontologías del mundo real.

Palabras Clave: Mapeo de ontologías, sistema basado en reglas difusas, similitud, conceptos.

1 INTRODUCCIÓN Y ANTECEDENTES

La investigación sobre los servicios de web semántica promete una mayor interoperabilidad entre agentes software y servicios web, utilizando ontologías compartidas publicadas en la web semántica. Sin embargo, los servicios producidos y descritos por diferentes desarrolladores pueden utilizar diferentes o quizás parcialmente solapados conjuntos de ontologías, por lo que, para compartir información, agentes que

utilizan diferentes vocabularios deben poder traducir los datos de un marco ontológico a otro.

El mapeo de ontologías no es más que el proceso de encontrar correspondencias entre los conceptos de dos ontologías a través de reglas. Este proceso es necesario para el intercambio de información y servicios dentro de una plataforma de agentes y puede ser un valioso paso en la solución al problema de la traducción de ontologías.

La incertidumbre está relacionada con todos los aspectos de la web semántica, puesto que una descripción de un concepto desconocido o un objeto puede ser incierta, y a menudo se da el caso de que un concepto definido en una ontología empareja solo parcialmente con uno o más conceptos en otra. En muchas aplicaciones el solapamiento entre conceptos y clases no puede ser representado lógicamente, lo que dificulta el proceso de mapeo.

Existen algunos trabajos anteriores encaminados al mapeo de ontologías, que han realizado aportes interesantes. Noy y Musen han desarrollado las herramientas SMART [15], PROMPT [16], y PROMPTDIFF [17], que usan emparejamiento de similitud lingüística entre los conceptos para iniciar un proceso de fusión o alineación y, entonces, utilizan las estructuras ontológicas para proporcionar un conjunto de heurísticos para identificar más coincidencias entre las ontologías.

McGuinness y colegas desarrollaron una herramienta interactiva similar, Chimaera [13], donde el ingeniero es el encargado de tomar las decisiones que afectarán el proceso de fusión. Chimaera analiza las ontologías que se van a fusionar, y si encuentra coincidencias lingüísticas, la fusión se realiza automáticamente. De lo contrario el usuario es avisado para que tome nuevas medidas. Es bastante similar a PROMPT en cuanto a que ambos están incrustados en los entornos de edición de la ontología, pero difieren en las sugerencias que hacen a sus usuarios con respecto a los pasos de fusión.

Otras investigaciones han seguido enfoques probabilísticos. Por ejemplo, Doan y colegas desarrollaron el sistema GLUE [6], que emplea técnicas de aprendizaje automático para encontrar mapas entre ontologías. Dadas dos ontologías, para cada concepto en una ontología, GLUE encuentra la mayoría de los conceptos similares en la otra ontología utilizando definiciones probabilísticas de varias medidas prácticas de

similitud. GLUE calcula la distribución conjunta de los conceptos, y permite a la aplicación la utilización de esta distribución para calcular cualquier medida adecuada de similitud. Sin embargo este enfoque no tiene en cuenta por completo la incertidumbre en el mapeo, ya que se ignoran conceptos que son similares pero en un menor grado.

Ron Pang y colegas desarrollaron un marco probabilístico de mapeo automático de ontologías basado en Redes Bayesianas (BNs) [18]. En este marco, las ontologías son primero traducidas a Redes Bayesianas, y a continuación el mapeo de conceptos se realiza como razonamiento evidencial entre las dos BNs. Las probabilidades necesarias tanto en la traducción como en el mapeo se obtienen mediante el uso de programas de clasificación de texto asociando distintos conceptos con documentos de texto relevantes recuperados de la web. Este enfoque solo tiene en cuenta la probabilidad de aparición de los conceptos en la web y no su similitud lingüística, por lo que falla en el caso en que dos conceptos muy similares no tengan el mismo índice de popularidad en la web.

Nuestra propuesta se centra en la primera etapa del mapeo de ontologías, hallando similitudes entre conceptos. Primeramente se calcula una similitud entre los conceptos utilizando el coeficiente de Jaccard, a partir de documentos relevantes recuperados de la web, específicamente de la Wikipedia [24]. Luego se calcula la similitud lingüística, y posteriormente se obtiene la similitud total utilizando un sistema basado en reglas difusas. La organización del trabajo es la siguiente: en la sección 2 se describe el uso del coeficiente de Jaccard y el método utilizado para obtener la similitud lingüística. En la sección 3 se muestra el sistema basado en reglas difusas, y en la sección 4 se reflejan los resultados de aplicar nuestro método en dos ontologías del mundo real. Finalmente en la sección 5 se presentan las conclusiones y las líneas de trabajo futuras.

2 CALCULO DE SIMILITUDES ENTRE CONCEPTOS

2.1. COEFICIENTE DE JACCARD

El coeficiente de Jaccard [21] es uno de los índices binarios de similitud más conocidos y utilizados. Se define como el tamaño de la intersección dividido entre el tamaño de la unión entre dos conjuntos de datos y su valor está en $[0, 1]$.

Dada la siguiente tabla de contingencia con 2 variables binarias,

Tabla 1 Parámetros para calcular el coeficiente de Jaccard

i / j	1	0	Total
1	a	b	$a+b$
0	c	d	$c+d$
Total	$a+c$	$b+d$	$a+b+c+d$

Donde:

- a es el numero de veces que ambas observaciones tienen valor 1.
- b es el numero de veces que la observación i tiene valor 1 y la observación j tiene valor 0.
- c es el numero de veces que la observación i tiene valor 0 y la observación j tiene valor 1.
- d es el número de veces que ambas observaciones tienen valor 0.

El coeficiente de similitud de Jaccard se calcula a través de la siguiente fórmula:

$$s_{ij} = \frac{a}{a+b+c} \quad (1)$$

En nuestro trabajo hemos aplicado el coeficiente de Jaccard para calcular la similitud entre conceptos de dos ontologías diferentes. Los valores que intervienen en esta fórmula fueron obtenidos a través de búsquedas sucesivas de documentos relevantes a cada concepto en la web. De manera similar a la desarrollada en [18], para asegurar que la búsqueda de documentos de la web solo devolviera aquellos que fueran relevantes al concepto, la consulta de búsqueda se formó mediante la concatenación de todos los conceptos que se encontraban en el camino desde la raíz hasta el nodo actual en la taxonomía.

Los ejemplares que contenían el concepto A serían obtenidos mediante la búsqueda de A y todos sus ancestros en la taxonomía, mientras que los ejemplares que no contenían a A serían aquellos en los que estuvieran presentes todos sus ancestros en la taxonomía, y no estuviera presente el concepto A. Para cada par de conceptos A y B, pertenecientes a las ontologías origen y destino respectivamente se contaron (a) los documentos en los que aparecían los dos conceptos, (b) los documentos en los que aparecía A y no aparecía B, (c) los documentos en los que aparecía B y no aparecía A y (d) los documentos en los que no aparecía ninguno de los dos conceptos. Una vez obtenidos estos valores, para cada par de conceptos de las ontologías origen y destino respectivamente, fue calculada su similitud de Jaccard por la fórmula (1).

Para la búsqueda de los documentos en la web hemos utilizado el API AJAX de Google [10]. En trabajos previos realizábamos la búsqueda en toda la web, pero se obtenían muchos documentos irrelevantes, por lo que en este trabajo restringimos el dominio de búsqueda al sitio de la Wikipedia [24].

2.2. SIMILITUD LINGÜÍSTICA

La similitud lingüística constituye el indicador más fuerte del parecido entre dos conceptos, debido a que por lo general los desarrolladores de ontologías dentro de un mismo dominio emplean términos relacionados lingüísticamente para expresar conceptos equivalentes.

Dados los conceptos A y B de dos ontologías diferentes, el primer paso de nuestra propuesta para calcular la similitud lingüística consiste en obtener las listas de sinónimos asociadas a cada uno de ellos, auxiliándonos de la herramienta WordNet [22]. Posteriormente, a cada una de las palabras de las listas de sinónimos obtenidas se les aplica el algoritmo de stemming de Porter [19], que no es más que el proceso de eliminar los finales morfológicos de las palabras en inglés, para obtener las raíces morfológicas de las mismas. Si llamamos L_A y L_B a las listas de raíces de los sinónimos de cada concepto obtenidas en esta etapa, podemos calcular la similitud lingüística entre los conceptos como sigue:

$$S = \min \left[\frac{c_A}{T_A}, \frac{c_B}{T_B} \right] \quad (2)$$

Donde c_A es la cantidad de palabras de la lista L_A que están en la lista L_B , T_A es la cantidad total de palabras de la lista L_A , c_B es la cantidad de palabras de la lista L_B que están en la lista L_A , y T_B es la cantidad total de palabras de la lista L_B .

Por ejemplo, si tenemos los conceptos “*Intelligent agents*” y “*agents*”. Después de ejecutar los primeros pasos del algoritmo obtendríamos como resultado la tabla 2.

Tabla 2 Valores obtenidos en los pasos iniciales del cálculo de la similitud lingüística

Conceptos	Lista de sinónimos del concepto	Aplicando el algoritmo de stemming
Intelligent Agents	intelligent, agile, alert, clever, smart, brilliant, agent, factor, broker, functionary	L_A =Intellig, agil, alert, clever, smart, brilliant, <u>agent</u> , <u>factor</u> , <u>broker</u> , <u>functionari</u>
Agents	agent, factor, broker, functionary	L_B = <u>agent</u> , <u>factor</u> , <u>broker</u> , <u>functionari</u>

Por tanto la similitud lingüística sería 0.4.

3 SISTEMA BASADO EN REGLAS DIFUSAS

La lógica difusa en la actualidad es un campo de investigación muy importante tanto por sus aplicaciones matemáticas y teóricas como por sus aplicaciones prácticas.

Los sistemas basados en reglas difusas constituyen una extensión de los sistemas basados en reglas clásicos debido a que se componen de reglas “*IF-THEN*” cuyos antecedentes y consecuentes están compuestos por instrucciones de lógica difusa en lugar de instrucciones de lógica clásica. Estos sistemas son usados como herramientas para representar diferentes formas de conocimiento de un problema, así como para modelar las interacciones y relaciones entre sus variables. Debido a esta propiedad, han sido exitosamente aplicados a un amplio rango de problemas en diferentes dominios con incertidumbre y conocimiento incompleto [4].

Los sistemas basados en reglas difusas se componen de una base de conocimientos, un motor de inferencias, una interfaz de fuzzificación en la entrada, y una interfaz de defuzzificación en la salida. Figura 1, tomada de [4].

La base de conocimientos está formada principalmente por una base de datos, que contiene todos los datos relativos al sistema, como variables y valores posibles, y una base de reglas, que contiene todas las proposiciones que van a regir el sistema. El motor de inferencias es el “supervisor”, y se encarga de extraer las conclusiones partiendo de los datos simbólicos que le han llegado, aplicando las reglas que rigen el sistema difuso en el cuál trabaja. Por último las interfaces de fuzzificación y defuzzificación tienen la función de convertir una entrada natural en un valor difuso (fuzzificación), y una salida difusa en un valor natural (defuzzificación).

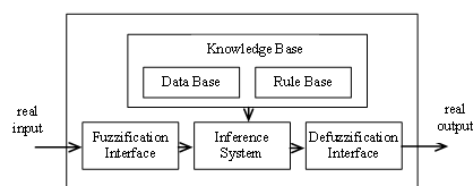


Figura 1. Estructura básica de un sistema basado en reglas difusas.

En este trabajo hemos utilizado el entorno de desarrollo XFuzzy 3.0 [23], que integra varias herramientas para cubrir las diferentes etapas del diseño de sistemas difusos.

3.1. VARIABLES DEL SISTEMA

En el sistema se han definido dos variables de entrada y una de salida. A cada una de ellas hemos asociado un conjunto de términos lingüísticos cuya semántica ha sido representada por medio de funciones triangulares de pertenencia. Estas variables son las siguientes:

a) **Sim_Jaccard**: Es una variable de entrada y representa el valor de la similitud calculado a través de coeficiente de Jaccard. Tiene asociado el siguiente conjunto de términos lingüísticos: $D_{jacc} = \{Baja, Regular, Media, Alta, MuyAlta\}$.

Para definir las funciones de pertenencia primeramente fue necesario dividir los datos de la similitud de Jaccard en grupos, por lo que decidimos utilizar los cuartiles del conjunto de similitudes para acotar los triángulos de pertenencia de la siguiente manera:

Baja: (-0.00324149, 0, 0.00324149)
Regular: (0, 0.00324149, 0.0831758)
Media: (0.00324149, 0.0831758, 0.20338982)
Alta: (0.0831758, 0.20338982, 1)
MuyAlta: (0.20338982, 1, 0.20338982)

Las funciones de pertenencia de la variable *Sim_Jaccard* se muestran gráficamente en la figura 2.

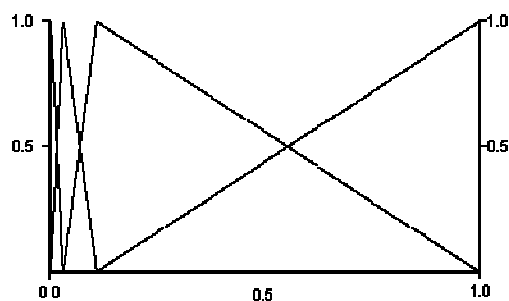


Figura 2. Funciones triangulares de pertenencia de la variable *Sim_Jaccard*.

b) **Sim_Ling**: Es una variable de entrada que representa el valor de la similitud lingüística. Tiene asociado el siguiente conjunto de términos lingüísticos: $D_{ling} = \{Baja, Regular, Media, Alta, MuyAlta\}$. Sus funciones de pertenencia se muestran en la figura 3.

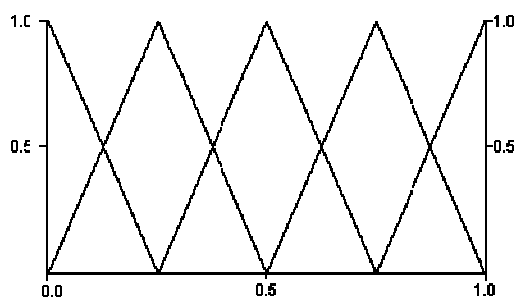


Figura 3. Funciones triangulares de pertenencia de la variable *Sim_Ling*.

c) **Similitud**: Es la variable de salida y representa el valor de la similitud total, obtenida por el sistema difuso. Tiene asociado el siguiente conjunto de términos lingüísticos: $D_{Similitud} = \{MuyBaja, Baja, MedioBaja, Regular, MedioAlta, Alta, MuyAlta\}$. Sus funciones de pertenencia se muestran en la figura 4.

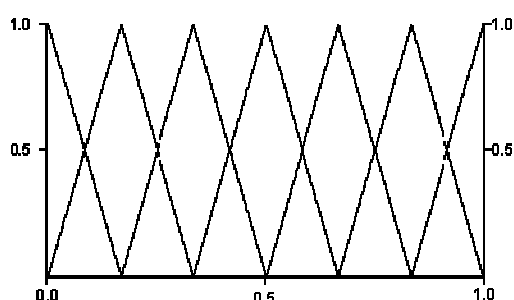


Figura 4. Funciones triangulares de pertenencia de la variable *Similitud*.

3.2 BASE DE REGLAS

La base de reglas que hemos definido en el sistema difuso ha sido formada mediante heurísticos de los expertos y se muestra en la Tabla 3.

Tabla 3. Base de reglas

Ling/ Jacc	Baja	Regular	Media	Alta	Muy Alta
Baja	Muy Baja	Baja	Medio Baja	Regular	Medio Alta
Regular	Baja	Medio Baja	Regular	Medio Alta	Alta
Media	Baja	Medio Baja	Regular	Medio Alta	Alta
Alta	Medio Baja	Regular	Medio Alta	Alta	Muy Alta
Muy Alta	Regular	Medio Alta	Alta	Alta	Muy Alta

4 EXPERIMENTOS

Hemos realizado varios experimentos, utilizando diversas ontologías, con resultados satisfactorios. A modo de ejemplo, describimos un experimento en el que se utilizan dos ontologías del mundo real extraídas de la Web. Estas son, la ontología ACM [1], y la ontología DMOZ [5]. Nuestro objetivo ha sido demostrar la eficacia de esta propuesta para la obtención de un índice de similitud válido entre conceptos, que ayude al proceso de mapeo de ontologías. De manera similar a [18], hemos seleccionado el tópico de Inteligencia Artificial, haciendo una poda de conceptos en ambas ontologías de forma tal que exista cierto grado de solapamiento entre ambas, y se pueda comprobar la efectividad del método. La figura 5 muestra los conceptos seleccionados de ambas ontologías, así como las relaciones evidentes entre ellos. Para cada par de conceptos A y B, pertenecientes a las ontologías ACM y DMOZ respectivamente, hemos calculado primero la similitud de Jaccard, utilizando los conjuntos de documentos relevantes recuperados de la Web. Como hemos mencionado anteriormente esta búsqueda está restringida a la Wikipedia. Posteriormente se calcula la similitud lingüística, como se describe en secciones anteriores.

Si consideramos como n la cantidad de conceptos de la primera ontología, y m la cantidad de conceptos de la segunda, el proceso se completaría en $n \times m$ iteraciones. La tabla 4 muestra los conceptos para los cuáles hemos obtenido índices de similitud más elevados. La tabla 5 muestra los conceptos para los cuales la similitud ha resultado ser más alta de lo debido.

En el trabajo de [18], utilizando fragmentos similares de las ontologías ACM y DMOZ, plantean que según su método se obtiene un índice de similitud de 0.61 para los conceptos “*connectionism and neural nets*” y “*Neural Networks*”, debido a que solo tienen en cuenta la probabilidad de aparición de ambos conceptos en la web, y el término “*connectionism*” es muy poco popular. Como se puede observar en la figura 4, aplicando nuestro método, hemos obtenido un grado de similitud más alto para estos dos conceptos, debido fundamentalmente a su relación lingüística.

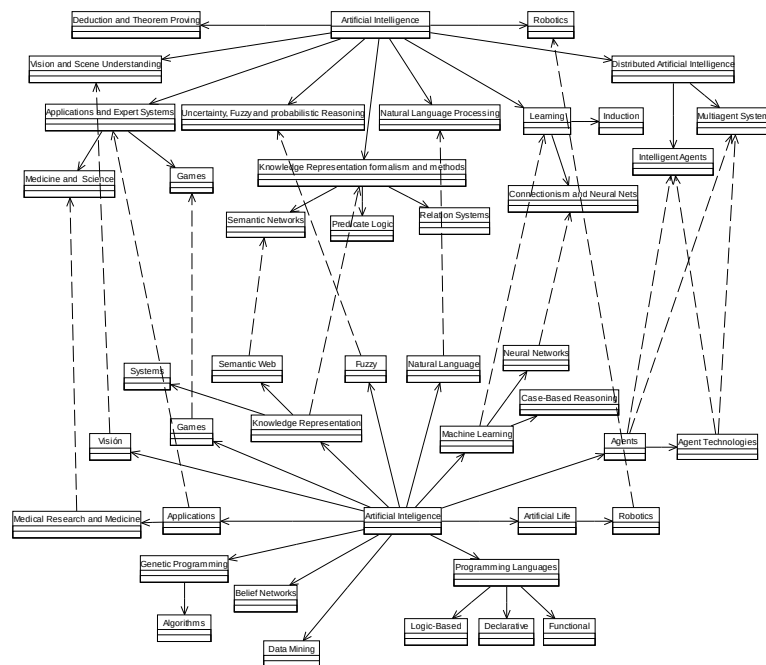


Figura 5. Conceptos de las ontologías ACM y DMOZ.

Tabla 4. Parejas de conceptos con similitud más alta.

Conceptos ACM	Conceptos DMOZ	Similitud
Robotics	Robotics	0.99
Games	Games	0.99
Predicate Logic	Logic-Based	0.91
Intelligent Agents	Agents	0.91
Natural Language Processing	Natural Language	0.91
Multiagent Systems	Systems	0.87
Learning	Machine learning	0.87
Multiagent Systems	Agent Technologies	0.85
Intelligent Agents	Agent Technologies	0.74
Multiagent Systems	Agents	0.72
Knowledge Representation Formalisms and Methods	Knowledge Representation	0.71
Connectionism and neural nets	Neural Networks	0.71
Applications and Expert Systems	Applications	0.71
Semantic Networks	Semantic Web	0.71
Relation Systems	Systems	0.58
Medicine and Science	Medical Research and Medicine	0.58
Natural Language Processing	Programming Language	0.55
Vision and Scene Understanding	Vision	0.55
Uncertainty, fuzzy, and probabilistic reasoning	Fuzzy	0.55
Semantic Networks	Neural Networks	0.55

Tabla 5. Parejas de conceptos con similitud más alta de lo esperado.

Conceptos ACM	Conceptos DMOZ	Similitud
Predicate Logic	Logic-Based	0.91
Multiagent Systems	Systems	0.87
Semantic Networks	Neural Networks	0.55
Natural Language Processing	Programming Language	0.55

Finalmente, hemos realizado una prueba estadística χ^2 para comprobar si los resultados obtenidos se corresponden con los esperados para estas dos ontologías. Para ello planteamos la hipótesis H_0 que consiste en suponer que no hay diferencia entre estos valores. Calculando con 1 grado de libertad, hemos comprobado que se cumple la hipótesis H_0 con un 95% de confianza, por lo que se acepta la suposición y se demuestra la efectividad del método propuesto. La tabla 6 muestra los valores de contingencia para el experimento.

Tabla 6. Tabla de contingencia para la prueba χ^2

	Baja	Alta	Total
Esperados	416	16	432
Observados	412	20	432

5 CONCLUSIONES Y TRABAJO FUTURO

En este artículo se describe nuestro trabajo encaminado a ofrecer un método que ayude a los expertos en la fase inicial del mapeo de ontologías. Nuestra propuesta consiste en un sistema difuso para obtener similitudes

entre conceptos de diferentes ontologías. Este sistema utiliza como entrada valores obtenidos aplicando el coeficiente de similitud de Jaccard a documentos recuperados de la Wikipedia, y un índice de similitud basado en la relación lingüística de los conceptos, con mucha influencia en el cálculo de la similitud total. Actualmente este proceso se realiza para cada pareja de conceptos de las dos ontologías, y estamos trabajando para reducir el número de iteraciones, haciendo que el sistema tenga en cuenta otros factores como las similitudes entre conceptos hermanos, hijos y padres en la taxonomía de cada ontología. Otro de nuestros objetivos es refinar el cálculo de la similitud lingüística incluyendo otro tipo de relaciones lingüísticas entre los conceptos, y la aplicación de reglas difusas para calcular la similitud lingüística global. También como trabajo futuro nos hemos propuesto incluir las propiedades de los conceptos en el proceso de mapeo, así como la escalabilidad de la aplicación.

Agradecimientos

Este trabajo es parte del proyecto T2C2, financiado por el Ministerio de Educación y Ciencia, TIN2008-06739-C04-04, y el proyecto europeo ITEA-2 No 2008005, "Do-it-Yourself Smart Experiences", financiado por el Ministerio de Industria, Turismo y Comercio de España.

Referencias

- [1] ACM Topic, <http://www.acm.org/about/class/1998/>
- [2] Barwise, J., Seligman, J.: *Information Flow: The Logic of Distributed Systems*. Cambridge University Press (1997).
- [3] Calvanese, D., Giacomo, G., Lenzerini, M.: *Ontology of integration and integration of ontologies*. In: *Description Logic Workshop (DL 2001)*, pp 10–19 (2001).
- [4] Cordon, O., Herrera, F., Hoffman, F., Magdalena, L.: *Genetic Fuzzy Systems. Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. World Scientific, Singapore (2001).
- [5] DMOZ hierarchie, <http://www.dmoz.org>
- [6] Doan A., Madhavan, J., Domingos, P., Halevy, A.: *Ontology Matching: A Machine Learning Approach*. *Handbook on Ontologies in Information Systems*. In: S. Staab and R. Studer (eds.), *Invited paper*. Pp. 397–416. Springer-Verlag, (2004).
- [7] Fernández-Breis, J., Martínez-Béjar, R.: *A cooperative framework for integrating ontologies*. In: *International Journal of Human-Computer Studies*, 56: pp. 665–720, (2002).
- [8] Gruber, T., Olsen, G.: *An ontology for engineering mathematics*. In: J. Doyle, P. Torasso, E. Sandewall, (eds). *Fourth International Conference on Principles of Knowledge Representation and Reasoning*, pp. 258–269, San Mateo, CA, USA (1994).
- [9] Grüninger, M.: *Ontologies for translation: Notes for refugees from Babel*. EIL Technical Report, Enterprise Integration Laboratory (EIL), University of Toronto, Canada (November 1997).
- [10] Google AJAX API, <http://code.google.com/intl/es/apis/ajaxsearch>
- [11] Jannink, J., Pichai, S., Verheijen, D., Wiederhold, G.: *Encapsulation and Composition of Ontologies*. In: *AAAI'98 Workshop on Information Integration*, Madison, WI, USA, pp. 43–51 (July 1998).
- [12] Kalfoglou, Y., Schorlemmer, M.: *Ontology mapping: the state of the art*. *The Knowledge Engineering Review*, 18(1) pp. 1–31, (2003).
- [13] McGuinness, D., Fikes, R., Rice, J., Wilder, S.: *An Environment for Merging and Testing Large Ontologies*. In: *17th International Conference on Principles of Knowledge Representation and Reasoning (KR-2000)*, Colorado, USA, pp. 483–493, (April 2000).
- [14] Mitra, P., Noy, N. F., Jaiswal, A. R. 2004. OMEN: A Probabilistic Ontology Mapping Tool. In: *Workshop on Meaning Coordination and Negotiation at the Third International Conference on the Semantic Web (ISWC-2004)*. Hiroshima, Japan, pp. 71–83, (2004).
- [15] Noy, N. F., Musen, M. A.: *SMART: Automated Support for Ontology Merging and Alignment*. In: *12th Workshop on Knowledge Acquisition, Modelling and Management (KAW'99)*, Banff, Canada, pp. 1–20, (October 1999).
- [16] Noy, N. F., Musen, M. A.: *The PROMPT suite: Interactive tools for ontology merging and mapping*. *International Journal of Human-Computer Studies*, 59(6), pp. 983–1024 (2003).
- [17] Noy, N. F., Musen, M. A.: *PROMPTDIFF: A Fixed-Point Algorithm for Comparing Ontology Versions*. In: *18th National Conference on Artificial Intelligence (AAAI'02)*, pp. 744–750, Edmonton, Alberta, Canada, (August 2002).
- [18] Pan, R., Ding, Z., Yu, Y., Peng, Y.: *A Bayesian Network Approach to Ontology Mapping*. *The Semantic Web – ISWC 2005*, Vol. 3729/2005, pp. 563–577. Springer Berlin / Heidelberg (October 2005).
- [19] Porter Stemming algorithm, <http://tartarus.org/~martin/PorterStemmer>
- [20] Quesada, V., Isidoro, A., López, L.: *Curso y Ejercicios de Estadística*. Alhambra Longman, Madrid (1994).
- [21] Rijsbergen, V., C. J.: *Information Retrieval*. Butterworths. Second Edition, London (1979).
- [22] Wordnet, <http://wordnet.princeton.edu>
- [23] XFuzzy 3.0 framework, http://www.imse.cnm.es/Xfuzzy/Xfuzzy_3.0/tools/xfuzzy_sp.html
- [24] Wikipedia, <http://www.wikipedia.org>

EVALUATING THE IMPACT OF SEWAGE SLUDGE APPLICATION ON AGRICULTURAL SOILS

Josep Pijuan¹

Aida Valls¹

Ana Pasuello²

Marta Schuhmacher²

¹ Department of Computer Science and Mathematics, Universitat Rovira i Virgili, Tarragona, Spain
{josep.pijuan, aida.valls}@urv.cat

² Department of Chemical Engineering, Universitat Rovira i Virgili, Tarragona, Spain
{ana.passuello, marta.schuhmacher}@urv.cat

Abstract

Sewage sludge application on agricultural soils is recommended by governments in order to recycle nutrients and organic matter. However, this practice may lead to ecological and human risks. In this paper, a decision support system for helping in the management of sewage sludge is presented. The system integrates different methodologies. In the first stage, fuzzy expert systems are used to assess the partial utility of the alternatives with respect to some complex criteria. This is combined with the use of classical piecewise linear functions for simple criteria. In the second stage, utilities are aggregated with conjunctive/disjunctive operators. The results obtained in a case study with real data are presented.

Keywords: Fuzzy expert systems, Aggregation operators, Decision Making, Environmental applications, Sewage sludge management.

developed a sophisticated decision support system that integrates the experts' knowledge about the impact on the human health and the environment, in relation to the different characteristics of the sewage sludge and the soil. This system is expected to be used by the manager that decides the best place to send the sewage sludge of each of the different WWTPs in a region [6]. From a technological point of view, the system developed is particularly interesting because the solution proposed integrates fuzzy expert systems into a traditional multi-attribute aggregation model.

The paper is organized as follows. Section 2 introduces the problem of management of sewage sludge (i.e. biosolids). Section 3 presents the decision support system designed to evaluate the suitability of applying some sludge into a certain agricultural soil. Section 4 shows the first part of the process, consisting in evaluating partial utilities with respect to low level criteria. Then, section 5 is devoted to present an aggregation model of the partial utilities in order to obtain the global utility. Section 6 shows a real case study and the results obtained. Finally, some conclusions and future work is outlined in section 7.

1 INTRODUCTION

This paper addresses the problem of recycling sewage sludge produced at wastewater treatment plants (WWTP). This sludge can be used as fertilizer in agricultural soils, helping to the sustainability of the water life cycle. Although this process has many advantages, it must be carefully decided where to send each type of sludge, because sludge contains contaminants (metals and organic compounds) that may impact humans and ecosystems. For this reason, it is interesting to have tools to evaluate the suitability of applying a certain sewage sludge on a particular soil.

Nowadays, there is only one regulation indicating the maximum metal content permitted in sewage sludge that is going to be used as fertilizer. In this work we have

2 BIOSOLID MANAGEMENT

Biosolids, also referred as sewage sludge, are residues generated at WWTPs, obtained from solids removal on various parts of the treatment system. In Spain the production of biosolids increased in a 39% from 1997 to 2005 [5]. The legislation requires building new plants in towns with a population higher than 2000 inhabitants, so more than 1000 WWTP have been build in all Spain.

Once treated, sludge can be recycled or disposed using three main routes: disposal on agricultural soil, incineration and landfill. Any of these three scenarios leads to different impacts to humans and ecosystems. In this work we focus on the study of biosolids disposal in agricultural soils. This study is part of some Spanish funded research projects.

Sewage sludge application on soil improves fertility and exempt fertilizers use. Furthermore the transfer of nutrients and organic matter to the soil increases crop production. For these reasons, the Spanish government wants that at least 70% of the WWTP sludge is applied into agricultural soils. In 2005 the recycling of sewage sludge to agriculture represented a 65% of the total disposal of biosolids [5]. However, it should be studied how this percentage may be increased. Some studies have proven that sewage sludge application improves soil fertility along the years [4]. Even so, sewage sludge application on soils may lead to groundwater nitrification as a result of nitrogen movement through lixiviate. The nitrates content of sewage sludge is a variable related to application dose. Special care must be taken on areas that are vulnerable to nitrogen pollution, as recommended on EC Nitrate Directive 91/676.

In addition to these issues, the impacts to humans and ecosystems must be considered, such as the exposition to organic contaminants through different routes (inhalation, ingestion and dermal contact), the lifecycle, and the field properties.

3 ARCHITECTURE OF THE DECISION SUPPORT SYSTEM

To design this decision support system, a Multi-Attribute Utility approach has been taken [3]. This approach considers that the different alternatives that are analysed in the decision process can be evaluated by a set of criteria G . Criteria are built on the basis of the alternatives' properties. Each criterion is a function g_i , where $g_i(a_i)$ is regarded as a partial utility (i.e. preference) from the decision maker point of view. The partial utility value denotes the degree to which the value of some property a_i satisfies a specific requirement of the user. It can also be called a preference or suitability degree.

In this problem, an alternative is the combination of a certain type of biosolid coming from some concrete WWTP into a certain agricultural soil. The number of features describing sewage sludge is 6 and 8 for characterizing the soil and landscape. Additionally 8 properties have been also included in the alternative's description, which are mainly related to economic issues and to the sensible population that might be affected.

Partial utility values of all the criteria are obtained from evaluating the features of the alternative with the corresponding utility function. Then, those partial utilities are aggregated to find a global utility value for each alternative. Afterwards, these ratings are rank ordered.

Finally, parameter uncertainty has been addressed. A sensitivity analysis tool has been implemented to

understand the effects of the parameter variations on the results. Sensitivity analysis can help to explain the results and make the recommendations with confidence [8].

Figure 1 shows the architecture of the decision support system developed. First the input data coming from the analysis of the chemical and physical properties of the sewage sludge and soils are evaluated using the utility functions. Then, the global utility value is calculated using the Logic Scoring of Preference model (LSP) [1]. Finally, the tool ranks the alternatives according to their global utility and performs a sensitivity analysis on the weights given to the different operators of LSP.

For the utility assessment, two different methodologies are used [7]: fuzzy expert systems to evaluate the utility of composite criteria and traditional linear functions for simple criteria. This is explained in more detail in the next section.

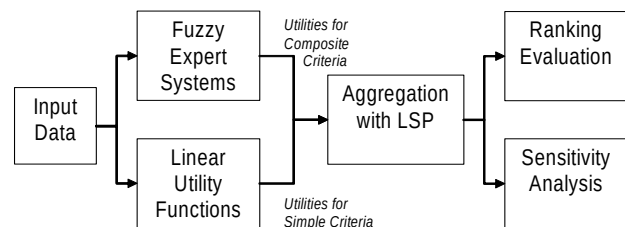


Figure 1: Decision support system architecture.

To organize the set of criteria, a hierarchical structure has been defined together with the experts. In the first level, three main types of criteria are distinguished: economical, environmental and social. Each of these classes of preference criteria represents a different aspect of the problem regarding different points of view. The hierarchy of criteria is shown in Figure 2. Simple and Composite criteria are denoted by (S) and (C), respectively.

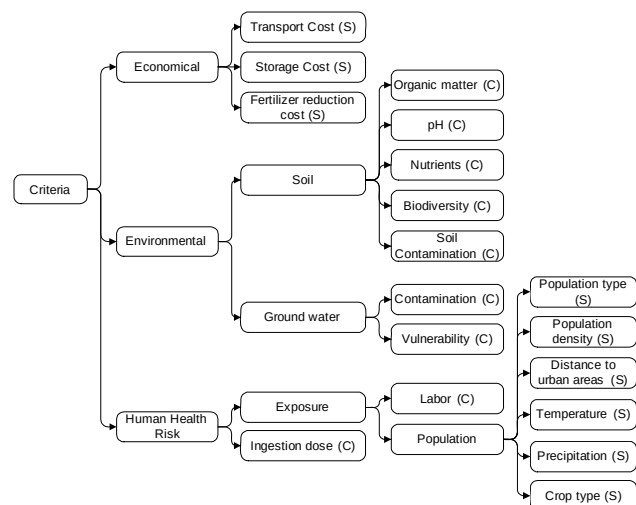


Figure 2: Family of criteria.

4 PREFERENCE ASSESMENT

Partial utility functions are built from the knowledge of the domain experts. Utility functions must give a preference score to each possible value in the reference domain of a certain feature of the alternatives. During this process we found that some of the criteria do not depend on a single feature but on the interactions among combinations of features, usually a combination of soil and sludge properties. Therefore, the characterization of the utility functions in this environmental problem was not straightforward. First of all, the interactions between those features have been studied in order to find which are the groups of variables that should be modelled together to define the preference criteria. Then, two types of criteria have been distinguished, which have been called: Simple criteria, S , and Composite criteria, C . In both cases, a preference scale from 0 to 10 has been selected, for expert's convenience.

Definition 1. Simple criteria $S \subset G$ are criteria of the form $g: \mathcal{H} \rightarrow [0..10]$

Definition 2. Composite criteria $C \subset G$ are criteria of the form $g: \mathcal{H} \times \mathcal{H} \times \dots \times \mathcal{H} \rightarrow [0..10]$

In Figure 2, the two types of criteria are indicated: (S) refers to Simple criteria and (C) corresponds to Composite criteria. For simple criteria, the classical value functions can be applied because a single feature must be considered. However, for composite criteria, the definition of the preference function depends on the combination of soil and sludge properties and, sometimes, also other variables. In this case, fuzzy rule-based systems have been proposed to model the interactions.

4.1. LINEAR UTILITY FUNCTIONS

In the family of criteria there are 9 simple criteria. They are related to the economical and social aspects of the problem. Those criteria do not have dependences with sludge properties, so that they can be directly evaluated from a single feature.

For this type of criteria, classical utility functions have been defined [3]. Each utility function is obtained by a linear interpolation between the utility scores that the experts have given to certain reference values.

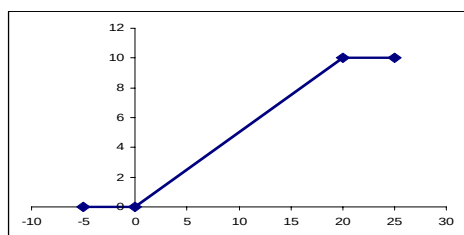


Figure 3: Utility function for the Temperature criterion.

Figure 3 shows the utility function for the Temperature criterion. This function takes as reference the average temperature is represented in Celsius degrees. The utility is an ascending function because high temperatures produce a high degradation of the organic matter and, thus, plant contamination decreases.

4.2. FUZZY EXPERT SYSTEMS

In order to represent the utilities in composite criteria, the use of fuzzy rules has been proposed [7]. Rules permit to model the interactions between the physical and chemical properties of the sewage sludge and the soil. Each rule premise represents a combination of values and the conclusion establishes the corresponding utility value (i.e. measuring the positive impact degree). In addition, this fuzzy approach permits to handle the uncertainty, naturally present in this kind of environmental problem.

In the initial stage, a group of environmental and toxicological experts identified which interactions between features are relevant for each composite criterion. Then, appropriate rules were defined by the experts with the help of knowledge engineers. For each composite criterion a hierarchical fuzzy expert system has been implemented. The construction process is the following one:

1. Define a fuzzy linguistic variable for each numerical feature that interacts with another one in some composite criterion.
2. Define a linguistic variable for giving utility values to the composite criteria. A variable with 11 fuzzy numbers (from 0 to 10) has been defined.
3. Design the rule blocks that are needed in each composite criterion and their interrelations.
4. Define the set of rules for each combination of interacting features. The conclusions of the rules are fuzzy preference values from the domain defined in the previous step.
5. Link the rule blocks according to the structure defined at step 3.

An example of the structure of the fuzzy expert system corresponding to the composite criterion "Groundwater Vulnerability" is shown in Figure 4.

Step 4 is the more difficult and time-consuming one, because the domain experts must be careful in considering all the combinations of values of different soil and sludge properties and give the appropriate utility value for them. However, in step 3, where experts choose the subsets of variables that should be evaluated together, we have been able to identify small groups of variables interacting at the same time. This reduces the complexity of the rule definition and reduces the risk of introducing errors or contradictions in the rules, which is one of the main problems in rule-based systems.

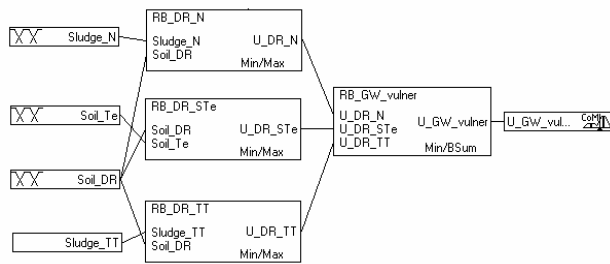


Figure 4: Expert System for Groundwater Vulnerability.

Figure 5 shows an example of the rules that are included in the RB_DR_TT rule block (Figure 4), which evaluates the suitability of an alternative with respect to the sludge treatment type (TT) and the groundwater nitrification test (DR - DRASTIC). The conclusion of the rules is a fuzzy number. DoS stands for Degree of Support of the rule.

IF		THEN	
Sludge_TT	Soil_DR	DoS	U_DR_TT
Mechanical	low	1.00	_7_
Mechanical	medium_low	1.00	_5_
Mechanical	medium_high	1.00	_3_
Mechanical	high	1.00	_1_
Thermal	low	1.00	_9_
Thermal	medium_low	1.00	_7_
Thermal	medium_high	1.00	_5_
Thermal	high	1.00	_3_
Compost	low	1.00	_10_
Compost	medium_low	1.00	_9_
Compost	medium_high	1.00	_7_
Compost	high	1.00	_4_

Figure 5: Fuzzy rules considering the sludge treatment type (TT) with respect to the Soil nitrification (DR).

5 PREFERENCE AGGREGATION

Once the impact on a soil has been evaluated with respect to a particular sludge, the numerical utility evaluations obtained for each simple and composite criterion have to be aggregated to calculate the global preference score of this alternative.

Because the criteria have been organized in a hierarchical classification with very clear groups regarding to different aspects of the problem, it is needed to select an aggregation method that permits to take into account this classification of criteria. After considering different aggregation approaches (Torra and Narukawa, 2005), we propose to apply the LSP method (Dujmovic and Hagashima, 2006). LSP (Logical Scoring of Preferences)

uses generalized conjunction/disjunction operators for combining the values taking into consideration the different levels in the hierarchy of criteria and the user weights and constraints over those criteria.

LSP method consists in logically aggregating the values of small sets of attributes in a compositional way. Thus, the result of the aggregation at the lower level criteria is aggregated again, in the next higher level, using the same type of logical operators. Those aggregation levels are defined by the structure of the hierarchy of criteria. In addition, the LSP aggregation method also permits to give adjustable levels of relative importance to criteria.

This family of operators permits to model situations in which some subsets of criteria must be fulfilled simultaneously while other ones are optional (replaceable). For modelling simultaneity, conjunctive operators must be selected, because they are representing the andness concept, whereas, for replaceability (orness) some kind of disjunctive operator must be used.

In the problem of sludge disposal management, experts agreed in the need of distinguishing between conjunctive and disjunctive groups of criteria. This is the reason why it is convenient to make the evaluation using the LSP method.

Although the level of simultaneity/replaceability can be adjusted to any degree, we have simplified the scenario considering only two degrees of conjunction, two degrees of disjunction and the average operator (which compensates both). As explained in [2], the parametric Weighted Power Mean is the best function to implement this aggregation model (eq. 1). The parameter r permits to adjust the degree of andness/orness. In addition, different weights can be given to each input, subject to adding up to 1.

$$U(a) = \left(w_1 (g_1(a))^r + w_2 (g_2(a))^r + \dots + w_n (g_n(a))^r \right)^{\frac{1}{r}} \quad (1)$$

Table 1 has an example of the results with two input criteria g_1 and g_2 and the different operators (r values) considered in this work. The weights have been established in 0.5 for both. It can be seen that when $r=1$ the average between the two values (0.4 and 0.9) is obtained. The higher the value of r the more disjunctive is the operator, giving more importance to the fulfilment of at least one input. Whereas, if the smaller the r value, the more conjunctive is the operator, penalizing the low values in the inputs.

Table 1: Example of LSP aggregation

		D++	DA	A	CA	C++
$g_1(a)$	$g_2(a)$	$r=20$	$r=4$	$r=1$	$r=-0.7$	$r=-9$
0.4	0.9	0.87	0.76	0.65	0.57	0.43

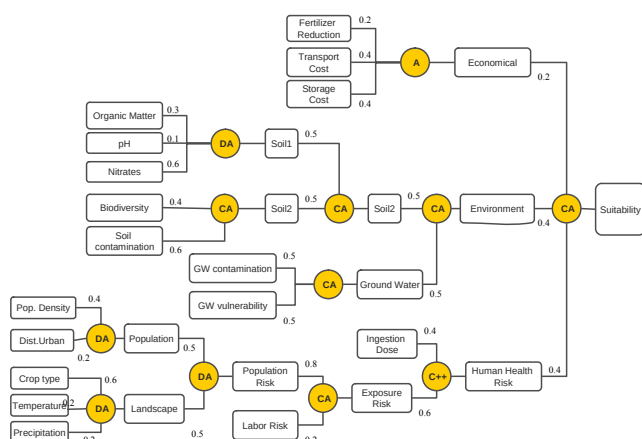


Figure 6: Aggregation model.

Following the hierarchical structure of the family of criteria (Figure 2), the LSP method has been used to aggregate the values according to this structure. At each aggregation point the most suitable aggregation operator has been selected, according with the knowledge of the environmental experts in the project. Notice that both conjunctive and disjunctive operators have been chosen. Experts have also defined the weights of each input. The aggregation model is represented in Figure 6.

At the execution step, before applying the aggregation function (Eq. 1), all the partial utility values are put into the $[0..1]$ interval.

6 TESTING AND ANALYSIS

No previous data sets exist for this problem and this has been an additional handicap for this study. The domain experts participating in this project have work hard to prepare a case study with real data. At the moment, 4 samples of different types of sludge coming from different Catalan WWTP have been collected. These plants receive waters coming from industrial and/or residential areas. In addition, we have 6 samples of different types of agricultural soils in Catalonia. The sludge and soil samples collected have been analyzed in different laboratories (toxicology, chemistry, etc.) in order to extract the values of the properties required. The soils sampled cover a wide range of situations, such as acid and alkaline soils, different textures and different levels of organic matter. Thus, the case study includes 24 alternatives (denoted as T#) that are used to study the performance of the decision support system developed.

Table 2 shows the position of each alternative in the partial rankings in each of the intermediate nodes of the model (see Figure 6: 1 is the best position, 24 the worst one). Each column corresponds to one node, being 'a' the right most one (the global utility or suitability), and continuing from right to left and up to down. Although

there are changes in the positions depending on the node evaluated, few alternatives are at the best positions for all the partial aggregations (f.i. T5, T23), while others are always placed at the worst positions (f.i. T12, T16, T18).

Table 2 : Ranking position at each aggregation point

	a	b	c	d	e	f	g	h	i	j	k	l
T1	17	24	5	19	5	6	22	9	3	8	4	5
T2	7	24	2	11	1	2	18	21	1	20	2	3
T3	12	13	3	18	4	5	23	17	1	16	2	3
T4	10	13	11	9	9	10	17	17	7	16	8	9
T5	9	13	1	17	2	3	21	5	2	4	3	4
T6	18	24	12	21	10	11	24	25	7	24	8	9
T7	19	21	19	15	17	18	14	9	16	8	17	18
T8	6	21	9	5	6	7	9	21	8	20	9	10
T9	8	9	10	9	8	9	17	17	5	16	6	7
T10	20	9	22	16	21	22	7	17	21	16	22	23
T11	5	9	6	7	3	4	12	5	4	4	5	6
T12	21	21	21	20	20	21	20	25	21	24	22	23
T13	22	10	20	23	22	23	14	9	19	8	20	21
T14	11	21	15	6	14	15	9	21	14	20	15	16
T15	16	21	16	13	16	17	17	17	15	16	16	17
T16	23	21	23	22	23	24	7	17	23	16	24	25
T17	15	21	13	14	13	14	12	5	12	4	13	14
T18	24	21	24	24	24	25	20	25	23	24	24	25
T19	4	1	14	4	15	16	3	9	13	8	14	15
T20	1	3	7	1	11	12	4	21	9	20	10	11
T21	3	6	8	2	12	13	7	17	11	16	12	13
T22	13	6	18	10	19	20	1	17	18	16	19	20
T23	2	6	4	3	7	8	2	5	10	4	11	12
T24	14	3	17	12	18	19	10	25	18	24	19	20

Table 3 : Ranking of soils with respect to WWTP sludge

	Best soil	Next options in the ranking	Worst soil
WWTP1	L2 (0.606)	L5 (0.592) L4 (0.588)	L6 (0.514)
WWTP2	L5 (0.619)	L2 (0.609) L3 (0.604)	L6 (0.454)
WWTP3	L2 (0.578)	L5 (0.557) L3 (0.551)	L6 (0.418)
WWTP4	L2 (0.682)	L5 (0.680) L3 (0.672)	L6 (0.566)

Table 3 presents the global utility values and ranking obtained at the end of the process. For each WWTP the ranking and utility degree of the soils is presented. This information is very useful for the decision maker that has to manage the sewage sludge application on those soils.

Considering that a single soil must be assigned to unique sludge, a reasonable decision could be the next one:

- WWTP1 sludge is sent to L4, because it permits a degree of suitability of 0.588, only slightly inferior to the best possibility for this sludge.

- WWTP2 sludge is sent to L5, because it is the best option, with a utility value of 0.619.
- WWTP3 sludge is sent to L2, because this is the best option for this sludge, reaching a value of 0.578,
- WWTP4 sludge is sent to L3, because this is the sludge that produces better evaluations with all the soils, and the three best options have all very good ratings. This option is evaluated with 0.672.

2.1. SENSITIVITY ANALYSIS ON THE WEIGHTS

The sensitivity analysis performed studies the effect of the parameters of the LSP method in the results, which are the weights w_i and the degree of andness/orness, r . Since the determination of weights has been quite difficult for the experts, we have concentrated on this parameter.

The sensitivity analysis has been performed converting the static weights, w , into a normal probability distribution, W , with a deviation of 10%, in the form $W=N(w,0.1w)$.

Table. 4: Number of different positions

	a	b	c	d	e	f	g	h	i	j	k	l
T23	2	1	2	1	4	1	2	1	1	1	1	1
T 8	3	1	2	2	1	4	1	2	2	1	1	1
T13	1	1	1	1	1	1	1	4	2	1	1	1
A v	2.1	1	1.2	1.7	1.4	1.7	1.3	3	1.5	1	1	1
MAx	5	1	2	4	4	4	3	6	3	1	1	1

The 3 first rows in Table 4 show an example of the sensitivity of 3 alternatives with respect to the weights. The forth row indicates the average in the number of different positions for each node, considering the 24 alternatives. Notice that last aggregation node 'a' is the most sensitive to the weights. In general, they are quite robust, with an average difference below 2 positions. Last row shows the maximum number of different positions that an alternative has suffered due to the weights change.

7 CONCLUSIONS AND FUTURE WORK

We have presented a decision support system for assisting the managers of biosolids produced at WWTP in finding the best distribution into agricultural soils.

The tool designed and developed follows the traditional MAUT approach, where two stages are considered: rating and ranking. The rating of the alternatives is achieved by means of the aggregation of partial utility values. In this case, we propose the use of a parametric operator that permits to model simultaneity and replaceability. In addition, fuzzy rules have been used to build a model for the utility evaluation in composite criteria. This is a novel approach to utility representation. From our experience,

fuzzy rules are suitable for implementing utility functions in composite criteria, because they are able to represent the expert knowledge about the behaviour of combinations of different variables in a particular domain that could not be modelled otherwise.

It is also worth to note that the results obtained by the system have been satisfactory. After an initial evaluation of these results, the experts affirm that the expected results were correlated to the results obtained. More tests with new samples of real data is going to be done. With more data, it will be possible to make a tuning of the parameters with machine learning techniques.

Now this model will be extended to the consideration of other destinations for biosolids, such as incineration in cement plants or disposal at landfill. Another future line consists in representing the data in geo-referenced maps.

Acknowledgements

Financial support was provided by the Spanish projects CTM2007-64490 and the SOSTAQUA funded by CDTI in the framework of the Ingenio 2010 – CENIT Program.

Referencias

- [1] J.J. Dujmović, H. Nagashima. LSP method and its use for evaluation of java IDE's. *Int. J. Approximate Reasoning*, vol. 41, no. 1, pp. 3–22, 2006.
- [2] J.J. Dujmović. Continuous Preference Logic for System Evaluation, *IEEE Transactions on Fuzzy Systems*, v.15:6, pp.1082-1099. 2007.
- [3] J. Figueira, S. Greco, M. Ehrgott (Eds). *Multiple criteria decision analysis. State of the art Surveys*, Springer, 2005.
- [4] J.S. Fuentes, F.C. Sagué, J.M.P. Grau, L.S. Genóher, C.N. Romeral. *Resultats de l'aplicació reiterada de fangs d'EDAR en els cultius del cereal*. IRTA Mas Badia Consorci Costa Brava: 13, 2007.
- [5] *Plan Nacional Integrado de Residuos 2008-2015* (PNIR), Anexo 5. Madrid, 2008.
- [6] M. Schuhmacker, A. Valls, J. Pijuan, A. Passuello, M. Nadal. Multicriteria analysis to manage sewage sludge application on agricultural soils, 19th Annual Meeting of the Society of Environmental Toxicology and Chemistry, pp.395-396, Goteborg (Sweden), 2009.
- [7] A. Valls, M. Schuhmacher, J. Pijuan, A. Passuello, M. Nadal. Some approaches to the use of MCDA tools for the management of sewage sludge application on agricultural soils, 69th Meeting of the EURO Working Group on Multiple Criteria Decision Aiding, pp. 10-11, Brussels (Belgium), 2009.
- [8] P. Vincke. Robust solutions and methods in decision aid. *Journal of Multicriteria Decisions Analysis*, 8, pp. 181-187, 1999.

Generación de emociones en agentes de recomendación usados en portales de comercio electrónico B2C

V. Herrera Tirado J.J. Castro-Schez J. Albusac C. González Morcillo

Escuela Superior de Informática, Universidad de Castilla-La Mancha,
Paseo de la Universidad 4, 13170 Ciudad Real
{Vanesa.Herrera, JoseJesus.Castro, JavierAlonso.Albusac, Carlos.Gonzalez}@uclm.es

Resumen

En este artículo se presenta un método basado en lógica difusa para generar las emociones de un agente virtual integrado en el portal de comercio electrónico B2C, e-Zoco, en base a los resultados obtenidos en el proceso de búsqueda de los productos. Tanto la información textual generada durante este proceso como la visual comunicada por el agente constituyen el módulo de recomendación encargado de asesorar a los clientes en la especificación de restricciones con el objetivo de obtener productos más interesantes para ellos, en un tiempo menor.

Palabras Clave: Lógica Difusa, Comercio Electrónico, Agentes Virtuales, Simulación de Emociones.

1. Introducción

La adaptación de los sistemas software al usuario final permite aumentar la usabilidad de los mismos [6]. Esta usabilidad, según Shackel [5], puede definirse en base a cuatro criterios operacionales: eficacia, capacidad de aprendizaje, flexibilidad y actitud. La eficacia está relacionada con la realización de un conjunto de tareas por un porcentaje de usuarios dentro del entorno de la aplicación. La capacidad de aprendizaje está definida en términos del tiempo de entrenamiento que necesita el usuario, dentro de un determinado nivel, para llevar a cabo una tarea. La flexibilidad se refiere a la diversidad de tareas y entornos permitidos por el diseño de la aplicación y, por último, los niveles de aceptación del usuario vienen dados por la actitud. Si aplicamos estos criterios a los portales B2C¹ surge

¹ *Business-to-Consumer*

la necesidad de establecer mecanismos de venta que se asemejen a los cotidianos, consiguiendo así aumentar su usabilidad. Estos mecanismos se basan en simular, en la medida de lo posible, la atención y apoyo que transmite un vendedor real al comprador, así como el tipo de información no precisa que se da en estas situaciones.

En esta idea se basa principalmente el portal de comercio electrónico e-Zoco [8] que es capaz de manejar la vaguedad e imprecisión intrínseca en la información que el usuario utiliza en la búsqueda de un producto y proporcionar recomendaciones relacionadas con dicha búsqueda. De esta forma el sistema interactúa con el comprador simulando la venta tal y como sucedería en la vida real. Pero en este tipo de portales se suele pasar por alto un factor básico utilizado en el proceso de comunicación entre el vendedor y el comprador: la emoción. En este artículo se presenta un mecanismo para la generación de emociones de un avatar empleando para ello un conjunto de reglas difusas. Se generarán las emociones en base a la salida del sistema de recomendación del portal B2C e-Zoco, simulando las emociones que transmitiría un vendedor real al interactuar con un determinado cliente.

2. e-Zoco: Portal de Comercio B2C

Con el objetivo de establecer el contexto en el que se aplican los métodos propuestos en este artículo, se describirá las principales características del portal e-Zoco. Se trata de un portal de comercio electrónico B2C desarrollado en trabajos previos [2] que hace uso de la lógica difusa para tratar la vaguedad e incertidumbre existente en la mente del comprador a la hora de especificar los criterios de búsqueda del producto deseado, así como para agrupar los resultados en base al posible interés que podrían tener para él.

Entre las principales características del portal cabe destacar *i)* un catálogo de productos; *ii)* un motor de búsqueda que trata la vaguedad e incertidumbre

presente en los criterios de búsqueda del cliente; finalmente, *iii*) un sistema de recomendación.

El catálogo de productos está organizado jerárquicamente de acuerdo a un conjunto de clases o categorías, donde la clase padre es la clase *objeto*. Cada producto en venta debe pertenecer al menos a una de las categorías. A su vez, cada uno de los productos está definido mediante un conjunto de variables las cuales determinarán sus características. A cada una de las variables se asocia un dominio de definición en el que se especifica el rango de valores que puede tomar la variable incluyendo los valores mínimo y máximo.

Por otro lado, la mayoría de clientes que hacen uso de portales de comercio electrónico tienden a emplear criterios de búsqueda vagos e imprecisos, sobre todo cuando no tienen una idea clara sobre el producto que buscan, y desean conocer las diferentes alternativas existentes en el mercado. Por este motivo es necesario la inclusión de servicios de selección de productos en portales de comercio electrónico que traten este problema. En particular, el servicio de selección de productos desarrollado en e-Zoco comienza con una primera fase en la que el usuario establece o define los criterios de búsqueda. Para ello se selecciona una categoría de productos del catálogo, se eligen un conjunto de variables de las que lo definen y se le asigna un valor preciso o impreciso a cada una de ellas. A esta especificación se le llama *descripción ideal del producto* y se utiliza para comparar los productos almacenados en la base de datos con las preferencias del usuario en el proceso de búsqueda. Además, el cliente puede especificar cómo de importante es cada una de las variables para él/ella, de tal forma que el peso o nivel de relevancia de cada variable influye notablemente en la lista de productos que finalmente se muestra al cliente. El siguiente paso consiste en calcular la similitud entre cada uno de los productos de la base de datos almacenados en la categoría que ha especificado el usuario, y la *descripción ideal del usuario*. Los productos que finalmente se muestran a los usuarios estarán ordenados de mayor a menor de acuerdo al valor de similitud que toma sus valores en el rango $[0, 1]$ calculado como se establece en [1].

Además el sistema de recomendación hace uso de la salida obtenida en el proceso de búsqueda para orientar al cliente en la redefinición de los criterios de búsqueda iniciales o en el establecimiento de nuevos criterios. Para ello, en primer lugar los productos se dividen en tres posibles clases: *Lo que buscas*, *Casi lo que buscas* y *Lejos de lo que buscas*. Esta división se realiza en base al valor de similaridad que toma cada producto con respecto a la *descripción ideal del producto* conforme al dominio de definición de la variable *Utilidad*. Siendo el primer valor el que determina el

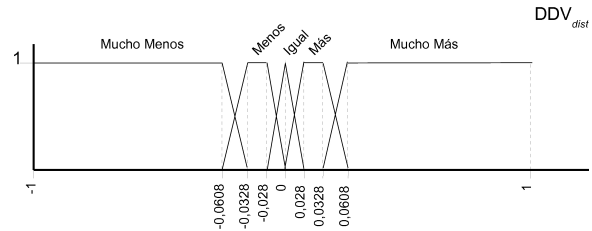


Figura 1: Dominio de definición de la variable *dist* (DDV_{dist}).

grupo donde se encuentran los productos donde la distancia es menor y el valor de similitud es mayor con la especificación proporcionada por el usuario.

En segundo lugar el sistema de recomendación puede generar un conjunto de reglas difusas de asociación entre variables que determinen la relación existente entre las variables de los productos. Un ejemplo de este tipo de reglas inferidas podría ser el siguiente: “Por *igual precio o más dinero* podría comprar cámaras con *mucha más* calidad de imagen. Donde *igual*, *más*, *mucho más* son valores lingüísticos tomados del dominio de definición de la variable *dist* (ver Figura 1) a partir del proceso de comparación de variables; *precio* y *calidad de imagen* son variables o atributos y, finalmente *cámaras* es la clase del producto. El conocimiento generado y mostrado al cliente contiene las relaciones que existen entre las características que describen los productos de una determinada categoría conforme a la especificación proporcionada por el cliente y permite redefinir la búsqueda o hacer búsquedas más realistas de acuerdo a los productos almacenados en el portal (la realidad del mercado). El uso de los consejos proporcionados por el asistente puede ayudar a encontrar los productos deseados con menor esfuerzo.

En las sucesivas secciones se describirá detalladamente como la inclusión de agentes virtuales con capacidad para generar emociones, a partir de los resultados obtenidos en la búsqueda de productos, puede mejorar notablemente el sistema de recomendación incluido en el portal de comercio electrónico descrito en esta sección.

3. GENERACIÓN DE EMOCIONES

El uso de agentes emocionales podría mejorar la interacción entre el usuario y los sistemas de comercio electrónico [4], además permite que el comprador entienda con mayor facilidad las recomendaciones realizadas por el sistema, obteniéndose así una interacción más rápida y sencilla.

Existen numerosas teorías que describen cómo se generan las emociones, siendo las más utilizadas en el

desarrollo de agentes emocionales las propuestas por Ekman [7] y el modelo OCC [9]. En nuestro caso, para determinar las emociones que puede simular el avatar, nos hemos centrado en el modelo de emociones de Ekman quien definió seis emociones básicas: miedo, sorpresa, asco, rabia, alegría y tristeza. Debido a que el personaje virtual se utiliza dentro del contexto del comercio electrónico se han tenido en cuenta un subconjunto de las emociones anteriormente descritas: sorpresa, alegría y tristeza. Estos tres tipos de emociones pueden darse de forma aislada o mezcladas. De esta forma el avatar puede estar alegre, triste o sorprendido, pero también puede expresar sorpresa y alegría, o sorpresa y tristeza, o bien una emoción de tristeza pero que trata de enmascarar aparentando estar alegre.

En el sistema propuesto en este artículo (ver Figura 2), las emociones del agente virtual vendrán determinadas por cuatro variables: *número de productos en el grupo lo que buscas* (v_1), *similitud media de los productos en el grupo Lo que buscas con la descripción ideal del producto* (v_2), *cantidad de reglas aprendidas para proporcionar consejo* (v_3) y *calidad de las reglas aprendidas para proporcionar consejo* (v_4). Estas variables toman sus valores en los dominios de definición que se muestra en la Tabla 1.

Cuadro 1: Variables (V) y Dominios de Definición de las mismas (DDV).

V_i	DDV_i
v_1	[0, 100]
v_2	[0, 0,028]
v_3	[0, 10]
v_4	[0, 100]

A continuación se describirá cómo se obtiene el valor de cada variable y cómo afecta a las emociones del agente:

- La variable número de productos en el grupo *Lo que buscas* (v_1) indica el porcentaje del total de productos de la clase en la que se hace la búsqueda que son agrupados en dicho grupo. Cuanto mayor es este número más opciones tiene el usuario para elegir, lo cual es bueno y por tanto debe reflejarse en el agente de recomendación.
- La variable similitud de los productos en el grupo *Lo que buscas con la descripción ideal de producto* proporcionada por el cliente (v_2), lo que mide es como de cercanos están como media los productos de dicho conjunto con la especificación de búsqueda dada por el usuario. Cuanto menor es este valor mejor, y tiene que tener una influencia sobre el agente de recomendación.

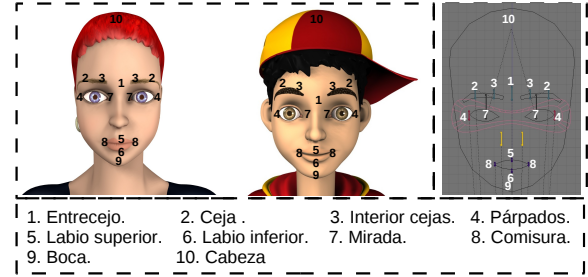


Figura 3: Manejadores de la cara definidos.

- La variable cantidad de reglas aprendidas para proporcionar consejo (v_3) puede tomar valores que van desde ningún consejo aprendido hasta en el peor de los casos el total de relaciones posibles entre valores de variables, es decir $|DDV_{dist}|^{|V|}$. Este caso límite nunca se va a dar ya que la cantidad de consejos está limitado en el portal a un total de diez. Esta variable tiene sentido emplearla cuando no se encuentran productos similares a lo que busca el usuario. Con respecto a como afecta esta variable a las emociones del agente, hay que indicar que en este caso los valores extremos no son buenos, es decir tan malo es no tener ningún consejo que proporcionar al usuario como proporcionar muchos consejos, ya que puede hacer que pierda la atención.
- La variable calidad de las reglas aprendidas para proporcionar consejo (v_4) indica el porcentaje medio de relaciones existentes en el catálogo (subcategoría del producto buscado) que son cubiertas por cada consejo. Es decir un porcentaje cercano al 100 % sería muy bueno ya que es una relación muy presente en la categoría. Por contra, valores cercanos al 0 % indicaría que las relaciones aprendidas no son fuertes.

Para la animación del personaje virtual se han aplicado técnicas de animación no lineal (NLA) [11]. NLA se utiliza para mezclar acciones simples individuales y obtener acciones complejas fluidas, de esta forma, se definen las acciones para cada una de las áreas faciales de forma individual y posteriormente se mezclan consiguiéndose así la animación facial completa. Con el fin de facilitar el movimiento de las diferentes áreas faciales se han definido manipuladores de alto nivel asociados a uno o varios músculos reales, así por ejemplo en lugar de músculo *orbicular* se ha definido el manipulador *párpados* (Fig. 3). Pero para que la simulación de emociones sea realista es necesario considerar diferentes grados de intensidad ya que es necesario distinguir, por ejemplo, entre *estar contento* y *estar muy contento*. Para simular esta intensidad se ha utilizado el posicionamiento de los manipuladores a distin-

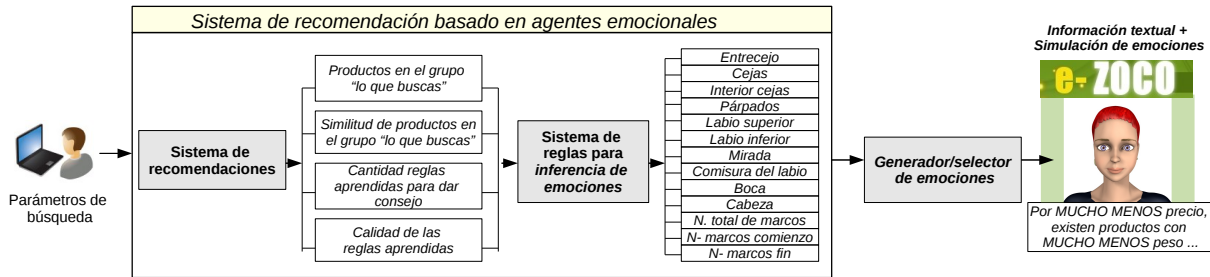
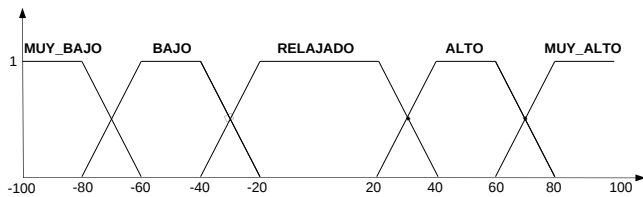


Figura 2: Esquema del sistema propuesto.

Figura 4: Dominio de definición de las variables *entrecejo*, *cejas*, *interior cejas*, *labio superior*, *labio inferior*, *mirada*, *comisura labio*.

tos niveles, por ejemplo elevando las cejas un poco o elevándolas mucho.

Al igual que en el trabajo presentado por Bevacqua et al. [10] se ha tenido cuenta el elemento temporal asociado a una emoción ya que según Ekman y Friesen [3] dependiendo del estado de ánimo varía la duración de una emoción, así por ejemplo la tristeza desaparece lentamente mientras que la alegría surge de manera muy rápida. En nuestro caso el elemento temporal se ha definido en función de *marcos* y teniendo en cuenta que un segundo de vídeo está formado por 24 marcos se han utilizado como parámetros temporales de la emoción el *número total de marcos*, el *número de marcos de comienzo* y el *número de marcos de fin*. El número de marcos que transcurre desde que empieza la emoción hasta que está en su punto álgido se obtiene con el *número de marcos de comienzo* y por el contrario con el *número de marcos de fin* se obtiene el número de marcos que transcurren desde que la emoción está en su punto álgido hasta que desaparece completamente del rostro del avatar. Por lo tanto, para la simulación de emociones se han tenido en cuenta los manejadores faciales, la intensidad de ejecución y el carácter temporal de las mismas, obteniéndose así trece variables de salida del sistema tal y como se puede observar en la Tabla 2.

Las etiquetas lingüísticas elegidas para cada uno de los conjuntos difusos, que por cuestiones de espacio no es posible mostrarlos todos (Figura 4), que forman parte de los dominios de definición de las variables de

Cuadro 2: Variables de salida (Y) y Dominios de Definición de las mismas (DDY).

Y_i	Nombre	DDY_i
y_1	Entrecejo.	$[-100, 100]$
y_2	Cejas	$[-100, 100]$
y_3	Interior cejas	$[-100, 100]$
y_4	Párpados	$[-100, 100]$
y_5	Labio superior	$[-100, 100]$
y_6	Labio inferior	$[-100, 100]$
y_7	Mirada	$[-100, 100]$
y_8	Comisura labio	$[-100, 100]$
y_9	Boca	$[-100, 100]$
y_{10}	Cabeza	$[-10, 10]$
y_{11}	N. marcos total	$[24, 84]$
y_{12}	N. marcos comienzo	$[0, 48]$
y_{13}	N. marcos fin	$[0, 48]$

entrada son los siguientes:

- $DDV_1 = \{\text{MUY\_POCOS, POCOS, ALGUNOS, BASTANTES, MUCHOS}\}.$
- $DDV_2 = \{\text{MUY\_CERCA, CERCA, LEJOS, MUY\_LEJOS}\}.$
- $DDV_3 = \{\text{POCOS, ALGUNOS, MUCHOS}\}.$
- $DDV_4 = \{\text{MALA, REGULAR, BUENA, MUY\_BUENA}\}.$
- $DDY_{1,2,3,5,6,7,8} = \{\text{MUY\_BAJO, BAJO, RELAJADO, ALTO, MUY\_ALTO}\}.$
- $DDY_4 = \{\text{CERRADOS, POCO\_CERRADOS, ABIERTOS, MUY\_ABIERTOS}\}.$
- $DDY_9 = \{\text{CERRADA, POCO\_ABIERTA, BASTANTE\_ABIERTA, MUY\_ABIERTA}\}.$
- $DDY_{10} = \{\text{MUY\_BAJA, BAJA, REPOSO, ALTA, MUY\_ALTA}\}.$
- $DDY_{11,12,13} = \{\text{MUY\_POCOS, POCOS, NORMAL, ALTO, MUY\_ALTO}\}.$

En la definición de las reglas se ha tenido en cuenta las diferentes reacciones emocionales de un vendedor real ante un conjunto de situaciones posibles. Así por ejemplo, si se obtienen pocos productos según las características de búsqueda establecidas por el usuario y

además no se encuentran cerca de lo que busca, la emoción que transmitirá el avatar será de gran tristeza. Pero si en este caso se le puede ofrecer al cliente un número restringido de consejos y además se aprenden bastantes reglas, la emoción se convertirá de tristeza pura a tristeza mezclada con alegría. Si los resultados obtenidos en general son negativos, el avatar transmitirá decepción en modo de tristeza y sorpresa. Todo lo contrario ocurre si se obtiene un número considerable de productos cerca de lo que busca el usuario y además se le ofrecen algunos consejos, aquí la emoción predominante será la alegría.

En cuanto a la generación de emociones y su simulación en personajes virtuales, nos hemos basado en la taxonomía facial de Ekman y Friesen [13] y la adaptación a un número determinado de músculos presentada por Cosi et al. [12] con las particulares modificaciones según nuestras variables de salida. Por cuestiones de espacio, se muestra a continuación sólo un subconjunto de las 28 reglas definidas para la inferencia de emociones:

R1: Si V_1 es MUY_POCOS y V_2 es MUY_LEJOS y V_3 es POCOS y V_4 es MALA **entonces** Y_1 RELAJADO, Y_2 MUY_ALTO, Y_3 MUY_ALTO, Y_4 POCO_CERRADOS, Y_5 MUY_BAJO, Y_6 MUY_BAJO, Y_7 MUY_BAJO, Y_8 MUY_BAJO, Y_9 CERRADA, Y_{10} [BAJA, MUY_BAJA], Y_{11} ALTA, Y_{12} NORMAL, Y_{13} MUY_ALTA.

R2: Si V_1 es POCOS y V_2 es LEJOS y V_3 es ALGUNOS y V_4 es MALA **entonces** Y_1 RELAJADO, Y_2 ALTO, Y_3 MUY_ALTO, Y_4 POCO_CERRADOS, Y_5 RELAJADO, Y_6 RELAJADO, Y_7 BAJO, Y_8 ALTA, Y_9 CERRADA, Y_{10} BAJA, Y_{11} ALTA, Y_{12} NORMAL, Y_{13} ALTA.

R3: Si V_1 es MUCHOS y V_2 es [MUY_CERCA, CERCA] **entonces** Y_1 RELAJADO, Y_2 RELAJADO, Y_3 MUY_ALTO, Y_4 MUY_ABIERTOS, Y_5 RELAJADO, Y_6 RELAJADO, Y_7 BAJO, Y_8 ALTA, Y_9 BASTANTE_ABIERTA, Y_{10} ALTA, Y_{11} NORMAL, Y_{12} MUY_POCOS, Y_{13} NORMAL.

R4: Si V_1 es ALGUNOS y V_2 es LEJOS **entonces** Y_1 RELAJADO, Y_2 ALTO, Y_3 ALTO, Y_4 POCO_CERRADOS, Y_5 BAJO, Y_6 BAJO, Y_7 RELAJADO, Y_8 ALTA, Y_9 POCO_ABIERTA, Y_{10} BAJA, Y_{11} NORMAL, Y_{12} NORMAL, Y_{13} NORMAL.

De este modo, la emoción del avatar vendrá determinada por la inferencia obtenida sobre las variables de salida a partir de las variables de entrada usando este conjunto de reglas.

4. EJEMPLO DE FUNCIONAMIENTO

En esta sección se expone un caso de uso del sistema descrito, en particular la búsqueda por parte del usuario de una cámara digital. Para ello, el sistema cuenta con una base de datos con una categoría de



Figura 5: Emociones obtenidas a partir de las reglas (a) R18 y (b) R13.

nominada *Cámaras Digitales* compuesta por 119 elementos y que está descrita por 51 variables como el *precio*, *modelo*, *resolución*, etc.

La entrada del sistema consiste en una descripción imprecisa, por parte del usuario, de la cámara deseada. Para ello suponemos que introduce los siguientes datos: *Precio* = 400, *Peso* = 300 y *Longitud* = 18, con una precisión del 95% y una importancia normal (0,33). El proceso de selección (o búsqueda) devuelve una lista de cámaras digitales ordenadas según sus valores de distancia global, que posteriormente es dividida agrupando los productos según la variable *similitud*.

En este caso de estudio, la primera búsqueda del usuario no devuelve ningún producto, por lo tanto no aparece ningún elemento en los grupos *lo que busca* o *cerca de lo que busca*. Pero si se pide consejo al sistema estudiando los tres parámetros (precio, peso y longitud) e-Zoco muestra seis recomendaciones siendo una de ellas: *Existen 47 productos con MUCHO MENOS Precio que el que usted busca, por mucho menos valor de Peso especificado/a, Y por el mismo o un poco más o mucho más valor de Longitud especificado/a.*

Teniendo en cuenta las variables de entrada para la simulación de emociones, la regla que se dispara en este caso es:

R18: Si V_1 es MUY_POCOS y V_2 es [LEJOS, MUY_LEJOS] y V_3 es ALGUNOS y V_4 es BUENA **entonces** Y_1 RELAJADO, Y_2 MUY_ALTO, Y_3 MUY_ALTO, Y_4 POCO_CERRADOS, Y_5 RELAJADO, Y_6 RELAJADO, Y_7 BAJO, Y_8 ALTO, Y_9 CERRADA, Y_{10} REPOSO, Y_{11} ALTA, Y_{12} NORMAL, Y_{13} MUY_ALTA.

Como resultado se obtiene la emoción que se muestra en la Figura 5 (a), en ella se puede observar una mezcla de emociones. Aparece tristeza en el rostro del avatar (cejas arqueadas, mirada baja y ojos un poco cerrados) por no haber encontrado productos que se ajusten a las necesidades del usuario, pero gracias a las recomendaciones que muestra el portal y a la calidad de las mismas, surge también la emoción de alegría (comisura de los labios hacia arriba).

Supongamos que el usuario modifica la especificación inicial de búsqueda guiándose por el consejo anteriormente descrito: *Precio = 200*, *Peso = 180* y *Longitud = 18*, con la misma precisión e importancia que en la primera búsqueda. En este caso el sistema recomienda la cámara *Sony Cyber-shot DSC-T200* (en el grupo *Lo que buscas* y la emoción resultante, que se muestra en la Figura 5 (b), se obtiene con la regla siguiente:

R13: Si V_1 es MUY_POCOS y V_2 es MUY_CERCA entonces Y_1 RELAJADO, Y_2 RELAJADO, Y_3 RELAJADO, Y_4 POCO_CERRADOS, Y_5 RELAJADO, Y_6 RELAJADO, Y_7 RELAJADO, Y_8 ALTO, Y_9 POCO_ABIERTA, Y_{10} REPOSO, Y_{11} NORMAL, Y_{12} POCOS, Y_{13} NORMAL.

En este caso la emoción predominante en el rostro del avatar es la alegría aunque con una intensidad no muy elevada debido a que se ha encontrado un número bajo de elementos que se ajusten a las necesidades de búsqueda introducidas.

Un ejemplo de cómo sería la interacción del avatar con el usuario puede obtenerse introduciendo los parámetros de búsqueda empleados en esta sección en el portal e-Zoco.

5. CONCLUSIONES

En este trabajo se ha desarrollado un sistema basado en reglas difusas para controlar las emociones de un avatar que pudiera actuar como un vendedor virtual dentro del portal de comercio electrónico e-Zoco. Este avatar comunica los resultados de las búsquedas y proporciona consejo a los usuarios del portal. La emoción con la que transmitirá sus mensajes al usuario dependerá de la información generada en el portal durante un proceso de búsqueda y generación de recomendaciones.

Los primeros análisis realizados sobre la opinión de los usuarios sobre la utilidad de la inclusión del avatar con emociones en el portal revelan datos prometedores, ya que es valorado positivamente frente a la no inclusión. No obstante estos datos deberán ser refrendados en el futuro.

Agradecimientos

Este trabajo ha sido financiado por el Proyecto Ganas de la Cátedra Indra-UCLM.

Referencias

[1] J.J. Castro-Schez, J.L. Castro, J.M. Zurita. Fuzzy Repertory Table, a Method for Acquiring Knowledge about Input Variables to Machine Learning Algorithm. *IEEE Transactions on Fuzzy Systems*, 12(1):123-139, 2004.

- [2] L. M. López-López, J. J. Castro-Schez, D. Vallejo-Fernandez, J. Albusac. A Recommender System based on Machine Learning Algorithm for B2C Portals. *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'09)*, 524-531, Milan, Italia, 2009.
- [3] P. Ekman, W. Friesen. Unmasking the Face: A guide to recognizing emotions from facial clues. Prentice-Hall, Inc., 1975.
- [4] N. Jascanu. Emotionally based multi-agent e-commerce platform. *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems*, 1738-1739, Estoril, Portugal, 2008.
- [5] B. Shackel. Human factors and usability, *Human-computer interaction*, Prentice Hall Press, 27-41, 1990.
- [6] D. Benyon. Adaptive systems: A solution to usability problems. *User Modeling and User-Adapted Interaction*, 3(1):65-87, 1993.
- [7] P. Ekman, W. V. Friesen, P. Ellsworth. Emotion in the human face: guidelines for research and a review of findings. New York. Permagon, isbn 978-0195155327, 1972.
- [8] Grupo de Investigación Oreto, e-Zoco (B|C)2C. <http://oreto.esi.uclm.es/aplicaciones/ezoco>.
- [9] A. Ortony, G. L. Clore, A. Collins, *The cognitive structure of emotions*, Cambridge University Press, isbn 978-0521386647, 1998.
- [10] E. Bevacqua, Mancini, C. Pelachaud. Speaking with Emotions. *AISB Symposium on Motion, Emotion and Cognition*, 197-214, Leeds, U.k, 2004.
- [11] A. Watt, M. Watt. *Advanced Animation and Rendering Techniques*, Addison-Wesley Professional, isbn 978-0201544121, 1992.
- [12] P. Cusi, A. Fusaro, D. Grigoletto, G. Tisato. Data-Driven Tools for Designing Talking Heads Exploiting Emotional Attitudes. *Proceedings of Tutorial and Research Workshop, Affective Dialogue Systems*, 101-112, Kloster Irsee, Alemania, 2004.
- [13] P. Ekman, W. V. Friesen. *Facial Action Coding System*. Consulting Psychologists Press, 1978.

SOFTCOMPUTING PARA LA ELABORACIÓN DE MODELOS PREDICTIVOS DE CALIDAD SUPERFICIAL EN FRESADO A ALTA VELOCIDAD

Víctor M. Flores

Maritza Correa

Jose R. Alique

Departamento de Informática Industrial, Instituto de Automática Industrial, Consejo Superior de Investigaciones Científicas, Madrid-España. {victor.flores, macorrea, jrallique}@iai.csic.es

Resumen

La calidad superficial (frecuentemente estimada en función a la rugosidad superficial R_a) en procesos de mecanizado es uno de los aspectos más cuidados en la fabricación y depende en gran medida de la combinación de factores en el proceso de corte [16]. Los trabajos que incorporan técnicas de softcomputing al estudio de la rugosidad superficial (R_a) en-proceso o pos-proceso son relativamente frecuentes en la literatura, sin embargo son casi inexistentes los dedicados al estudio de la R_a en pre-proceso. Contar con técnicas de predicción del acabado superficial pre-proceso puede ayudar, por ejemplo, a reducir los gastos asociados de la estimación de la calidad superficial basada en medidas de R_a tras procesos de corte mecánico. En este trabajo se presenta una técnica softcomputing para generar un modelo predictivo de R_a en pre-proceso basado en una experimentación que relaciona características del proceso de fresado a alta velocidad. El modelo de predicción es un clasificador Bayesiano desarrollado con una herramienta software especializada, y validado con el método k-fold cross-validation. Se obtuvo una precisión en la clasificación de 77,6%.

Palabras Clave: Mecanizado a alta velocidad, fresado a alta velocidad, redes Bayesianas, modelos predictivos.

1 INTRODUCCIÓN

En procesos de mecanizado a alta velocidad (MAV), el arranque de material es uno de los que produce mayor impacto económico en la industria de fabricación con metales. Esto gracias (entre otros aspectos) al acabado superficial posible de obtener, de gran importancia en dominios como la aeronáutica [1]. En muchos casos el acabado superficial influye en el comportamiento funcional de la pieza resultante. Por tanto, la calidad superficial en procesos de MAV es uno de los factores más importantes a cuidar en la fabricación, y que depende en gran medida de la combinación de factores como las propiedades del material utilizado, las características de la máquina y de la herramienta [16].

También en la fabricación con metales es frecuente estimar la calidad de la pieza mecanizada en función a su textura superficial. La integración de varias cantidades se puede usar para describir la textura superficial, de entre ellas, la rugosidad superficial (R_a) es el parámetro más ampliamente utilizado debido a su influencia sobre la fricción, resistencia eléctrica y térmica, apariencia, etc. de la pieza mecanizada, y que además puede ser calculada de forma experimental. La R_a es importante en un mecanizado por varias razones, como por ejemplo: puede aportar ideas de cómo se comporta una superficie en contacto con otras [9], lo que es vital en dominios como el mencionado anteriormente.

Son múltiples los trabajos que estudian las posibles combinaciones de los factores que inciden en el proceso de MAV para calcular o predecir la R_a en-proceso o pos-proceso (durante o después del proceso de mecanizado de la pieza, respectivamente). Sin embargo, son casi inexistentes los trabajos de predicción pre-proceso de la R_a , es decir, antes de iniciar el proceso de corte mecánico. En este trabajo se describe la experimentación realizada para construir un modelo predictivo pre-proceso de la R_a , asociado con condiciones específicas de preparación del proceso de MAV, usando técnicas de softcomputing.

1.1. CALIDAD SUPERFICIAL Y SISTEMAS CAD/CAM

La construcción de modelos de piezas con determinadas características de acabado es un proceso cotidiano en la industria. Actualmente, en técnicas como arranque de material o fabricación de moldes, la integración de sistemas CAD/CAM se usa frecuentemente para generar los programas pieza que usa el CNC (*control numérico computerizado*) para alcanzar valores deseados de R_a [6, 8]. Los pasos principales de este proceso se esquematizan en la figura 1, y pueden ser descritos de la forma siguiente:

- De forma general, el primer paso consiste en la caracterización geométrica deseada de la pieza, mediante un sistema CAD (*computer aided design*),

es decir, aporta características o especificaciones geométricas como la forma, acabado, etc.

- La representación geométrica de la pieza generada con este tipo de software representa la entrada a la aplicación CAM (Computer-aided manufacturing) que se usa para generar las secuencias de instrucciones o programa pieza que usa el CNC, para completar el proceso de corte mecánico.

Normalmente, y como paso previo al mecanizado (ver el esquema de la figura 1), el programa pieza es simulado en el CNC para verificar características geométricas.

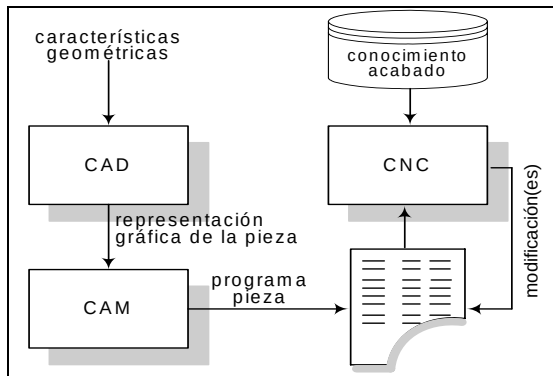


Figura 1: Integración de sistemas CAD/CAM con CNC.

Los sistemas CAD/CAM más actuales (como por ejemplo Catia¹) tienen entre otras capacidades, la posibilidad de admitir modificaciones del CNC en el diseño geométrico o admitir entradas de modificaciones CNC directamente sobre el CAM para cambiar el programa pieza (tras la validación de la simulación), pero entre sus capacidades no cuenta con la posibilidad de predecir características de acabado superficial, en los procesos de corte mecánico.

1.2. NECESIDAD IDENTIFICADA

Específicamente en el proceso de fresado, es el operario quien modifica las condiciones de corte en el programa pieza (avance y giro) para adecuar el mecanizado y aproximarlos al acabado que desea obtener. Esto lo hace normalmente el operario basándose en su experiencia y en las observaciones in-situ del proceso de corte. Cuando este proceso se programa para una producción en serie, la búsqueda por ensayo y error de una calidad superficial específica puede resultar costosa, tanto en tiempo como en uso de materiales. Dicho de otra forma: con la combinación de técnicas existentes, la comprobación de la calidad superficial (mediante la R_a) en un proceso fresado no es posible si no se mecaniza la pieza.

Se precisa entonces de mecanismos que faciliten la predicción de la calidad superficial deseada antes del proceso de corte mecánico, para no tener que recurrir a

técnicas como mecanizar una o varias piezas para verificar la validez del programa pieza o como el pulido pos-proceso para mejorar la calidad superficial. En concreto se identifica en este sector del mercado necesidades de contar con posibilidades como: (1) dada una R_a deseada, definir en el diseño las condiciones de corte adecuadas para lograrla o (2) según ciertas condiciones geométricas y de corte, predecir la R_a futura (posible de obtener), sin pasar por la etapa de experimentación.

Para lo expuesto en (1) es escasa la documentación, mientras que para dar respuesta a lo planteado en (2), existe en la literatura trabajos recientes que predicen la R_a según ciertas condiciones específicas. Buena parte de los trabajos consultados relacionan variables de una máquina, un tipo de herramienta particular y un material específico para proponer modelos predictores de la calidad superficial. Algunos de estos trabajos usan técnicas de softcomputing, ejemplos de éstos son los modelos predictivos que estiman valores futuros o desconocidos de variables de interés (variables objetivos o dependientes), usando otras variables independientes o predictoras. Casos concretos de este tipo de trabajos son [2, 3]. En el primero se usan redes Bayesianas con estructuras naïve Bayes y TAN (Tree Aumented naïve Bayes [6]) para estimar la R_a en proceso de fresado a alta velocidad, usando acero F114 para mecanizado de ranuras y en el segundo trabajo se describen modelos predictivos de R_a usando redes neuronales artificiales y redes Bayesianas, con aleaciones de aluminio para mecanizar formas cóncavas y convexas diversas (de diferentes radios y profundidades de corte axial).

En este documento se describe el trabajo realizado usando estructuras naïve Bayes y TAN para generar un modelo predictivo de la R_a en pre-proceso para fresado, en condiciones específicas. Para generar el modelo se usan variables predictoras y datos experimentales extraídos de la configuración experimental descrita en [2].

El resto de este documento está estructurado de la forma siguiente: en el apartado 2 se describen brevemente trabajos recientes que aplican técnicas de softcomputing, tanto para generar modelos predictivos de R_a como en aplicaciones software que se incorporan a las librerías software de un CNC para mejorar sus capacidades de estimación de valores de R_a pos-proceso, el apartado 3 describe el modelo propuesto para la predicción de la R_a , el 4 detalla los resultados obtenidos con el clasificador Bayesiano propuesto, en el apartado 5 se presentan las conclusiones y en el 6 las posibles líneas de trabajos futuros, seguido de la bibliografía.

2 TRABAJOS RELACIONADOS

Trabajos recientes en la industria de corte de metales reflejan la aplicación, cada vez mayor de técnicas de

¹ <http://www-01.ibm.com/software/applications/plm/catia/v5/>

softcomputing en predicción de R_a en procesos de mecanizado. Por ejemplo en [7] y [5] proponen técnicas basadas en redes neuronales artificiales para optimizar los valores de parámetros que participan en el proceso de corte, en [16] y [17] se aplican igualmente redes neuronales artificiales para calcular la incidencia de las fuerzas de corte en la R_a .

En [11] se describe un método que usa redes neuronales artificiales para estimar la influencia de los parámetros de la máquina (velocidad de corte, avance y fuerzas aplicadas) y de la herramienta (avance por diente y número de dientes) en la R_a con acero AISI1054. Concretamente las fuerzas consideradas son fuerza de corte (fc), fuerza de arranque (ff) y fuerza de avance (ft) que se usan para generar la fuerza del proceso mecanizado (fm) según la ecuación:

$$fm = \sqrt{fc^2 + ff^2 + ft^2}$$

Estas fuerzas y otros parámetros como los mencionados anteriormente alimentan la red (entrenada con casos con diferentes condiciones de corte y considerando varias características herramientas) que genera entre las salidas la R_a predicha. Los resultados de las pruebas en este trabajo (usando herramienta enteriza) fueron comparados con otros obtenidos en pruebas similares usando herramientas de plaquitas intercambiables, los resultados verifican la conveniencia de usar herramientas enterizas para obtener mejores resultados de R_a . Otro trabajo reciente de modelo predictivo de R_a , pero trabajando con piezas de aluminio es el descrito en [14]. En este trabajo aplican técnicas de softcomputing para generar un modelo para predecir la R_a con variables como avance por diente, profundidad de corte, velocidad de corte, etc. en un proceso de mecanizado.

El trabajo descrito en [2] detalla el experimento realizado, usando una máquina Kondia HS1000 equipada con un CNC Siemens 840D. El material usado para estas pruebas fue acero F114 sobre el que se realizaron cortes a diferentes profundidades, siendo la máxima de 1.2 mm. Las herramientas usadas para los cortes (Karnash modelo 30.6472) tienen diferentes diámetro y número de dientes.

El entrenamiento para generar un modelo Bayesiano que relaciona la R_a con variables predictoras en-proceso se realiza en este trabajo usando el software Elvira², que por razones de funcionamiento requiere valores discretos en todas las variables. A tal efecto se seleccionó el algoritmo K-means [10] para la discretización detallada en la tabla 1 (los detalles de este proceso pueden consultarse en [2]).

Particularmente las etiquetas de R_a se asignaron de acuerdo a los valores medios de rugosidad superficial

establecidos en la norma ISO 1302:2002. Parte del objetivo en este experimento fue medir la R_a pos-proceso. Para calcular la R_a en [2] se usa la ecuación:

$$Ra = \frac{1}{L} \int_0^L |f(x)| dx$$

Donde $f(x)$ es la función usada para medir valores medios de rugosidad (por contacto) con intervalos de longitud L sobre la pieza mecanizada. Entre los resultados de este trabajo destaca el uso de una red Bayesiana para estimar valores de R_a dados ciertos valores de las variables predictoras, y el caso contrario: dado un valor de R_a deseada, calcula los valores (las probabilidades) de las variables predictoras para obtenerla.

También se encuentra en la literatura trabajos recientes que incorporan técnicas de softcomputing para modificar programas de CNC. Un ejemplos de estos es [13] que usan un modelo de regresión múltiple y redes neuronales artificiales para generar un modelo de predicción de la R_a . El valor obtenido para ciertas condiciones se incorpora al CNC para mecanizar aleaciones de aluminio. Los resultados experimentales demuestran que se mejoran los resultados en el acabado.

3 DESCRIPCIÓN DEL MODELO

Generalmente, los trabajos en esta área de investigación focalizan la construcción de modelos predictivos de R_a a las siguientes características: la geometría deseada, la máquina y las herramientas usadas, y al proceso mecánico específico [12]. Por otro lado, la experimentación en este campo suele ser costosa en términos de tiempo y recursos. Considerando estas experiencias y particularidades de experimentación, en este trabajo se usan para generar el modelo, los datos experimentales y discretizaciones de variables descritos en [2].

En concreto se seleccionaron 5 variables predictoras, dos relacionadas con la herramienta: el diámetro (diam) y el número de dientes (flutes) y tres relacionadas con el proceso: la profundidad de corte radial (ap), el avance (feed-rate) y la velocidad de giro (rpm).

Respecto a la conveniencia del uso del clasificador Bayesiano seleccionado, en [2] se explica la eficiencia de usar redes Bayesianas en la construcción de un modelo predictivo de R_a que relaciona características del proceso de MAV con las variables que inciden en el proceso.

Los modelos Bayesianos aprenden a partir de datos, en este experimento se usaron datos experimentales recolectados de la forma que se detalla en [2] igualmente en este documento se puntualiza sobre el diseño experimental diseñado para asegurar la validez estadística de los datos recolectados.

² <http://www.ia.uned.es/~elvira/index-en.html>

Tabla 1: Rangos y valores discretos de variables del dominio

ESTADO	Ra (μm)	flutes	diam (mm)	ap (mm)	feed-rate (mm/min)	rpm
0	Liso (L)=[0.7,1.1)	2	6	0.20	<1000=[300,1000)	Lento=[5250,7964)
1	Fino (F)=[1.1,2.1)	6	8	0.40	<2050=[1000,2050)	Medio=[7964,10758)
2	Semifino(S)[2.1,5.1)		10	0.60	<2775=[2050,2775)	Semi-rápido=[10758,14813)
3	Medio(M)=[5.1,8.5)		12	1.00	<3850=[2775,3850)	Rápido=[14813,19400)

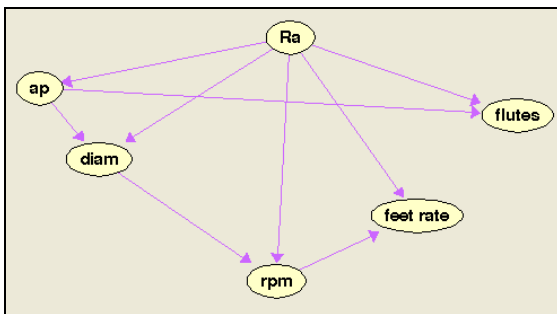
El clasificador usado en el trabajo antes citado es el clasificador Bayesiano aumentado a árbol TAN, que es un extendido del clasificador Bayesiano naïve Bayes con una estructura de árbol sobre las variables predictoras. Este árbol se obtiene adaptando el algoritmo propuesto en [4] y calculando la información mutua condicional para cada par de variables dada la clase. La clase R_a se discrimina como se expresa en la tabla 1.

Una ventaja adicional del uso de TAN frente a otros clasificadores como por ejemplo las redes neuronales artificiales, es la capacidad interpretativa de los resultados y la forma intuitiva en que describe gráficamente las relaciones causales entre los nodos de la red, lo que facilita la interpretación de resultados. La figura 2 ilustra la estructura TAN aprendida, en la que se aprecian los arcos causales entre las variables y la clase.

La información proporcionada que relaciona los nodos de la figura 2 puede interpretarse como influencias probabilísticas en las relaciones físicas existentes entre las variables predictoras, difícilmente deducibles de ecuaciones físicas aplicables al proceso de corte mecánico. Por ejemplo, se “descubre” que el número de dientes depende de la profundidad de corte radial.

También se deduce una dependencia lineal entre las variables ap, diam, rpm y feed-rate, lo que puede interpretarse como que una profundidad radial de corte radial deseada incidiría en la selección del diámetro de la herramienta, que a su vez determinaría la velocidad de giro que fijaría el avance por diente.

La tabla 2 refleja los valores de las probabilidades condicionadas entre las variables. La probabilidad más alta de la variable flutes es para 2 dientes (0.72), lo que inclina el R_a al valor Liso (0.60).

Figura 2: Red TAN generada con Elvira¹.

Analizando la tabla 2 se observan influencias con altas relaciones de probabilidad entre algunos nodos, priorizando el valor de flutes=2 (por ser la más alta inicialmente) se observa que la mayor variación para R_a se ubica en el rango evaluado como Medio. En contra, si se prioriza flutes=6 se observa que la mayor variación en R_a se ubica en Fino con una probabilidad del 29%.

Tabla 2: Tabla de probabilidad condicionada

ESTADO	flutes	ap	feed-rate	rpm	diam	R_a
0	0,72	0,26	0,29	0,20	0,22	0,60
1	0,28	0,19	0,29	0,27	0,22	0,29
2		0,23	0,20	0,32	0,19	0,21
3		0,32				

Los valores numéricos de las probabilidades condicionadas expresados en la tabla 2 sugieren conclusiones como las siguientes: La calidad del acabado no parece depender tanto de la velocidad de giro (rpm), sino: (1) de la relación entre la profundidad del corte radial (ap) y el número de dientes de la herramienta (flutes) y (2) de la relación entre la profundidad del corte radial (ap) y la selección del diámetro de la herramienta (diam), velocidad de giro (rpm) y la tasa de avance (feed rate) fijadas en el proceso de mecanizado.

4 VALIDACIÓN

Un objetivo en esta sección es reflejar la estimación de la probabilidad de acierto del modelo de clasificación, ante casos nuevos. A tal efecto se tomó como método de validación el K-Fold Cross-Validation [15] que básicamente consiste en dividir el conjunto total de casos en K subconjuntos disjuntos. De estos K subconjuntos uno se reserva como datos de validación para probar el modelo y los restantes se utilizan como datos de entrenamiento. El proceso se repite K veces (con cada uno de los K usado sólo una vez como datos de validación). El resultado de esta validación se muestra en la tabla 3.

En concreto la precisión en la clasificación resultó del 77,6%. Analizando los datos de la tabla 3, se observan valores resaltantes como una exactitud del 100% para los casos de R_a =Liso. Otro aspecto importante es que el resto de las clasificaciones supera en todos los casos al 60% como correctamente clasificados. La experimentación arrojó un total de 625000 medidas de variables predictoras descritas en el apartado 3, que fueron filtrados

según objetivos de experimentación (según los rangos de variables expresados en la tabla 1) y agrupados en 251 casos, generados a partir de valores medios del total de medidas. Estos casos fueron finalmente usados para el entrenamiento.

También se usaron las fórmulas descritas en la tabla 4 para estimar las medidas de mérito de: sensibilidad, especificidad, precisión, exactitud, tasa de acierto y tasa de error del modelo respecto a los datos de entrenamiento. Las abreviaturas usadas en estas formulas corresponden a los siguientes valores: VP= verdaderos positivos, VN= verdaderos negativos, FP=falsos positivos y FN= falsos negativos.

Tabla 3: Matriz de confusión.

Real \ Asignado	Liso	Fino	Semi-fino	Medio
Liso	30 (100%)	0	0	0
Fino	17	82 (78,8%)	4	1
Semi-fino	2	16	38 (62,3%)	5
Medio	1	2	8	44 (80%)

De los valores de estas medidas de mérito reflejados en la tabla 4 es destacable lo siguiente:

- El modelo tiene una alta probabilidad de acierto y precisión de clasificación (con 0.88 y 0.96 respectivamente).
- La proporción de clasificación de verdaderos positivos es bastante alta (supera el 0.80)

Tabla 4: Fórmulas y valores de medidas de mérito.

$sensibilidad = \frac{VP}{VP + FN}$	0.83
$especificidad = \frac{VN}{FP + VN}$	0.95
$precisión(Pr) = \frac{VP}{VP + FP}$	0.96
$exactitud = \frac{VP + VN}{VP + FN + FP + VN}$	0.88
$tasa\_de\_error = \frac{FN + FP}{VP + FN + FP + VN}$	0.12
$tasa\_de\_acierto = \frac{VP + VN}{VP + FN + FP + VN}$	0.88

5 CONCLUSIONES

En este trabajo se presenta una técnica softcomputing para generar un modelo predictivo de R_a en pre-proceso, basado en la experimentación descrita en el apartado 2. El modelo de predicción es un clasificador Bayesiano desarrollado usando la herramienta software Elvira. La red obtenida con el clasificador TAN se basa como se ha

dicho en parámetros de corte y características de herramienta para obtener la predicción, con resultados fiables (estadísticamente hablando).

Las redes Bayesianas con estructura naïve Bayes y TAN son estructuras usadas en trabajos previos, donde se ha validado su aplicabilidad a problemas de predicción de la R_a en procesos de corte mecánico a alta velocidad, pero no dejando de lado las condiciones de corte, por lo que el trabajo presentado puede aportar en áreas de investigación como (1) desarrollo de aplicaciones software para actuar sobre sistemas CAM, es decir: incorporar modelos de conocimiento de R_a en sistemas CAM para generar predicciones de acabado más precisas y ajustadas a la realidad del mecanizado.

También puede servir de soporte a mejorar las calidades obtenidas, pero también los costes de tiempo y material usado en estos procesos. (2) incorporación a herramientas software de la posibilidad de introducir valores de R_a deseado para generar los parámetros de corte adecuados. Es decir, con una calidad superficial deseada poder definir avance, velocidad de giro y otros aspectos como características de la herramienta (por ejemplo diámetro, número de dientes, etc.), profundidad de corte, etc. que resulten más adecuados para obtener la R_a deseada, para incorporarlos directamente al programa pieza.

Un tercer aporte (potencial) es, dado que en el contexto de trabajo actual se puede planificar un mecanizado pero no verificar su acabado hasta el pos-proceso, modelos como el propuesto podrían incorporarse en herramientas software de control de procesos mecánico. Esto podría asegurar mejores resultados para unas condiciones dadas, puesto que el seguimiento del mecanizado se mejoraría sustancialmente.

Actualmente y como ilustra la figura 3, existen aplicaciones para controlar el proceso de corte, pero es habitual la participación del operario, quien en base a su experiencia y a la observación del proceso de mecanizado, altera las condiciones de corte en pro de un mejor acabado.

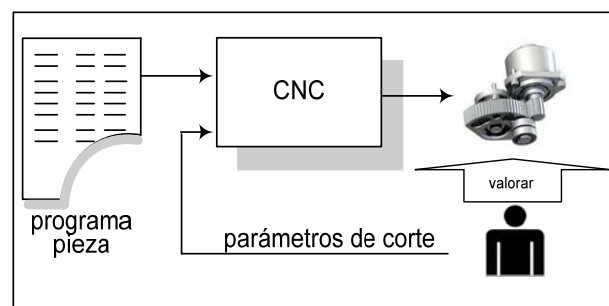


Figura 3: Esquema (simplificado) de supervisión y modificación de condiciones de corte en proceso.

6 LÍNEAS FUTURAS

Como se ha dicho, los resultados del entrenamiento del clasificador son buenos (77,6% de precisión en la clasificación), sin embargo, el modelo generado puede usarse con nuevos casos para lograr su generalidad, aprovechando las bondades del modelo Bayesiano seleccionado.

La experimentación descrita anteriormente representa un paso adelante en la predicción pre-proceso de Ra, pero la generalidad del modelo depende de realizar otros entrenamientos para disponer de nuevos casos, con por ejemplo condiciones de corte diferentes, pero con los mismos parámetros de experimentación. También se estima conveniente usar otros paradigmas clasificatorios como por ejemplo redes neuronales artificiales sobre los mismos datos y diseño experimental, para validar el modelo propuesto.

Otro tópico importante de atender es la aplicación de los resultados del modelo a herramientas CAD/CAM, lo que puede comenzar por validarse las conclusiones expresadas en este trabajo respecto a las relaciones de dependencia probabilística entre las variables predictoras que relacionan características de la herramienta de corte con características de la máquina usada. También puede resultar interesante verificar estas relaciones de dependencia, en otros casos particulares, como por ejemplo experimentación usando el mismo material y herramientas para mecanizar con una máquina de características diferentes a la máquina Kondia HS1000.

Referencias

- [1] Y. Altintas, M. Weck, "Chatter stability of metal cutting and grinding," *CIRP Annals Manufacturing Technology*. Vol. 53. pp. 619-642. 2004.
- [2] M. Correa, C. Bielza, M. Ramírez, J.R. Alique, "A Bayesian network model for surface roughness prediction in the machining process". : *International Journal of Systems Science*. Vol. 39, Number 12. pp. 1181-1192. December. 2008.
- [3] M. Correa, C. Bielza, J. Pamies-Teixeira, "Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process". *Expert Systems with Applications: An International Journal*, Vol. 36 Number 3, pp. 7270-7279. April. 2009.
- [4] C.K. Chow, C. Liu "Approximating discrete probability distributions" *IEEE Transactions on Information Theory*. Vol. 14. pp. 462-467. 1968.
- [5] E. O. Ezugwua, D. Faderea, J. Onney, R. Da Silva and W. F. Salas, "Modelling the correlation between cutting and process parameters in high-speed machining of Inconel 718alloy using artificial neural network". *International Journal of Machine Tools & Manufacture*, Vol. 45. pp. 1375-1385. 2005.
- [6] N. Friedman, D. Geiger, M. Goldszmit, "Bayesian network classifiers". *Machine Learning*. Vol. 29. pp. 131-161. 1997.
- [7] W. Hao, X. Zhu, X. Li and G. Turyayagyenda, "Prediction of cutting force for self-propelled rotary tool using artificial neural network". *Journal of Material processing Technology*. Vol. 180. pp. 23-29. 2006.
- [8] J. Ko, Tae1, P. Jung, K. Hee, K. Sun, "On-machine measurement using a noncontact sensor based on a CAD model". *International Journal of Advanced Manufacturing Technology*. Vol. 32. pp. 739-746. April 2007.
- [9] B. Lela, D. Bajie, S. Jozié, "Regression analysis, support vector machines, and Bayesian neural Network approaches to modelling surface roughness in face milling". P. Gil. *International Journal of Advanced Manufacturing Technology* (0268-3768) 42, pp. 1082-1088. 2009.
- [10] J. MacQueen "Some Methods for Classification Analysis of Multivariate Observations" *Proc. 5th Berkeley Symp. Math. Statistics and Probability* (1). pp. 281-297. 2003.
- [11] T. Ozel, A. Esteves, J. Davim "Neural network process modelling for turning of steel parts using conventional and wiper inserts" *International Journal of Materials and Product Technology*. Vol. 35. pp. 246 – 258. 2009.
- [12] M. Ramírez-Cadena, M. Correa, C. Rodríguez-Gonzalez and J.R. Alique, "Surface Roughness Modelling Based on Surface Roughness Feature Concept for High Speed Machining". *ASME Int. Mechanical Engineering*. 2005.
- [13] B. Reddy, G. Padmanabhan G. and K. V. Reddy, "Surface Roughness Prediction Techniques for CNC turning", *Asian Journal of Scientific Research*, Vol. 1, Number 3, pp. 256-264. 2008.
- [14] H Soleimanimehr, MJ Nategh, S Amini, "Modelling of Surface Roughness in Vibration Cutting by Artificial Neural Network". *Proceedings of World Academy Of Science, Engineering And Technology* Vol. 40. pp. 386-390. April 2009.
- [15] M. Stone, "Cross-validatory choice and assessment of statistical prediction". *Journal of the Royal Statistical Society B*. Vol. 36. pp. 111-147. 1974.
- [16] L. Zhou, K. Cheng, "Dynamic cutting process modelling and its impact on the generation of surface topography and texture in nano/micro cutting". In *Proc of IMechE-2009*. 2009.
- [17] U. Zuperl, F. Cus, "Optimization of cutting conditions during cutting by using neural networks", *Robot. Comput. Integr. Manuf.* Vol. 19. pp. 189-199. 2003.

APLICACIÓN DE TÉCNICAS DE SOFT COMPUTING PARA LA ACCELERACIÓN DE LA TRANSCODIFICACIÓN DE VÍDEO

C. Solana-Ciprés¹, G. Fernández-Escribano², L. Rodríguez-Benítez¹,
J. Moreno-García³ y L. Jiménez-Linares¹

¹ Grupo de Investigación ORETO, Universidad de Castilla-La Mancha, Departamento de Tecnologías y Sistemas de Información, Paseo de la Universidad s/n, 13071 Ciudad Real, España, {CayetanoJ.Solana, Luis.Rodriguez, Luis.Jimenez}@uclm.es

² Instituto de Investigación en Informática de Albacete, Avenida de España s/n, 02071 Albacete, España, Gerardo.Fernandez@uclm.es

³ Escuela Ingeniería Técnica Industrial, Avenida. Carlos III s/n, 45071 Toledo, España. Juan.Moreno@uclm.es

Resumen

En este artículo se presentará el trabajo realizado en el área de la transcodificación de vídeo empleando técnicas de Minería de Datos. Se comentarán los algoritmos utilizados para la creación de árboles de decisión capaces de solucionar uno de los principales problemas que acarrea este tipo de sistemas: el elevado coste computacional. Por último, se plantea la posibilidad de emplear técnicas de Soft-Computing, concretamente Lógica Difusa, para solucionar dicho problema con el fin de integrar el transcodificador difuso en un sistema completamente difuso capaz de capturar, procesar, segmentar e interpretar escenas de tráfico de forma inteligente.

Palabras Clave: Transcodificación de Vídeo, Minería de Datos, Árboles de Decisión y Lógica Difusa.

1 INTRODUCCIÓN

Las condiciones de la red de redes (Internet) cambian dinámicamente. Los recursos disponibles en un instante pueden aumentar o disminuir al instante siguiente. Y no sólo los recursos de la red pueden verse alterados; también es posible que el usuario disponga de aparatos de recepción de distintos tipos, como PDA o móviles, y desee determinadas prestaciones en cada momento. Este problema admite dos soluciones: almacenar las secuencias de vídeo codificándolas de diferente forma para que, dependiendo del estado de la red o las características del dispositivo de recepción, se seleccione aquella que permita su transmisión sin problemas o realizar un procesamiento de la secuencia para adaptarla a las nuevas condiciones demandadas por el usuario. La primera solución resulta prácticamente inviable debido a que las empresas deberían mantener una gran cantidad de información replicada, la cual pudiera llegar a no usarse nunca. La segunda solución dio origen a lo que actualmente se conoce como Transcodificación de Vídeo.

La *Transcodificación de Vídeo* es el proceso mediante el cual una secuencia de vídeo almacenada bajo cierto patrón de codificación (estándar de vídeo) se convierte en otra secuencia, la cual, manteniendo íntegra la semántica de su contenido, puede verse alterada en cualquiera de los parámetros que son definibles durante el proceso de codificación, contemplándose además el cambio de estándar. Los dispositivos que realizan esta conversión se denominan *transcodificadores* o *transcoders* y pueden ser hardware o software, aunque en un sistema muy especializado y que requiera una gran velocidad de respuesta son implementados mediante hardware.

La transcodificación de vídeo se encuentra actualmente en expansión debido a que el número de aplicaciones multimedia que hacen uso de esta técnica crece rápidamente. Desde el punto de vista del usuario, la arquitectura subyacente en este tipo de aplicaciones es totalmente desconocida; lo más importante es que el servicio que le presta alcance la calidad deseable. Por tanto, este tipo de sistemas han de ser totalmente transparentes de cara al usuario. El principal problema que presentan las técnicas de transcodificación de vídeo, si su diseño no es el adecuado, es su elevado coste computacional para aplicaciones que demandan requisitos de tiempo real. Este alto coste computacional puede ser reducido eficientemente reutilizando la información obtenida durante la primera etapa del transcodificador, encargada de decodificar la secuencia codificada en el estándar de partida. Sin embargo, las grandes diferencias que normalmente existen entre los estándares que precisan la transcodificación en tiempo real hacen que esta tarea sea más compleja que la simple concatenación de un decodificador seguido de un codificador (Arquitectura de Referencia, Figura 1).

Por otro lado, la inminente aparición de los sistemas basados en codificación escalable (*Scalable Video Coding* o SVC) no resulta un impedimento para continuar realizando trabajos en la línea de transcodificación de vídeo, ya que los escenarios de trabajo de unos sistemas frente a los otros son, al menos en principio, diferentes. Existe a día de hoy una gran cantidad de material de vídeo

codificado mediante estándares clásicos, como el MPEG-2, con necesidades reales de una transcodificación previa al envío de dicho material a través de las redes de comunicación. En SVC el principio de funcionamiento es distinto. El formato de codificación es independiente de las necesidades de transmisión, ya que éste se basa en realizar una codificación mediante un estándar que tenga en cuenta un posible truncamiento del flujo de salida en niveles, para el control de la calidad del vídeo enviado.

Por estas razones, la realización de trabajos en la línea de la transcodificación de vídeo contará con un gran apoyo por parte de las empresas que necesitan resolver este problema para poder atender a un mayor número de clientes sin que estos perciban una disminución de las prestaciones y servicios que, hasta ese momento, recibían.

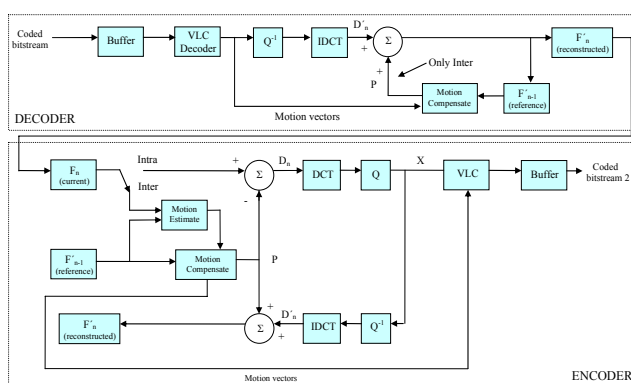


Figura 1: Arquitectura de Referencia.

El resto del artículo está organizado de la forma que a continuación se describe. La Sección II relata brevemente el funcionamiento de la estimación y compensación de movimiento en los estándares de codificación de vídeo de arquitectura asimétrica clásica (codificador complejo, decodificador sencillo); tales como el MPEG-2, el H.263, y el recientemente estandarizado H.264/AVC. En la Sección III se resume el trabajo realizado hasta la fecha en el área de la transcodificación de vídeo basada en la Minería de Datos y el empleo de árboles de decisión con el objetivo de reducir el tiempo de re-codificación de una secuencia de vídeo. Finalmente, la Sección IV motiva la posibilidad de emplear lógica difusa para el diseño de un transcodificador de vídeo MPEG-2/H.264 como alternativa a la Minería de Datos. Se plantea la integración del mismo en un sistema de procesamiento inteligente de secuencias de vídeo compuesto de diferentes etapas: captación, procesamiento, clasificación y finalmente toma de decisiones.

2 ESTÁNDARES DE CODIFICACIÓN DE VÍDEO

Los estándares de compresión de vídeo definen de forma rigurosa y detallada tanto la sintaxis de la señal de vídeo comprimida como el proceso de decodificación de la misma, si bien se deja libertad a la hora de implementar el

codificador. El codificador debe generar flujos de vídeo comprimido con sintaxis válida y perfectamente interpretable por parte del decodificador. Las operaciones que se realicen con el fin de acelerar el proceso de codificación siempre serán bienvenidas, en tanto que no modifiquen el formato de salida del vídeo comprimido. Las técnicas de compresión de vídeo emplean mecanismos para reducir el tamaño de las secuencias de vídeo intentando aprovechar la redundancia espacial dentro de una misma imagen y la redundancia temporal entre imágenes sucesivas. Estos son los dos grandes pilares en los que se basan la mayoría de los estándares o formatos de compresión de vídeo.

2.1. Estimación y Compensación de Movimiento

Entrando en detalle, un vídeo digital está compuesto por una serie de imágenes en movimiento. Sin considerar ninguna técnica de compresión, éste podría verse como una sucesión de imágenes mostrada a una velocidad tal que produce en nuestro sistema visual la sensación de movimiento. Las técnicas de compresión de vídeo emplean mecanismos para reducir el tamaño de las secuencias de vídeo e intentan aprovechar la redundancia espacial dentro de una misma imagen y la redundancia temporal entre imágenes sucesivas. Estos son los dos grandes pilares en los que se basan la mayoría de los estándares o formatos de compresión de vídeo.

En concreto, las técnicas de estimación y compensación de movimiento son el método con el que se pretende eliminar la redundancia temporal mediante la evaluación de un modo eficiente del movimiento de los objetos entre imágenes. En vez de codificar la imagen completa (Figura 2), se codificará la diferencia entre la imagen o imágenes tomadas como referencias (pasadas, futuras o ambas) y la imagen actual (compensación del movimiento). A esta diferencia se conoce como residuo del movimiento compensado. Este residuo habitualmente contendrá muchos valores iguales o próximos a cero, los cuales pueden ser codificados empleando pocos bits (Figura 3). La diferencia entre imágenes se realizará mediante la comparación de bloques de píxeles de diferente tamaño, dependiendo del estándar de codificación o de las opciones de codificación dentro del mismo. La unidad básica de comparación es el Macrobloque (MB), bloque de 16x16 píxeles. A menor tamaño de bloque, mayor tiempo de cómputo, pero también mejores prestaciones. La estimación y compensación de movimiento en MPEG-2 puede realizarse en bloques de tamaño 16x16 o 16x8. Sin embargo, en el recientemente estandarizado H.264 [10], la estimación y compensación de movimiento puede realizarse en bloques de píxeles que van desde los 16x16 hasta los 4x4 píxeles.

El análisis y el estudio de la correlación existente entre el residuo de movimiento compensado, junto con los diferentes tamaños o modos de decisión empleados para la realización de la estimación y compensación de

movimiento en un estándar de vídeo (codificación clásica) o entre varios (transcodificación), dará lugar a codificadores y transcodificadores de baja complejidad. Es más, debido a que la estimación y compensación de movimiento es con mucho el proceso más costoso en este tipo sistemas, toda investigación en este área mejorará las prestaciones de dispositivos que, hasta la fecha, no podían plantearse la codificación de vídeo empleando estándares de última generación (como el H.264) debido a las necesidades de cálculo que implica la ejecución de complejos mecanismos de codificación.



Figura 2: Segunda imagen de la secuencia Flower Garden.

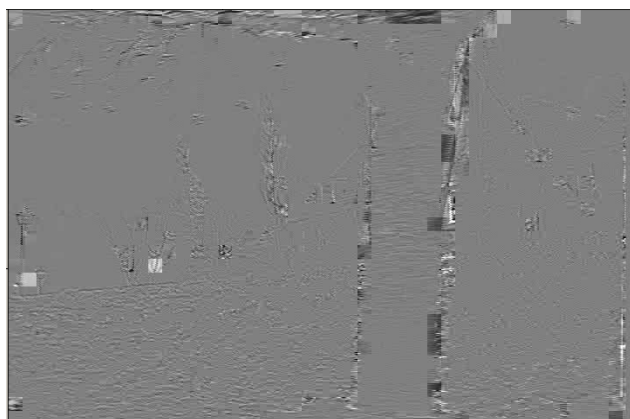


Figura 3: Residuo MPEG-2 de la imagen anterior (+128).

2.1. MPEG-2

MPEG-2 [9] es el estándar de compresión de vídeo más ampliamente utilizado en la actualidad y sobre el que se ha invertido la mayor cantidad de dinero en infraestructura. Entre sus aplicaciones destacan: el *Digital Versatile Disk* (DVD), la televisión por cable, la televisión por satélite, la *Televisión Digital Terrestre* (TDT) y el vídeo bajo demanda. Actualmente es la norma empleada en la televisión de alta definición (*High Definition TeleVision* o HDTV). Permite realizar la estimación y compensación de movimiento en bloques de tamaños 16x16 u 8x8 píxeles.

2.3. H.264/AVC

H.264/AVC [10] surgió con el objetivo de crear un estándar que fuera capaz de proporcionar una buena calidad reduciendo notablemente el flujo de bits de salida de la secuencia de vídeo codificada con respecto a otros estándares previos tales como MPEG-2. Además, se pretende conseguir esto sin incrementar la complejidad evitando convertir el estándar en imposible de implementar. Como objetivo adicional se fijó que el nuevo estándar pudiera ser aplicado a una gran variedad de aplicaciones como el almacenamiento en DVD, vídeo conferencia, televisión por cable, aplicaciones de bajo caudal, media y alta definición en televisión, flujos de vídeo a través de Internet, etc.

3 APRENDIZAJE AUTOMÁTICO APLICADO A LA TRANSCODIFICACIÓN DE VÍDEO

El aprendizaje automático se refiere al estudio de algoritmos y sistemas que *aprenden* o adquieren conocimiento a partir de experiencias. Mediante el análisis y el estudio de las mismas, este tipo de técnicas aplican una serie de estadísticas con el fin de clasificar los datos en una serie de experiencias *tipo* que serán empleadas como referencia a la hora de clasificar nuevos casos. Hasta el momento, el aprendizaje automático ha sido usado en un amplio abanico de aplicaciones, entre las que se incluyen las herramientas de búsqueda, el diagnóstico médico, el análisis de los mercados, la clasificación de secuencias de ADN, el reconocimiento de la escritura y del habla, el reconocimiento de objetos en visión artificial, los videojuegos y el movimiento de robots, entre muchas otras.

Las técnicas de aprendizaje automático que hemos venido usando para acelerar la transcodificación de vídeo están basadas en aprendizaje supervisado, es decir, se emplean ejemplos (conjunto de datos de entrenamiento), de los cuales se conoce su pertenencia a una clase en concreto, para crear reglas de pertenencia a dicha clase. Estas reglas se emplearán en la construcción de árboles de decisión. El conjunto de datos de entrenamiento deberá cumplir una serie de propiedades [8]:

1. Cada atributo o variable puede tomar valores continuos o discretos, pero el número de atributos no puede variar de una muestra a otra. Es decir, todos los casos del conjunto de datos de entrenamiento deben tener el mismo número de variables.
2. El número de categorías en las que es posible clasificar los casos debe ser conocido a priori, al tratarse de aprendizaje supervisado.
3. El número de categorías debe ser finito y disjunto entre ellas.

4. El número de muestras disponibles debe de ser lo suficientemente numeroso para que sea posible la generalización.

3.2. Transcodificación MPEG-2/H.264

El primer transcodificador de vídeo con el que se trabajó estaba basado en un decodificador MPEG-2 seguido de un codificador de H.264. En concreto, se emplearon los árboles de decisión para determinar cuál debía ser el modo de codificación a emplear en un MB en las imágenes de tipo P, en la codificación de la secuencia de vídeo en H.264/AVC, basándose en la información obtenida durante la decodificación de MPEG-2.

La secuencia de vídeo de entrada es decodificada mediante un decodificador de MPEG-2. Durante dicho proceso se almacena información relativa al mismo: el modo de codificación del MB y el patrón de codificación (CBPC), así como la información de residuo de movimiento compensado. En primera instancia, una secuencia de vídeo se emplea como secuencia de entrenamiento. Se trata de la parte más crítica de todo el proceso, ya que la misma debe ser lo suficientemente representativa como para permitir la generalización de los árboles de decisión. Dicha secuencia de entrenamiento será posteriormente codificada mediante H.264/AVC, almacenando la información relativa al modo de codificación empleado para cada MB. La elección del modo de codificación para cada uno de ellos será la variable clase. Con toda esta información, se generarán los árboles de decisión. La Figura 4 describe el proceso.

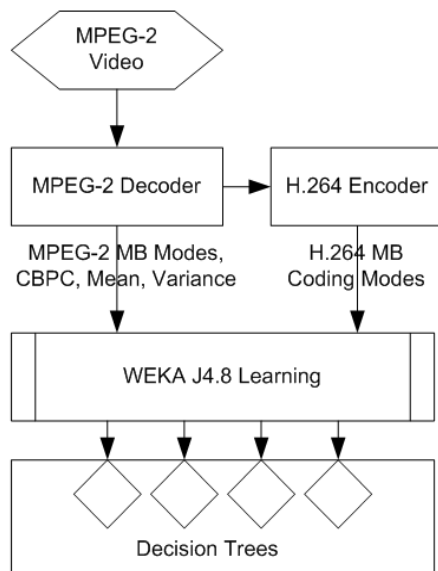


Figura 4: Proceso de creación de los árboles de decisión.

Después de estudiar los árboles de decisión generados, se comprobó que se establecía una correspondencia entre los modos más sencillos de MPEG-2 y los de H.264; seleccionando modos más complejos en éste último cuando el MB procedente de MPEG-2 presentaba una alta

cantidad de movimiento (medida en términos de residuo, media y varianza), y patrones complejos de codificación de MPEG-2 (CBPC y modos MPEG-2). En La herramienta empleada para la construcción de los árboles de decisión fue la aplicación de libre distribución WEKA [8]. Hasta la fecha, dos de los algoritmos que incluye han sido probados y testeados con muy buenos resultados:

1. El algoritmo J48. Implementación del algoritmo C4.5 propuesto por Ross Quinlan [12].
2. El algoritmo JRip. Implementación del algoritmo RIPPER, propuesto por William Cohen [14].

Una vez obtenido dicho árbol se pasará a testear el mismo con otras secuencias de vídeo distintas de la empleada como secuencia de entrenamiento, lo que permitirá validar el modelo propuesto. Un ejemplo del funcionamiento del mismo se muestra en las Figuras 5 y 6, en las que se compara la selección realizada por el software de referencia de H.264/AVC, y el propuesto por el algoritmo de selección del modo de decisión, respectivamente. El código de colores que se ha empleado en dichas figuras es Rojo para la codificación Intra, Azul para la codificación Inter, y Verde para el caso en el que no es necesario aplicar ningún tipo de codificación. El lector interesado puede consultar [7] si desea profundizar más sobre los modos de decisión en el estándar H.264/AVC. La secuencia elegida para el testeo del modelo propuesto fue Foreman, mientras que la secuencia de entrenamiento empleada fue Flower Garden. De esta forma, es posible reducir los tiempos del transcodificador de referencia en torno a un 85 % en media (Tabla 1), haciendo factible el uso de los mismos en escenarios de tiempo real. Algunas de las publicaciones más relevantes realizadas empleando este tipo de transcodificadores, y que el lector interesado puede consultar son [5] [6] [15]. En la Tabla 1 se muestran los resultados obtenidos en cuanto a diferencias incrementales de tiempo, calidad (PSNR) y caudal (Bitrate) mediante un Transcodificador MPEG-2/H.264 basado en el algoritmo JRIP [14], comparado frente a la arquitectura MPEG-2/H.264 de referencia (decodificador/codificador).

Tabla 1: Resultados de Transcodificación MPEG-2/H.264 para secuencias de vídeo VGA (640x480 píxeles).

Secuencia y número de imágenes codificadas		Δ Bitrate, Δ PSNR y Reducción Temporal (en media)		
		Transcodificador propuesto MPEG-2/H.264		
		Tiempo (%)	Δ PSNR (dB)	Δ Bitrate (%)
Fast Food	220	- 73.02	- 0.086	3.67
Football	260	- 70.84	- 0.066	3.27
Harp	220	- 87.91	- 0.057	2.61
Parade	220	- 93.01	- 0.029	1.41
Sgi-ant	220	- 91.57	- 0.186	6.29
Ship	260	- 91.90	- 0.017	0.72
Skyline	260	- 87.30	- 0.077	2.98
Soft Football	220	- 73.25	- 0.058	3.00
Suzie	260	- 87.57	- 0.045	1.90
Tempete	260	- 90.77	- 0.067	3.18
Waterfall	260	- 91.85	- 0.019	0.78
Average	2660	- 85.36	- 0.064	2.71

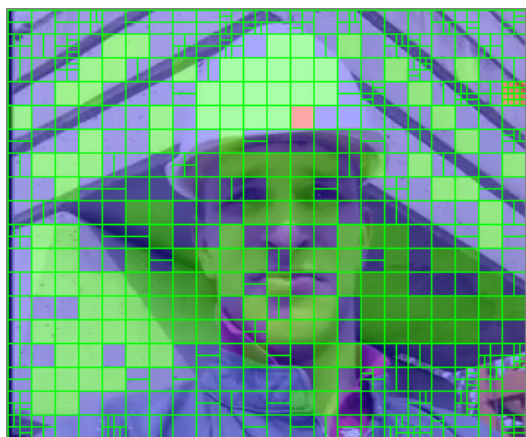


Figura 5: Proceso de codificación realizado por el estándar H.264/AVC.

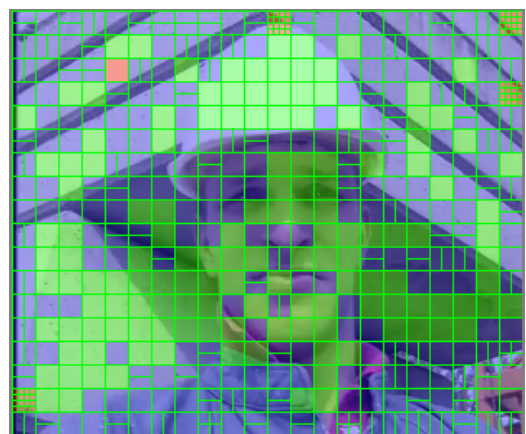


Figura 6: Proceso de codificación realizado por el estándar H.264/AVC y supervisado mediante aprendizaje automático.

4 TRANSCODIFICACIÓN DIFUSA ORIENTADA A UNA APLICACIÓN DE VÍDEO-VIGILANCIA

En trabajos anteriores se ha desarrollado un algoritmo de segmentación en tiempo real de objetos móviles para video H.264/AVC [2]. Dicho trabajo está orientado a video-vigilancia, por lo que se impone el tiempo real y la probabilidad de detección como dos factores determinantes a la hora de evaluar su idoneidad. Además, el algoritmo de segmentación sirve como soporte a otras herramientas de clasificación de objetos [3] e interpretación de comportamientos [11] recientemente desarrolladas. La Figura 7 muestra, a modo de resumen, el sistema propuesto y su división en las diferentes etapas abordadas o pendientes de abordar.

No obstante, a día de hoy es reducido el catálogo de cámaras de seguridad en H.264 y, de hecho, las existentes no permiten aprovechar las características del estándar. Además, desde el punto de vista económico, supondría un coste adicional la actualización de las cámaras de un sistema de vídeo-vigilancia en explotación para adaptarlo al algoritmo de segmentación en H.264. Por estos motivos, se plantea la posibilidad de diseñar un

transcodificador difuso de video MPEG-2/H.264 que cumpla los siguientes requisitos:

- a) Diseñar el transcodificador orientado a la optimización del factor temporal, es decir, lograr cotas máximas de velocidad en perjuicio de un mayor bit rate, dado que el flujo resultante no se va a transmitir, y de un menor PSNR, dado que la calidad del video no influye en el resultado de la segmentación y además permitiría preservar la privacidad de las personas detectadas.

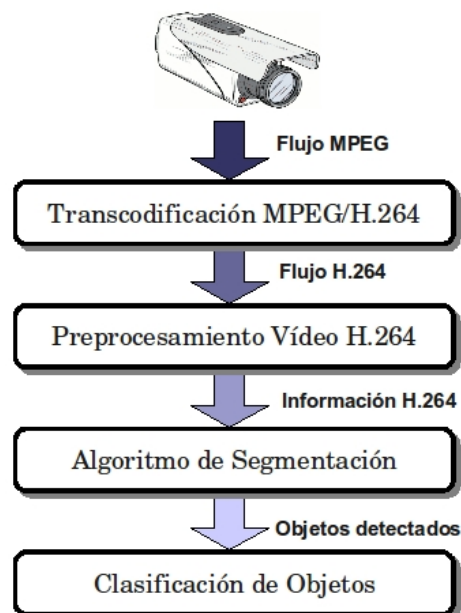


Figura 7: Sistema propuesto basado en Soft-Computing.

- b) Las técnicas de Soft-Computing permiten acelerar el proceso de transcodificación mediante la restricción de los modos de decisión en los que se puede codificar cada macrobloque. Métodos como los árboles de decisión o los conjuntos de reglas permiten obtener resultados aceptables consumiendo poco tiempo de procesamiento.

- c) Puesto que el algoritmo de segmentación está basado en movimiento, explota el campo de vectores de movimiento del vídeo sin utilizar información a nivel de píxel. Esta característica permite diseñar un transcodificador dedicado de mayor velocidad.

- d) En la actualidad, el algoritmo de segmentación analiza todas las imágenes, es decir, entre 25 y 30 imágenes por segundo en función del vídeo. Sin embargo, no es necesario analizar todas las imágenes para detectar los objetos en movimiento e inferir sus trayectorias. Por tanto, un transcodificador dedicado permite seleccionar qué conjunto de imágenes son más interesantes y cuales pueden ser consideradas espurias. Esta decisión permite desechar aquellas que no tengan apenas movimiento, dado que no aportarán objetos móviles, y otras con mucho movimiento ruidoso porque dificultará la extracción de los objetos móviles; por ejemplo, en caso de hojas de árboles o del oleaje del agua.

Por tanto, la línea de trabajo se centra en el diseño e implementación de un transcodificador difuso de tiempo real MPEG-2/H.264 que cumpla los requisitos expuestos y pueda ser integrado con la arquitectura de segmentación diseñada, y con el resto de módulos de mayor nivel. Dicho transcodificador optimizará el criterio tiempo sin considerar en demasía la calidad objetiva (PSNR) y subjetiva (SSIM) y el caudal (bit rate).

De esta forma, el transcodificador difuso MPEG-2/H.264 habrá de elegir el modo de decisión óptimo para codificar cada macrobloque a partir de las variables de entrada procedentes de la información obtenida y post-procesada de MPEG-2: tipo de MB, tipo de codificación, cantidad de residuo de movimiento compensado, vectores de movimiento, etc. La Figura 7 muestra, a modo de resumen, el sistema propuesto y su división en las diferentes etapas que se pretenden abordar.

5 CONCLUSIONES

En este artículo se ha presentado un resumen del trabajo realizado hasta la fecha en el área de la transcodificación de vídeo, en concreto, el realizado con transcodificadores MPEG-2/H.264. Por otro lado, se plantea realizar un transcodificador difuso entre ambos estándares con el fin de ser integrado en un sistema para la clasificación de objetos. Por último, se plantea la posibilidad de emplear el sistema difuso ADRI [13] como herramienta de selección de variables previa al trabajo mediante WEKA.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Ciencia y Tecnología de España bajo el Programa CONSOLIDER INGENIO CSD2006-46 y el proyecto CICYT TIN2006-15516-C04-02; por la Consejería de Ciencia y Tecnología de Castilla-La Mancha bajo los proyectos PEII09-0037-2328, PII2I09-0045-9916, PII1C09-0137-6488 y FEDER; y por la Universidad de Castilla-la Mancha bajo el proyecto TC20091095.

Referencias

- [1] A. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder". *IEEE Transactions on Information Theory*, vol. IT-22, pp. 1-10, January 1976.
- [2] C.J. Solana-Cipres, G. Fernandez-Escribano, L. Rodriguez-Benitez, J. Moreno, L. Jimenez, "Real-time moving object segmentation in H.264 compressed domain based on approximate reasoning". *International Journal of Approximate Reasoning*, DOI:10.1016/j.ijar.2009.09.002, 2009.
- [3] C.J. Solana-Cipres, J.A. Albusac, J.J. Castro-Schez, L. Rodriguez-Benitez, "Automatic Object Labelling for Monitored Environments using Clustering Techniques", *International Conference on Imaging for Crime Detection and Prevention*, in press, London, 2009.
- [4] D. Slepian, J.K. Wolf, "Noiseless coding of correlated information sources". *IEEE Transactions on Information Theory*, vol. IT-19, pp. 471-480, July 1973.
- [5] G. Fernández-Escribano, H. Kalva, P. Cuenca, L. Orozco-Barbosa and A. Garrido, "A Fast MB Mode Decision Algorithm for MPEG-2 to H.264 P-Frame Transcoding". *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 2, pp. 172-185, February 2008.
- [6] G. Fernández-Escribano, J. Bialkowski, J.A. Gámez, H. Kalva, P. Cuenca, L. Orozco-Barbosa and A. Kaup, "Low-Complexity Heterogeneous Video Transcoding Using Data Mining". *IEEE Transactions on Multimedia*, vol. 10, no. 2, pp. 286-299, February 2008.
- [7] I. E. G. Richardson, "H.264 and MPEG-4 Video Compression", John Wiley & Sons Ltd, New Jersey, 2003.
- [8] I.H. Witten and E. Frank. "Data Mining: Practical Machine Learning Tools and Techniques". 2nd Edition, Morgan Kaufmann, San Francisco, 2005.
- [9] ISO/IEC JTC1 SC29 WG11 13818 "Generic Coding of Moving Pictures and Associated Audio". Part1: Systems; Part 2: Video; Part 3: Audio, ISO, 1994.
- [10] ISO/IEC 14496-10 and ITU-T Rec. H.264, Advanced Video Coding, 2003.
- [11] J. Albusac, J.J. Castro-Sánchez, L. Lopez-Lopez, D. Vallejo, L. Jimenez-Linares. A supervised learning approach to automate the acquisition of knowledge in surveillance systems. *Journal of Signal Processing*, vol. 89, no. 12, pp. 2400-2414, 2009.
- [12] J.R. Quinlan. "C4.5: Programs for Machine Learning". Morgan Kaufmann, 1993.
- [13] M. Delgado, A.F. Skarmeta, L. Jimenez, "A regression methodology to induce a fuzzy Model". *International Journal of Intelligent Systems*, vol. 16, nº. 2, pp. 169-190, 2001.
- [14] W. W. Cohen and Y. Singer, "A simple, fast, and effective rule learner", in *Proceedings of the Sixteenth National Conference on Artificial Intelligence 1999*, pp. 335-342, Orlando (Florida), USA, July 18-22, 1999.
- [15] J. Xin, J. Li, A. Vetro, H. Sun y S. Sekiguchi., "Motion Mapping for MPEG-2 to H.264/AVC Transcoding", *Multimedia Tools and Applications*, vol. 35, no. 2, pp. 203-223, Noviembre 2007.

APLICACIÓN DE XFUZZY 3 AL PROCESADO DE IMÁGENES BASADO EN REGLAS

Iluminada Baturone Piedad Brox Rosario Arjona

Instituto de Microelectrónica de Sevilla, (IMSE-CNM-CSIC) y Dpto. Electrónica y Electromagnetismo,
Universidad de Sevilla (España); {lumi, brox, arjona}@imse-cnm.csic.es

Resumen

Los entornos de desarrollo de sistemas fuzzy se han empleado normalmente para diseñar sistemas de control y de toma de decisiones pero apenas para diseñar sistemas de procesamiento de imágenes, a pesar de que este campo cuenta ya con numerosas soluciones basadas en Lógica Fuzzy. En este artículo se muestra cómo el entorno Xfuzzy 3 desarrollado en el Instituto de Microelectrónica de Sevilla posee la versatilidad necesaria para abordar el diseño de estos sistemas, facilitando su descripción, verificación, ajuste y síntesis.

Palabras Clave: Sistemas fuzzy, procesamiento de imágenes, herramientas de CAD.

1 INTRODUCCIÓN

Al poco tiempo de introducirse la Lógica Fuzzy en 1965, se presentó el fundamento del control fuzzy en 1972 [4] y la primera realización práctica en 1976 [10]. Desde entonces, las aplicaciones en control han sido las más numerosas y las que más atención han recibido a nivel académico y comercial. Este enorme interés ha dado lugar al desarrollo de distintos entornos software especialmente orientados al diseño de controladores difusos (como FIDE de Apronix, Inc., rFLASH, de Rigel Corporation y FuzzyTECH, de Inform).

Más tarde, las ventajas de la Lógica Fuzzy empezaron a explotarse en otros campos de aplicación. En relación al procesamiento de imágenes, los primeros trabajos se publican en los 80 [14]. La imprecisión y ambigüedad inherentes a las imágenes las hacen adecuadas para su procesamiento mediante Lógica Fuzzy. Estas características se dan a bajo nivel debido al ruido y la posible baja resolución y calidad de las imágenes. También se dan a alto nivel porque las

fronteras de los objetos y regiones de una imagen así como sus distancias, posiciones relativas, etc. se pueden modelar con conceptos fuzzy. Una línea de trabajo seguida por muchos autores ha sido desarrollar una morfología matemática fuzzy como extensión de la morfología clásica, definiendo nuevos operadores morfológicos fuzzy [13]. Otra línea con gran aceptación ha sido emplear reglas fuzzy basadas en conocimiento heurístico. Varios autores han propuesto bases de reglas para el suavizado, aumento de contraste, extracción de bordes y filtrado [16]. También a alto nivel se han empleado reglas fuzzy para reconocer y localizar objetos en aplicaciones de visión por ordenador o por robot [6]. Sin embargo, pocos entornos software para Lógica Fuzzy han sido empleados en estas aplicaciones. Específicamente para el procesamiento de imágenes tan sólo encontramos en la literatura la herramienta propuesta por Russo, que se aplica a un problema de extracción de bordes [15].

El entorno Xfuzzy 3¹ desarrollado en el Instituto de Microelectrónica de Sevilla para el diseño de sistemas fuzzy ha sido empleado típicamente para resolver problemas de control como la mayoría de los entornos [1]. Sin embargo, la versatilidad que le proporciona su lenguaje de especificación XFL3, que soporta reglas cercanas a las expresadas en lenguaje natural, con operadores que pueden ser definidos libremente por el usuario y estructuras jerárquicas con módulos fuzzy y no fuzzy, permite la utilización del entorno en muchos otros campos de aplicación, en particular en el del procesamiento de imágenes.

En este artículo se presenta cómo Xfuzzy 3 permite el diseño de sistemas de procesamiento de imágenes basados en reglas. En la Sección 2 se resumen las características y herramientas que facilitan dicho diseño. En la Sección 3 se ilustran 3 ejemplos de aplicación en: (a) extracción de bordes, (b) aumento de resolución y (c) desentrelazado de

¹ <https://forja.rediris.es/projects/xfuzzy>
<http://www.imse-cnm.csic.es/Xfuzzy>

secuencias de vídeo. Las conclusiones se indican en la Sección 4.

2 XFUZZY 3 PARA PROCESADO DE IMÁGENES

El entorno Xfuzzy 3 posee una serie de herramientas de CAD que comparten un lenguaje común de especificación de sistemas fuzzy denominado XFL3 [12]. Una de las características de este lenguaje es que separa la descripción de una base de reglas de la definición de los operadores que intervienen en ella. Esto significa que el cálculo de los grados de pertenencia (definición de las funciones de pertenencia), el de los grados de activación de las reglas (definición de las conectivas, de los operadores complemento y modificadores lingüísticos) y el cálculo de la salida no fuzzy (definición del método de concreción) se hace en base a la elección de funciones matemáticas definidas en ficheros externos al que define la estructura de la base de reglas. Estos ficheros, denominados “paquetes”, pueden ser definidos libremente por el usuario, lo cual es una gran ventaja para extender la aplicabilidad de Xfuzzy porque según el campo de aplicación los operadores pueden ser diferentes.

En el caso del procesamiento de imágenes, son varios los trabajos en que aparecen operadores particulares que no son habituales en control. Así por ejemplo, en [8] se utiliza el modificador lingüístico “la mayoría de”, empleándose una función particular para definirlo, y en [7] se define un operador de “intensificación” y de concreción particulares. Otra diferencia con respecto al control es que el número de entradas que se contempla en las bases de reglas suele ser más elevado por lo que suelen especificarse sólo algunos casos posibles y el resto se engloba en una regla del tipo “else”, cuyo grado de activación se calcula de forma diferente según los autores [2, 8, 16].

Para estos casos, el usuario de Xfuzzy 3 puede definir paquetes específicos con funciones particulares de tipo: (a) binarias (operando sobre dos argumentos, como t-normas y s-normas), (b) unarias (sobre un argumento, como c-normas y modificadores lingüísticos), (c) funciones de pertenencia y (d) métodos de concreción. La definición de una función incluye su nombre (y alias posibles), los parámetros que especifican su comportamiento (así como posibles restricciones sobre estos parámetros), la descripción de su comportamiento en Java (y en C y C++ si se quiere generar síntesis a esos lenguajes) e incluso la descripción de sus derivadas si se quiere emplear esta función en métodos de aprendizaje guiados por gradiente. El usuario puede introducir toda esta información de forma gráfica mediante la herramienta *xfpkg* (Figura 1). Con esta utilidad se genera automáticamente una clase Java que incorpora todas las

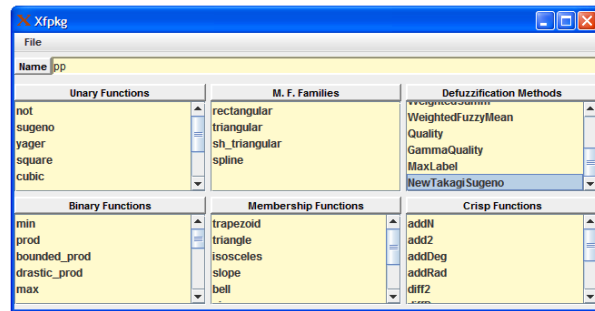


Figura 1: Ventana principal de la herramienta *xfpkg*.

características de la función y que puede emplearse por cualquier especificación XFL3.

Por otro lado, el procesamiento de imágenes suele encadenar razonamientos. Por ejemplo, en [9] se emplea una base de reglas fuzzy para seleccionar los parámetros de un filtro gaussiano (no fuzzy) y en [5] la inferencia fuzzy determina el tipo de filtro (fuzzy o no) a seleccionar.

Para estos casos, el usuario de Xfuzzy 3 puede definir los distintos módulos constitutivos tanto fuzzy como no fuzzy (o *crisp*) y conectarlos con la jerarquía adecuada. Esto lo puede hacer de forma gráfica mediante la herramienta *xfedit*. En la ventana principal de esta herramienta, el usuario puede definir las distintas bases de reglas o módulos fuzzy (cada una con sus funciones de pertenencia y operadores propios) así como seleccionar los distintos módulos crisp. Los módulos crisp se definen en Java en un paquete que deberá estar cargado en el entorno (para esta definición de módulos crisp también es muy útil la herramienta *xfpkg* comentada anteriormente). Pueden definirse para aplicar cualquier función matemática sobre sus entradas. Otra versatilidad interesante que permite XFL3 es que, a la hora de interconectar módulos fuzzy, se pueden intercambiar valores tanto crisp como fuzzy, según que se empleen o no métodos de concreción antes de intercambiar los valores.

Para analizar cómo el sistema definido procesa las imágenes, el usuario puede emplear la herramienta de simulación *xfsim*. Esta herramienta conecta el sistema con un modelo del contexto de operación (una planta, en la nomenclatura de control), que el usuario definirá como una clase Java. En el caso del procesamiento de imágenes, esta clase Java generará las entradas al procesamiento a partir de una(s) imagen(es) dada(s) y tomará las salidas del procesamiento para generar la(s) imagen(es) resultante(s) y los ficheros de datos (*log*) necesarios.

Si en la simulación no se obtienen buenos resultados, suele ser muy útil la herramienta *xfmt*, que permite monitorizar las entradas y salidas de cada módulo del sistema de procesamiento visualizando las funciones de pertenencia en cada nivel y los grados de activación de las

reglas. Así puede detectarse si se ha cometido algún error en la traducción a reglas del conocimiento heurístico.

Para mejorar de forma automática los resultados de un sistema de procesamiento se pueden aplicar métodos de aprendizaje que ajusten los parámetros del sistema. Para ello puede emplearse la herramienta *xfsl*, que permite aplicar una gran variedad de algoritmos de aprendizaje supervisado. Incluso si no se dispone de conocimiento heurístico para definir el sistema pero sí de datos numéricos, se puede emplear la herramienta *xfdm* para extraer automáticamente una base de reglas. En este caso, suele ser útil emplear a continuación la herramienta *xfsp*, que permite simplificar las funciones de pertenencia y/o las reglas de la base extraída.

Por último, el sistema de procesamiento definido en XFL3 puede traducirse de forma automática a Java, C y C++ utilizando las herramientas de síntesis *xfj*, *xfc* y *xfc*, respectivamente.

3 EJEMPLOS DE DISEÑO

3.1. EXTRACCIÓN DE BORDES

Existen gran variedad de algoritmos que resuelven el problema de la extracción de bordes. Entre los que emplean técnicas de Lógica Fuzzy encontramos como referente el trabajo de Russo [16], en el que se propone un sistema basado en reglas. Se trata de un sistema que admite como entradas las diferencias de luminancia de un píxel con sus ocho vecinos más cercanos. De esta manera, los antecedentes de la base de reglas tienen en cuenta los valores positivos o negativos de estas diferencias y las posibles combinaciones de los bordes que se pueden dar (Figura 2). Tanto los conceptos “positivo”, “negativo”, como los posibles valores de la salida “negro”, “blanco”, se representan por funciones de pertenencia triangulares.

Estudios posteriores han modificado la base de reglas propuesta en [16]. Así por ejemplo, en [11], se añaden tres reglas más que cubren los casos en los que no existe borde y, por tanto, establecen el píxel de salida como

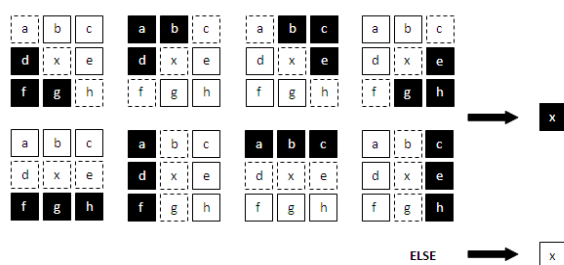


Figura 2: Reglas para la extracción de bordes en [16].

(b == Positive & c == Positive & d == Negative & e == Positive & f == Negative & g == Negative)	-> x = Black
(a == Negative & b == Negative & d == Negative & e == Positive & g == Positive & h == Positive)	-> x = Black
(b == Negative & c == Negative & d == Positive & e == Negative & f == Positive & g == Positive)	-> x = Black
(a == Positive & b == Positive & d == Positive & e == Negative & g == Negative & h == Negative)	-> x = Black
(a == Positive & b == Positive & c == Positive & f == Negative & g == Negative & h == Negative)	-> x = Black
(a == Negative & c == Positive & d == Negative & e == Positive & f == Negative & h == Positive)	-> x = Black
(a == Negative & b == Negative & c == Negative & f == Positive & g == Positive & h == Positive)	-> x = Black
(a == Positive & c == Negative & d == Positive & e == Negative & f == Positive & h == Negative)	-> x = Black
(a == Positive & b == Positive & c == Negative & d == Positive & e == Positive & f == Negative & g == Positive & h == Negative)	-> x = White
(a == Positive & b == Positive & c == Positive & d == Positive & e == Positive & f == Positive & g == Positive & h == Positive)	-> x = White
(b == Positive & d == Positive & e == Positive & g == Positive)	-> x = White

Figura 3: Base de reglas en *xfedit* para la extracción de bordes.

blanco. Esta base de reglas más extensa ha sido la descrita en Xfuzzy mediante *xfedit* (Figura 3).

Como operadores se han utilizado el producto para definir los conectivos de los antecedentes, min-max para la implicación y agregación, y el centro de área como método de concreción. Tanto los operadores como las funciones de pertenencia empleadas están ya incluidas en el paquete por defecto que carga el entorno por lo que, para este ejemplo, no ha sido necesario definir un nuevo paquete.

La simulación del sistema descrito se puede realizar haciendo uso de la herramienta *xfsim*. El modelo en Java construido para dicho fin nos permite seleccionar una imagen para calcular la extracción de bordes y ofrecer una imagen de salida como resultado de la aplicación del sistema difuso (Figura 4).

Puesto que las cuatro funciones de pertenencia involucradas en las reglas se han definido de forma heurística, lo más probable es que sus parámetros no sean óptimos. En vez de probar de forma manual diferentes valores, podemos emplear los algoritmos de aprendizaje supervisado de la herramienta *xfsl* y realizar un ajuste automático de parámetros (Figura 5a). En general, los resultados se mejoran, como se muestra en la Figura 5b.

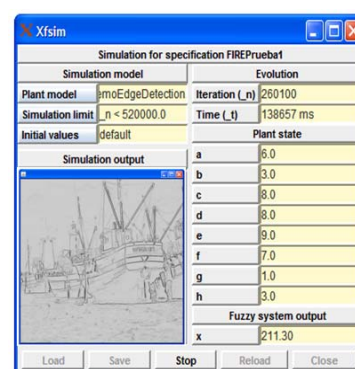


Figura 4: Simulación del sistema de extracción de bordes con *xfsim*.

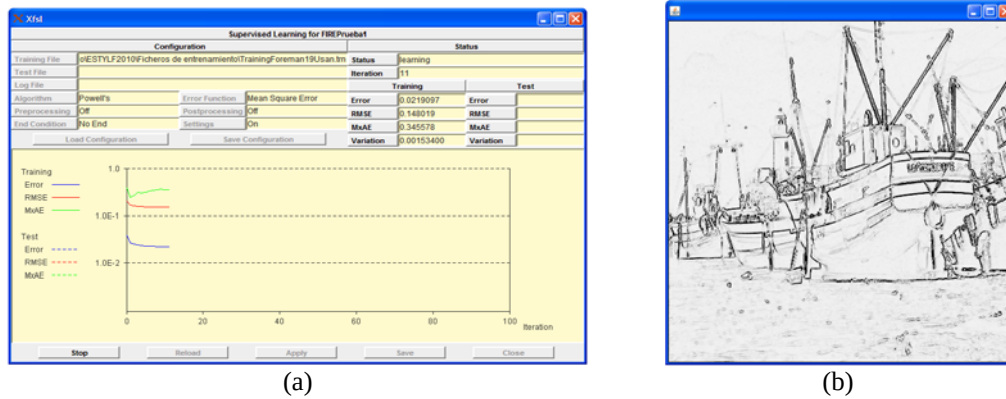


Figura 5: (a) Ventana de la herramienta de aprendizaje *xfsf*. (b) Resultado de la simulación tras el aprendizaje.

3.2. AUMENTO DE RESOLUCIÓN

Para agrandar o aumentar la resolución de una imagen hay que introducir nuevos píxeles que constituyan nuevas líneas y columnas. Esto se ilustra en la Figura 6 para un factor de ampliación igual a 2.

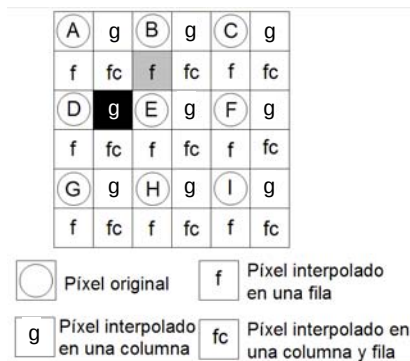


Figura 6: Aumento de resolución por un factor de 2.

El algoritmo propuesto en [2] consta de una primera fase en la que se interpolan los píxeles de las nuevas filas y columnas, que se muestran con los símbolos 'f' y 'g' respectivamente, en la Figura 6. Por ejemplo, para interpolar el píxel 'f' sombreado en gris se utilizan los píxeles originales {A, B, C, D, E, F}, mientras que para el píxel 'g' sombreado en negro, se emplean los píxeles originales {A, B, D, E, G, H}. La base de reglas fuzzy toma como entradas las 3 diferencias de luminancias en valor absoluto mostradas en la Figura 7a e infiere la luminancia del píxel a interpolar con las reglas mostradas en la Figura 7b. Se trata de reglas heurísticas que evalúan

la posible existencia de bordes para interpolar adaptándose a ellos. Los píxeles mostrados con el símbolo 'fc' en la Figura 6 se interpolan en una segunda fase como el valor medio de los resultados obtenidos al aplicar la base de reglas anterior en los 3+3 píxeles de las líneas superior e inferior, y los 3+3 píxeles de las columnas izquierda y derecha (utilizando los cuatro píxeles originales más cercanos y los cuatro píxeles más cercanos interpolados en la fase previa).

El sistema fuzzy fundamental en este algoritmo es el que contiene las reglas en la Figura 7b. Éste es el que se ha descrito en Xfuzzy 3 con la herramienta *xfedit* (Figura 8a). Para ello se ha tenido que definir un nuevo operador de inferencia con la herramienta *xfpkg* puesto que la regla 'otherwise' tiene la particularidad de activarse como el complemento de la suma de los grados de activación de las otras 3 reglas.

Las funciones de pertenencia para los conceptos 'small', 'strongly small' y 'large' que aparecen en la base de reglas pueden describirse heurísticamente, sin embargo, puesto que podemos disponer de un conjunto de datos numéricos correspondientes a pares de imágenes con distinta resolución, estas funciones se han ajustado en Xfuzzy 3 con la herramienta *xfsf* aplicando el algoritmo de Levenberg-Marquardt.

Para simular el procesado se ha creado una clase Java que pide al usuario seleccionar una imagen. La clase va recorriendo los píxeles de la imagen proporcionando las entradas necesarias para ir interpolando los diferentes píxeles. Esta clase genera la imagen agrandada permitiendo su visualización (Figura 8b) y escritura en un

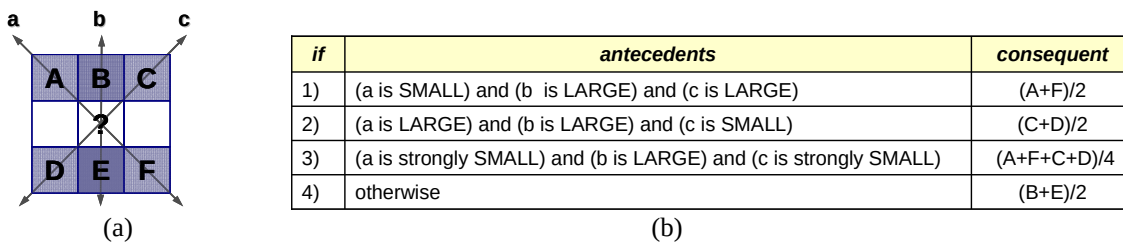


Figura 7: (a) Direcciones evaluadas para aplicar interpolación adaptativa a los bordes según las reglas en (b).

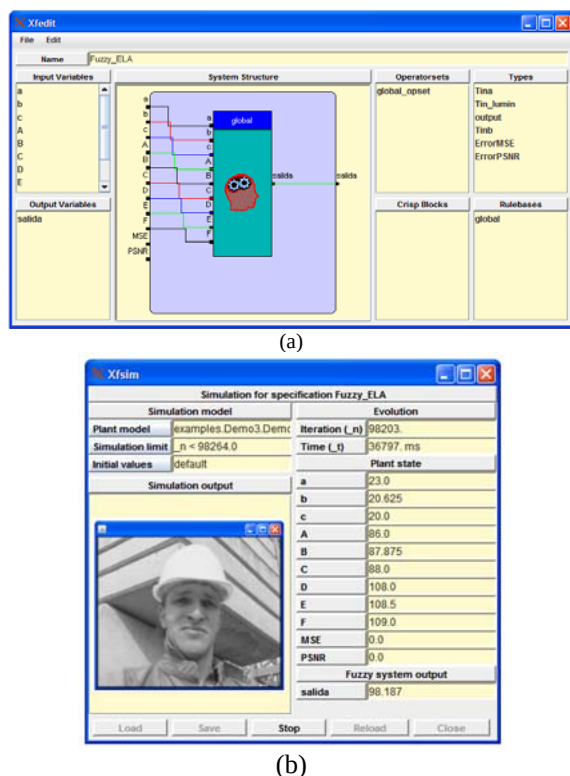


Figura 8: (a) Sistema fuzzy para agrandado de imágenes descrito con *xfedit* (b) Simulación con *xfsim*.

fichero. Además, la clase permite seleccionar al usuario una imagen ya agrandada con respecto a la que calcular errores. Si se selecciona dicha imagen, la simulación compara con ella la imagen resultante del procesado y calcula el MSE (Mean Square Error) y el PSNR (Peak Signal-to-Noise Ratio).

3.3. DESENTRELAZADO DE VÍDEO

En los estándares de TV analógica y en varios digitales no se envían las imágenes completas sino que se van intercalando imágenes con sólo las filas pares con otras con sólo las filas impares. Si el aparato donde se va a reproducir la secuencia no admite formato entrelazado es necesario desentrelazar las imágenes interpolando las filas que falten.

El algoritmo fuzzy descrito en el apartado anterior puede emplearse para calcular las filas que faltan a partir de las existentes. En la nomenclatura del desentrelazado se trataría de un interpolador espacial porque sólo considera las filas existentes en la imagen actual pero no en las precedentes (que sería una interpolación temporal). Sin embargo, el desentrelazado es mejor si se consideran ambos tipos de información espacial y temporal. Los algoritmos que se adaptan al movimiento se apoyan en la heurística para seleccionar interpolación espacial cuando el movimiento es grande e interpolación temporal cuando el movimiento es pequeño. El algoritmo propuesto en [3]

if	antecedents	consequent
1)	Motion is SMALL	I_T
2)	Motion is MEDIUM	$\gamma I_T + \lambda I_S$
3)	Motion is LARGE	I_S

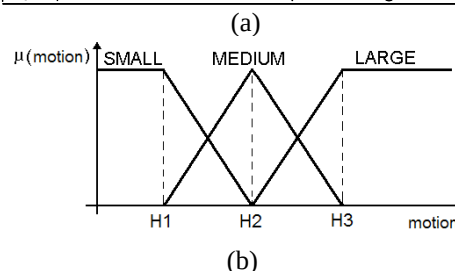


Figura 9: (a) Reglas para adaptar la interpolación al movimiento. (b) Funciones de pertenencia.

usa la base de reglas mostrada en la Figura 9a para evaluar de forma fuzzy el posible movimiento que haya en torno al píxel actual y seleccionar un tipo de interpolador u otro. Para implementar la interpolación espacial se emplea el algoritmo fuzzy comentado anteriormente. Como interpolación temporal se recurre a copiar las líneas de la imagen entrelazada anterior.

En esta aplicación se ha aprovechado la versatilidad de Xfuzzy para trabajar con sistemas jerárquicos. La Figura 10 muestra la ventana de *xfedit* donde se aprecia la estructura del sistema formado por dos módulos fuzzy que intercambian información no fuzzy. El módulo que se adapta al movimiento se implementa como un sistema Takagi-Sugeno de orden 1, de modo que no requiere definición de nuevos operadores.

Las funciones de pertenencia para los conceptos 'small', 'medium' y 'large' referidos al movimiento (Figura 9b) pueden describirse heurísticamente. No obstante, se obtienen mejores resultados si se aprenden empleando la herramienta *xfsl*. Conviene aprender también los parámetros γ y λ de los consecuentes e incluso ajustar los antecedentes del interpolador espacial, aprovechando la capacidad de Xfuzzy para ajustar sistemas jerárquicos. En este caso, hemos aplicado el algoritmo de Levenberg-Marquardt pero estimando, en vez de calculando, las derivadas.

Para simular este procesado se ha creado una clase Java que pregunta al usuario por las 4 imágenes a partir de las cuales se va a obtener la imagen desentrelazada actual. La clase va generando las entradas correspondientes al sistema y toma el valor del píxel inferido para construir la imagen resultante que se visualiza gráficamente y puede salvarse a un fichero (Figura 11). Las imágenes de partida son progresivas por lo que la clase las entrelaza como paso previo. Esto permite que la clase calcule los errores MSE y PSNR entre la imagen progresiva perfecta y la desentrelazada.

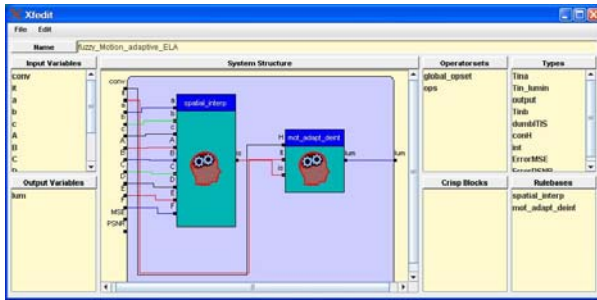


Figura 10: Sistema fuzzy para desentrelazado descrito en *xfedit*.



Figura 11: Simulación del sistema de desentrelazado con *xfsim*.

4 CONCLUSIONES

El entorno Xfuzzy 3, tradicionalmente usado en aplicaciones de control, también facilita el diseño de sistemas fuzzy para procesamiento de imágenes. Esto se ha ilustrado con tres ejemplos diferentes: la extracción de bordes, el aumento de resolución y el desentrelazado de imágenes.

Agradecimientos

Los autores agradecen el trabajo de Alejandro Valdivia, el primero en probar Xfuzzy 3 para procesar imágenes. Este trabajo ha sido parcialmente financiado por el Ministerio de Ciencia y Tecnología mediante el proyecto TEC2008-04920 y por la Junta de Andalucía mediante el proyecto P08-TIC-03674.

Referencias

- [1] I. Baturone, F. J. Moreno-Velo, S. Sánchez-Solano, A. Ollero. "Automatic design of fuzzy controllers for car-like autonomous robots". *IEEE Transactions on Fuzzy Systems*. Vol. 12, Pág. 447- 465, Aug. 2004.
- [2] P. Brox, I. Baturone, S. Sánchez Solano, A. Barriga. "Image enlargement using the Fuzzy-ELA

algorithm". *Proc. IPMU 2006*, Pág. 866-873, Julio 2006.

- [3] P. Brox, I. Baturone, S. Sánchez-Solano, J. Gutiérrez-Ríos, F. Fernández-Hernández. "A fuzzy edge-dependent motion adaptive algorithm for deinterlacing". *Fuzzy Sets and Systems*. Vol. 158 (3), Pág. 337-347, Feb. 2007.
- [4] S. S. L. Chang y L. A. Zadeh. "On fuzzy mapping and control". *IEEE Trans. on Syst., Man and Cybern.* Vol. 2, Pág. 30-34, 1972.
- [5] Y. S. Choi and R. Krishnapuram. "A robust approach to image enhancement based on fuzzy logic". *IEEE Trans. Image Processing*. Vol. 6, No. 6, Pág. 808-825, 1997.
- [6] K.-J. Choi, Y.-H. Lee, J.-W. Moon, C.-K. Park, F. Harashima. "Development of an automatic stencil inspection system using modified Hough transform and fuzzy logic". *IEEE Trans. Ind. Electr.* Vol. 54, No. 1, Pág. 604-611, Feb. 2007.
- [7] T. K. De and B. N. Chatterji. "An approach to a generalized technique for image contrast enhancement using the concept of fuzzy set". *Fuzzy Sets and Systems*. Vol. 25, Pág. 145-158, 1988.
- [8] F. Farbiz, M. B. Menhaj, S. A. Motamedi, and M. T. Hagan. "A new fuzzy logic filter for image enhancement". *IEEE Trans. Syst., Man and Cybern.* Vol. 30, No. 1, Pág. 110-119, Feb. 2000.
- [9] T. Law, H. Itoh, and H. Seki. "Image filtering, edge detection, and edge tracing using fuzzy reasoning". *IEEE Trans. Pattern Analysis and Machine Intelligence*. Vol. 18, No. 5, Pág. 481-491, 1996.
- [10] E. H. Mamdani. "Advances in the linguistic synthesis of fuzzy controllers". *Int. Jour. Man-Machine-Stud.* Vol. 8, No. 6, Pág. 669-678, 1976.
- [11] S. Mathur, A. Ahlawat. "Application of fuzzy logic on image edge detection". *Int. Conf. Intelligent Information and Engineering Systems, INFOS 2008*, Varna, Bulgaria. June-July 2008.
- [12] F.J. Moreno-Velo, S. Sánchez-Solano, A. Barriga, I. Baturone, D.R. López. "XFL3: An specification language for fuzzy systems". *Mathware & Soft Computing*, Vol. VIII, No. 3, Pág. 239-253, 2001.
- [13] M. Nachtgael and E. E. Kerre. "Classical and fuzzy approaches towards mathematical morphology". Chapter 1 in *Fuzzy techniques in image processing*, Physica-Verlag, 2000.
- [14] S. K. Pal and R. A. King. "Image enhancement using smoothing with fuzzy sets", *IEEE Trans. Syst., Man and Cybern.* Vol. 11, No. 7, Pág. 494-501, 1981.
- [15] F. Russo. "A user-friendly research tool for image processing with fuzzy rules". *Proc. First IEEE Int. Conf. Fuzzy System*, Pág. 561-568. 1992.
- [16] F. Russo and G. Ramponi. "Edge extraction by FIRE operators". *Proc. of the Third IEEE Int. Conf. on Fuzzy Systems*, Pág. 249-253, 1994.

GUAJE - A JAVA ENVIRONMENT FOR GENERATING UNDERSTANDABLE AND ACCURATE MODELS

José M. Alonso Luis Magdalena

European Centre for Soft Computing (ECSC)
33600 Mieres, Asturias, Spain
{jose.alonso,luis.magdalena}@softcomputing.es

Abstract

The term Soft Computing is usually used to refer to a family of several preexisting techniques (Fuzzy Logic, Neuro-computing, Probabilistic Reasoning, Evolutionary Computation, etc.) able to work in a cooperative way, taking profit from the main advantages of each individual technique, in order to solve lots of complex real-world problems for which other classical techniques are not quite well suited. In the specialized literature there are many Soft Computing tools, most of them freely available as open source software. This work gives an overview on existing tools for system modeling. Moreover, it introduces a new environment for building interpretable and accurate systems by means of combining several preexisting tools.

Keywords: Soft Computing, fuzzy modeling, open source software, interpretability-accuracy trade-off.

1 INTRODUCTION

Soft Computing (SC) is usually defined by its essential properties, as a family of techniques, as a complement of hard computing, and/or as a tool for coping with imprecision and uncertainty [10].

One of the main issues regarding SC techniques is their cooperative nature. Each individual technique, even each individual algorithm, has its own advantages and drawbacks. Therefore, designing hybrid systems made up of different techniques working together let us achieving more powerful systems, overcoming the problems which turn up when dealing with the component techniques alone. That is why hybrid systems

like for instance neuro fuzzy systems (NFS) [15] and genetic fuzzy systems (GFS) [5] are becoming more and more popular.

Since this paper deals with modeling understandable and accurate systems, it is mainly focused on those SC techniques which give priority to interpretability what involves two main aspects. First, the system description readability, i.e., the system description has to be transparent enough to present the system as a whole, describing its global behavior and trend. Second, the system understandability, i.e., its explanation ability for considering all possible individual situations and explaining specific behaviors for specific events.

The semantic expressivity of Fuzzy Logic (FL) [18] is well-known to be close to expert natural language yielding powerful tools for linguistic concept modeling. The use of linguistic variables [19] and linguistic rules [11] favors the interpretability of fuzzy models, at least from the readability or structural transparency point of view. As a result, Fuzzy Modeling (FM) [8], i.e., system modeling with fuzzy rule-based systems (FRBS), represents a fruitful research line. Unfortunately, using FL is not enough for building interpretable models. The whole modeling process must be carried out carefully, paying special attention to interpretability from the beginning to the end and imposing several constraints [13]. In addition, when dealing with complex real-world problems fuzzy models can be upgraded with learning capabilities of other SC techniques. Nevertheless, hybrid systems (mainly NFS and GFS) should also be designed keeping in mind the interpretability requirement.

Interpretability must be the central point on system modeling. In fact, some of the most hot and modern research topics like Precisiated Natural Language (PNL), Computing With Words (CWW), and/or Human Centric Computing (HCC) strongly rely on the characteristic interpretability of fuzzy models. However, it is important to remark that generating inter-

pretable systems is not a straightforward task. Nowadays, most fuzzy models are obtained giving priority to accuracy, disregarding their interpretability, what may be a great error because it yields almost black-box models. Notice that, looking for a good interpretability-accuracy trade-off is one of the most complex tasks on system modeling. It demands the aid of powerful software tools.

The rest of the paper is structured as follows. The next section makes a global review on available software that implements SC techniques for system modeling. Then, section 3 presents a new modeling framework based on the combination of several available tools. Finally, section 4 draws some conclusions.

2 AVAILABLE SOFTWARE

Most software for system modeling is available on the Web in the form of libraries and/or small tools which often come from academics and small research groups. Some of the most famous packages and tools are the following. In the field of evolutionary computation, JCLEC¹ (Java Class Library for Evolutionary Computation) and JMetal² (Metaheuristic Algorithms in Java) provide two nice frameworks for both evolutionary and multi-objective optimization. JavaNNS³ (Java version of Stuttgart Neural Network Simulator) is probably the best free suite for neural networks. Regarding fuzzy modeling, Xfuzzy⁴ (a development environment for fuzzy-inference-based systems), FisPro⁵ (Fuzzy Inference System Professional) and KBCT⁶ (Knowledge Base Configuration Tool) represent three useful tools. Finally, regarding neuro-fuzzy algorithms we can point out, among others, to NEFCLASS⁷ (Neuro-Fuzzy Classification).

In order to get wider visibility and cooperation with other researchers all tools enumerated above are freely downloadable as open source software, at least for research and education purposes. Thanks to the huge amount of available software it is really easy creating new small prototypes for lots of applications without the effort of starting from scratch. However, such prototypes not always work properly when dealing with real-world complex problems like large scale business applications. The main problem of such developments is their maintenance cost. Keeping a flexible and well-documented source code is a mandatory requirement

in order to promote the cooperation of several researchers in a common development. In addition, the coordination and control of subversions is a really difficult task when several researchers, sometimes located at different parts of the world, are only working on the software development during their own free time.

In consequence, the vast majority of freely available software is developed by small isolated groups and it becomes out of date resulting obsolete after a few months or years. A solution to this problem would be creating an international project to promote the cooperation of several research groups with the aim of creating a new open source software GNU⁸ Fuzzy that could be taken as a reference implementation [14]. It should be developed following open standards in order to make easier the cooperation of researchers all along the world. In addition, it should let adding easily new functionalities and modules keeping a common interface over a base core. Of course, a much more conservative alternative consists in using commercial tools like the Matlab⁹ toolboxes which include the well-known Fuzzy Toolbox and ANFIS (Adaptive Neuro-Fuzzy System) tool. Nevertheless, we should not reject the use of open source software because it would mean losing the richness of quickly incorporating new developments made by the active research community which is always working in emerging fields.

We are still far from a standard GNU Fuzzy Toolbox, but there are some interesting and successful attempts for going beyond the small and specialized tools, all of them with both research and educational purposes. For instance, FrIDA [4] is free and open source software in the form of a java-based graphical user interface (GUI) that joins several individual tools for data analysis and visualization. In this case all small programs were developed by the same researchers over the years. KEEL (Knowledge Extraction based on Evolutionary Learning) [6] is another more ambitious software tool created as part of a research project with several goals. To start with it includes a huge repository made up of hundreds of evolutionary learning algorithms developed by several authors (belonging to different research groups) as part of their own research works. Furthermore, new algorithms can be easily added. In addition, KEEL offers a user-friendly java GUI for designing experiments where different algorithms can be fairly compared with exactly the same data under a complete statistical analysis. Lastly, another quite famous tool putting together several machine-learning algorithms under the same interface is Weka¹⁰ (Data Mining Software in Java) [17]. It is also developed fol-

¹<http://jclec.sourceforge.net/>

²<http://jmetal.sourceforge.net/>

³<http://www.ra.cs.uni-tuebingen.de/SNNS/>

⁴<https://forja.rediris.es/projects/xfuzzy/>

⁵<http://www.inra.fr/internet/Departements/MIA/M/fispro/>

⁶<http://www.mat.upm.es/projects/advocate/kbct.htm>

⁷<http://fuzzy.cs.uni-magdeburg.de/nefclass/>

⁸<http://www.gnu.org/copyleft/gpl.html>

⁹<http://www.mathworks.com/>

¹⁰<http://www.cs.waikato.ac.nz/ml/weka/>

lowing the open source philosophy and it counts with a lot of related projects and contributors. It applies the Linux model of releases. It focuses on automatic extraction of knowledge from data but, unfortunately, it does not take care of the interpretability of the generated models and it does not include any algorithms for fuzzy modeling.

3 GUAJE ENVIRONMENT

The main novelty of the GUAJE approach is that it is the first one combining several software tools (not only libraries) with the aim of building interpretable fuzzy models. Notice that, interpretability is the main requirement and it is taken into consideration along the whole modeling process. Of course, accuracy is not forgotten because all kind of system must achieve at least a minimum accuracy, being completely useless otherwise. What we want to highlight is the fact that our proposal is especially designed for humanistic systems (defined by Zadeh as those systems whose behavior is strongly influenced by human judgment, perception or emotions [19]) used in real-world applications (for instance decision-support systems in fields like education, robotics, medicine, etc.) where there is a huge human-system interaction and, as a result, the interpretability of the model is strongly appreciated. Moreover, some loss of accuracy may be tolerated in exchange for a more interpretable model.

The core of GUAJE is the last downloadable version of KBCT (version 3.0) which has been upgraded with new functionalities and implementing the HILK (Highly Interpretable Linguistic Knowledge) fuzzy modeling methodology [2]. GUAJE combines the following six preexisting tools (the first five ones are freely available as open source) making use of the tools and methods that they provide:

- **KBCT**. Open source software for knowledge extraction and representation which combines expert knowledge and induced knowledge (knowledge automatically extracted from data) [1]. The combination of both kind of knowledge is made carefully and it includes consistency analysis, simplification, and optimization tasks.
- **FisPro**. An open source tool for creating fuzzy inference systems (FIS) to be used for reasoning purposes, especially for simulating a physical or biological system. It includes many algorithms (most of them implemented as C programs) for generating fuzzy partitions and rules directly from experimental data. In addition, it offers data and FIS visualization methods in a java-based user-friendly GUI.
- **Xfuzzy**. A free software development environment for generating FIS. It integrates a set of tools that ease the user to cover the several stages involved in the whole designing process, from their initial description to their final implementation, including simulation, edition and program synthesis. It is written in Java and all its tools are based on a common specification language named XFL3.
- **ORE**¹¹ (Ontology Rule Editor) [12]. A java-based open source platform-independent application for defining, managing and testing inference rules on a model represented by a specific ontology.
- **Weka**. An open source tool providing lots of algorithms for data mining. It includes the implementation of many classical algorithms like for example J48 which corresponds to the well-known C4.5 algorithm.
- **Matlab Fuzzy Toolbox**. It is the most widely used commercial tool for fuzzy systems. Its main advantage is that it takes profit from the fact that it is fully integrated with all functionalities provided by Matlab environment which is commonly used in engineering for both educational and business applications.

The Figure 1 illustrates how all these tools cooperate in the GUAJE environment. It shows the tasks made by each tool.

We consider two main sources of knowledge, experimental data and expert knowledge. An expert is a person who perfectly knows the problem under analysis. The expert is able to describe the system behavior and his/her background (knowledge, experience, preferences, etc.) is essential in order to get a good model. For that reason, as it can be seen in the diagram expert knowledge plays a key role being present all along the modeling process. Although GUAJE can work in a fully automatic way it also lets expert supervision and interaction at each step. Such integration only is possible if both expert and induced knowledge are formalized using the same language. In this case we use FL along with a set of constraints to guarantee the interpretability of the generated model.

Elicitation of expert knowledge is a really complex task and it usually becomes a bottleneck in the whole modeling process [7, 9]. In order to make easier expert knowledge extraction and representation we can use an intermediate level, the domain ontology. There are so many web ontologies that it is really easy to find one

¹¹<http://sourceforge.net/projects/ore/>

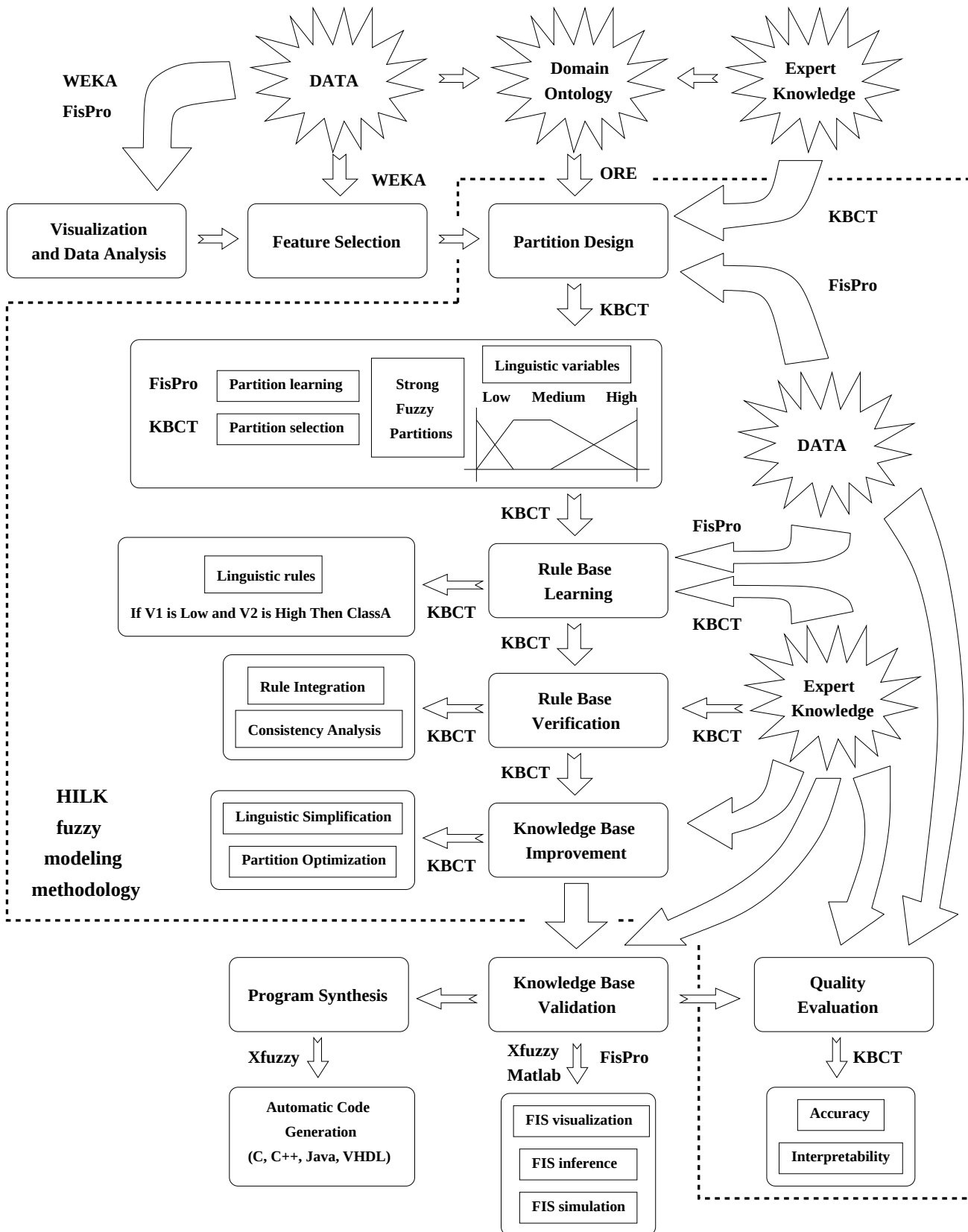


Figure 1: Scheme of the proposed GUAJE environment.

related to the problem under consideration. The hierarchy of concepts included in the selected ontology can be shown graphically to the expert who should identify the main influential input-output variables as well as a small set of basic expert rules [3]. All tasks to be done in relation with ontologies such as web searching, downloading, handling, representation, and so on are made calling to ORE functions directly from the KBCT graphical interface. As result of this preliminary stage, regardless of whether we use ontologies or not, we obtain a first simple expert knowledge base (KB) that has to be complemented and upgraded later with knowledge induced from experimental data.

The handling of data includes both visualization and analysis tasks that are carried out by KBCT making calls to Weka and FisPro algorithms. The first step consists in feature selection which is crucial to keep interpretability when dealing with large data files made up of dozens of input variables. Weka provides many algorithms for feature selection. Then, it is possible to use algorithms provided by FisPro for partition and automatic rule generation from data. Such algorithms are especially designed for generating interpretable partitions and rules. As a result, we can build a whole KB directly from experimental data.

It is important to remark that expert and induced KBs can be built in parallel or sequential steps. However, in order to get a unique KB the integration must be made carefully at both partition and rule levels. Therefore, for the sake of interpretability we recommend first closing the partition design stage (including both expert and data), i.e., defining fuzzy partitions with a global semantics before starting the rule base definition. Although it is not mandatory, we recommend the use of strong fuzzy partitions (SFP) [16] which satisfy most demanded semantic constraints (distinguishability, coverage, normality, convexity, etc.) to design interpretable partitions. Notice that, linguistic comparison for consistency analysis is only feasible when all rules (expert and induced ones) are defined using the same linguistic terms defined by the same fuzzy sets.

Once we have achieved a unique and consistent KB it is time to think about the interpretability-accuracy trade-off. KBCT offers powerful algorithms for linguistic simplification with the aim of increasing even more the KB interpretability while preserving the accuracy. It starts looking for redundant elements (labels, inputs, rules, etc.) that can be removed without altering the system accuracy. Then, it tries to merge elements always used together. Lastly, it forces removing elements apparently needed but not contributing too much to the final accuracy.

After getting a compact KB, accuracy can also be increased by applying optimization techniques for tuning the fuzzy partitions. Notice that, KBCT provides some optimization algorithms which are strongly constrained with the aim of not penalizing too much interpretability. The partition tuning process must keep the matching in between fuzzy sets and linguistic terms which should be fully meaningful according to the problem context and the expert background.

To sum up, the five main stages of the HILK methodology (partition design, rule base learning, rule base verification, knowledge base improvement, and quality evaluation) constitute the core of GUAJE environment. They are surrounded by a dash line in Figure 1 and they are directly implemented in the new enhanced version of KBCT. Although, for the sake of clarity, the five steps are represented sequentially, in practice the tool is absolutely flexible and the whole process is iterative. Everything can be supervised by an expert who can decide going back to the previous steps at whatever moment.

Finally, at the last stage, KBs built with KBCT can be exported to the format recognized by FisPro, Matlab, and Xfuzzy. In consequence, designed KBs can be used with the inference engines provided by such tools. In addition, we can take profit of the well-known Matlab Simulink environment in order to include the modeled system as part of a more complex simulated system. Furthermore, the program synthesis made by Xfuzzy is really useful for generating a final stand-alone module to be embedded in a real application. Note that the inverse translation is not allowed, i.e., KBs modified with FisPro, Matlab, or Xfuzzy can not be imported and opened again by KBCT. This is a restriction to preserve the interpretability of the final model because FisPro, Matlab, or Xfuzzy may violate the interpretability constraints imposed and satisfied by KBCT.

4 CONCLUSIONS

This paper has presented a new system modeling suite mainly focused on designing FRBSs with a good interpretability-accuracy trade-off by means of combining several preexisting tools. This approach lets us saving a lot of time because we reuse many algorithms already freely available on the Web as part of other tools which are distributed as open source software.

New algorithms can be added in the future with the aim of complementing the existing ones or adding new functionalities. For instance, we may incorporate some of the algorithms provided by Xfuzzy for partition tuning as well as for rule induction and simplification. We also would like to explore the possibility of

incorporating some algorithms for data analysis, like for example the fuzzy c-means clustering, included in FrIDA. Another open research line regards on adapting HILK methodology (and its implementation included in KBCT) to multi-objective problems. We are thinking on reusing some of the metaheuristic algorithms provided by JMetal.

GUAJE is freely available as open source software. If you are interested in trying its last release, please contact directly by e-mail with the first author of this contribution.

Notice that, due to space limitation we have not included any picture showing graphical interfaces nor application examples. Please, the interested reader is referred to the cited links and papers for further information.

Acknowledgements

GUAJE can be seen as an enhanced version of KBCT. The first version of KBCT was developed as part of the European research project ADVOCATE II supported by the European Commission (IST-2001-34508). The initial development started from FisPro what explains why both tools are so closely linked.

References

- [1] J. M. Alonso, L. Magdalena, and S. Guillaume. KBCT: A knowledge extraction and representation tool for fuzzy logic based systems. In *IEEE International Conference on Fuzzy Systems*, pages 989–994, 2004.
- [2] J. M. Alonso, L. Magdalena, and S. Guillaume. HILK: A new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism. *International Journal of Intelligent Systems*, 23(7):761–794, 2008.
- [3] J. M. Alonso, A. Muñoz, J. A. Botía, L. Magdalena, and A. F. Gómez-Skarmeta. Uso de ontologías para facilitar las tareas de extracción y representación de conocimiento en el diseño de sistemas basados en reglas borrosas. In *XIV Spanish ESTYLF conference on fuzzy logic and technologies*, pages 233–240, 2008.
- [4] C. Borgelt and G. González-Rodríguez. FrIDA - a free intelligent data analysis toolbox. In *IEEE International Conference on Fuzzy Systems*, pages 1892–1896, 2007.
- [5] O. Cordon, F. Herrera, F. Hoffmann, and L. Magdalena. *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*, volume 19. Advances in Fuzzy Systems - Applications and Theory, World Scientific Publishing Co. Pte. Ltd., 2001.
- [6] J. Alcalá-Fdez et al. KEEL: A software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13(3):307–318, 2009.
- [7] F. Hayes-Roth, D. A. Waterman, and D. B. Lenat. *Building expert systems*. Addison-Wesley, 1983.
- [8] H. Hellendoorn and D. Driankov. *Fuzzy model identification*. Springer-Verlag London, UK, 1997.
- [9] A. L. Kidd. *Knowledge elicitation for expert systems: A practical handbook*. Plenum Press, 1987.
- [10] L. Magdalena. What is soft computing? revisiting possible answers. In *8th International FLINS Conference on Computational Intelligence in Decision and Control*, pages 3–10, 2008.
- [11] E. H. Mamdani. Application of fuzzy logic to approximate reasoning using linguistic systems. *IEEE Transactions on Computers*, 26(12):1182–1191, 1977.
- [12] A. Muñoz, A. Vera, J. A. Botía, and A. F. Gómez-Skarmeta. Defining basic behaviours in ambient intelligence environments by means of rule-based programming with visual tools. In *1st Workshop of Artificial Intelligence Techniques for Ambient Intelligence. ECAI*, 2006.
- [13] C. Mencar and A. M. Fanelli. Interpretability constraints for fuzzy information granulation. *Information Sciences*, 178:4585–4618, 2008.
- [14] D. Nauck. GNU Fuzzy. In *IEEE International Conference on Fuzzy Systems*, pages 1019–1024, 2007.
- [15] D. Nauck, F. Klawonn, and R. Kruse. *Foundations of neuro-fuzzy systems*. J. Willey & Sons, Chichester, UK, 1997.
- [16] E. H. Ruspini. A new approach to clustering. *Information and Control*, 15(1):22–32, 1969.
- [17] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. 2nd Edition, Morgan Kaufmann, San Francisco, 2005.
- [18] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.
- [19] L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Parts I, II, and III. Information Sciences*, 8, 9:199–249, 301–357, 43–80, 1975.

SOFTWARE DIDÁCTICO SOBRE CONTROL DIFUSO

Enrique León González

José Galindo

E.U. Politécnica. Universidad de Málaga. c4776lg@copitima.com, jgg@lcc.uma.es

Resumen

La aplicación Software Didáctico para la Introducción al Control Difuso (SDICD) ha sido desarrollada en la Universidad de Málaga con la intención de acercar las materias de la lógica difusa y el control difuso a cualquier persona que esté interesada. El programa es más que un libro electrónico. Incluye diversos temas, ejemplos interactivos y test de autoevaluación, entre otras opciones.

1. INTRODUCCIÓN

El objetivo de Software Didáctico para la Introducción al Control Difuso (SDICD) es acercar, a cualquier interesado, los conceptos básicos de la lógica difusa y del control difuso [6][7], de un modo gradual y controlado. Esta aplicación se considera complementaria a SCD [3][4], un software de simulación de sistemas con control difuso altamente configurable (reglas, opciones de control, etc.). El carácter eminentemente práctico de SCD requiere conocimientos teóricos básicos antes de su utilización. Esos conocimientos se incluyen en el software didáctico que aquí presentamos.

La aplicación se ha diseñado a modo de libro con la intención de facilitar al usuario o alumno un aprendizaje teórico de los principios de la lógica difusa en general [6] y del control difuso en particular [7], de modo que pueda adquirir conocimientos de la materia de un modo progresivo, e intentando simplificar los conceptos matemáticos, definiciones y principios, de forma gráfica e interactiva para que permita al usuario asimilar los conceptos clave de la lógica difusa e introducirse en el interior de un controlador difuso.

La lógica difusa y el control difuso son temas incluidos dentro del programa de estudios de diversas titulaciones universitarias y programas de doctorado, tales como informática o ingeniería industrial. Como veremos, el temario incluido es suficientemente amplio para abarcar la mayoría de los temarios universitarios estudiados en este campo o, al menos, servir como buena introducción.

La movilidad a través de este libro electrónico se realiza de un modo intuitivo, a través de una serie de índices y enlaces, así como mediante el uso de cursores de movilidad incluidos en la aplicación. En todo momento, el usuario tiene disponibilidad de consultar las referencias y los significados de los términos importantes que se encuentren durante el estudio de la materia. El usuario podrá autoevaluarse de la materia para así medir sus conocimientos de un modo objetivo y claro.

Uno de los pilares es el apoyo que dan los dos ejemplos gráficos desarrollados para poder facilitar la comprensión de los conceptos teóricos. Estos ejemplos son interactivos, es decir, el usuario podrá ir variando los parámetros de entrada o las características de los controladores para comprobar cómo afectan dichos cambios al sistema o al propio controlador. Sendos ejemplos interactivos permiten comprender el desarrollo y funcionamiento interno de un controlador difuso, así como simular sistemas basados en lógica difusa.

Hay que destacar, como herramientas complementarias a SDICD, el software Xfuzzy [9], que permite diseñar y verificar sistemas difusos complejos (<http://www2.imse-cnm.csic.es/Xfuzzy>), y también el libro electrónico FLEB [1], que utiliza Xfuzzy.

En la Sección 2 explicamos el entorno de programación de la aplicación. En la Sección 3 se resumen muy brevemente las características más importantes de SDICD y finalmente damos unas breves conclusiones.

2. ENTORNO DEL PROGRAMA

El desarrollo de esta aplicación se ha realizado mediante el software de programación Microsoft Visual Basic 6.0® [2], debido a la pretensión de obtener una aplicación para Windows con un estilo de interfaz familiar a los usuarios.

Desde un punto de vista funcional se ha diseñado una aplicación que permita al usuario navegar por el programa de un modo dinámico, intuitivo y estructurado. Para conseguir esto se ha desarrollado un sistema de índices y enlaces que permiten al usuario en todo momento desplazarse por las secciones del programa, a modo de apartados y sub-apartados.

El programa está apoyado en un amplio sistema de archivos que se van cargando a medida que se va ejecutando. En dicho sistema de archivos podremos encontrar todos los textos que componen el software, las imágenes y formulas, las referencias bibliográficas, el glosario de términos, el sistema de evaluación (test), así como todo el sistema de control de los usuarios que puedan estar ejecutando el software. Dentro de SDICD se podrá encontrar una aplicación denominada *Editor de Herramientas*, que permitirá a un usuario *administrador* (con acceso restringido) modificar el glosario, las referencias bibliográficas y los test.

3. CARACTERÍSTICAS

Resumimos en ocho puntos las características principales de SDICD.

3.1 TEMARIO

SDICD es una aplicación tipo libro electrónico en la que se pueden encontrar 151 páginas a modo de presentaciones o pantallas. Para el temario nos hemos basado en [7] y otros trabajos, distribuyéndolo finalmente en 7 módulos o temas del siguiente modo:

- **Tema 1: Lógica Difusa.** Se introduce al usuario el concepto de *lógica difusa* en comparación con la denominada *lógica clásica*, se desarrolla brevemente la evolución histórica que ha seguido la materia, y se citan las principales aplicaciones.
- **Tema 2: Teoría de Conjuntos Difusos.** Se explican los conceptos de *conjunto difuso*, *función de pertenencia*, tipos de funciones de pertenencia que podemos encontrar, y algunos métodos que se pueden utilizar para obtener dichas funciones.
- **Tema 3: Conceptos y Operaciones con Conjuntos Difusos.** Este tema se divide en dos sub-bloques. En primer lugar se definen los principales conceptos con los que se pueden clasificar los distintos conjuntos difusos (soporte, núcleo, altura, etc.). Posteriormente se desarrollan los contenidos teóricos de las distintas operaciones que se pueden aplicar a los conjuntos difusos, como son las operaciones unarias, operaciones de *unión*, *intersección* y *complemento*, incluyendo las principales *t-normas* y *s-normas* que podemos encontrar (véase un buen compendio en [6]). Para acabar este tema se explican las operaciones más importantes que podemos encontrar para *comparar* conjuntos difusos.
- **Tema 4: Relaciones Difusas y Números Difusos.** En lo que se refiere a las *relaciones difusas* se centra en las operaciones de extensión y proyección cilíndrica. Con respecto a los *números difusos* se introduce al usuario en

los conocimientos teóricos de los mismos para posteriormente desarrollar el *principio de extensión*, concepto fundamental dentro de la lógica difusa.

- **Tema 5: Conceptos Básicos de Control.** Se introduce al usuario, de un modo abreviado, en los principales conceptos referentes a las teorías de *control*.
- **Tema 6: Controladores Difusos.** Este es el tema más importante y más extenso de todos. Aquí se desglosan las distintas partes en las que se puede dividir un *controlador difuso*, especialmente el módulo *difuminador* (*fuzzyfication*), la *base del conocimiento*, el *motor de inferencia* y el módulo de *concreción* (*defuzzyfication*).
- **Tema 7: Métodos de Ajuste y Tipos de Controladores Difusos.** En este último tema se desarrolla una clasificación de los controladores difusos en función de su método de ajuste o de su modo de funcionamiento.

3.2 CONTROL DE USUARIOS

Hay que distinguir dos tipos de usuario, el *administrador*, y el *usuario estándar*. El administrador podrá ejecutar ciertas herramientas del programa restringidas para los demás usuarios, como son, consultar las puntuaciones en los test de todos los usuarios registrados, ejecutar el *Editor de Herramientas*, y estará exento de realizar los test que se pueden encontrar al ejecutar la aplicación.

Dentro del control de usuarios podremos dar de alta nuevos usuarios, continuar la aplicación para un usuario ya registrado, consultar la puntuación obtenida hasta el momento en los distintos test que cada usuario haya realizado, y dar de baja a un usuario.

3.3 MENÚ DE SDICD

En SDICD hay 4 tipos de menús principales: *Archivo*, *Herramientas*, *Ejemplos* y *Ayuda*.

Dentro del menú *Archivo* podremos encontrar los sub-menús *Control de Usuarios* (apartado 3.2), *Imprimir*, *Ruta Acrobat Reader* (para configuración) y *Salir*.

El menú *Herramientas* se usa para consultar las distintas *Referencias*, los *Términos* (definiciones de palabras clave) de un modo ordenado (ver Figura 1), ó para ejecutar los test que podremos encontrar a lo largo de la aplicación.

El menú *Ejemplos* incluye los dos importantes ejemplos para el entendimiento del control difuso. Estos ejemplos se explican en el apartado 3.8.

Para poder orientar al usuario sobre este programa se usa el menú *Ayuda*, que consta de tres sub-menús: *Acerca de SDICD*, *Manual de Usuario* y *Glosario PDF*.

3.4 MOVILIDAD

Durante la ejecución del programa encontramos una serie de controles que utilizamos para movernos a lo largo de las páginas que componen la aplicación. Los botones de *Índice* y sendas flechas para *página Anterior* y *Siguiente*, están situados como muestra la Figura 2 en la parte inferior de la ventana principal de SDICD.

Por otra parte, mediante la utilización del teclado también podremos navegar por la aplicación. Además, casi todas las páginas incluyen enlaces, por los que podemos acceder a otros apartados relacionados.

3.5 TÉRMINOS Y REFERENCIAS

Se pueden consultar en cada página las definiciones de los términos clave y las referencias. Aparecerán en color azul los términos o expresiones clave contenidos en el glosario, y en naranja las referencias bibliográficas. La forma más simple para ello es usando los botones *Términos* y *Referencias*, situados en barra de estado inferior, junto al número de página y el botón de flecha izquierda (ir a *Página Anterior*), como se muestra en las Figuras 2 y 3. Estos botones se muestran cuando existe algún término y/o referencia resaltado dentro del texto contenido por esa página.

Al pulsar sobre cualquiera de estos dos botones se desplegará una lista que mostrará todos los términos o referencias mostradas en esa página, como se observa en la Figura 3. Eligiendo un término o referencia de la lista, aparecerá en la pantalla una ventana que contiene la información correspondiente al término o a la referencia seleccionada en la lista, y que desaparecerá de la pantalla en el momento que se pulse el botón *Aceptar*.

3.6 TEST

SDICD contiene en sus archivos más de 140 preguntas repartidas entre sus 7 temas. Puede accederse a esta herramienta de evaluación del alumno a través del Menú *Herramientas/Test*.

Además, al pasar por la última página de cada uno de los 7 temas nos encontraremos con la opción de realizar el test correspondiente a dicho tema.

El programa nos evaluará mediante un test compuesto por 10 preguntas elegidas aleatoriamente de las contenidas por SDICD para ese tema. Tanto el orden de las preguntas como el de las respuestas posibles también serán aleatorios. En la Figura 4 podremos observar la ventana *Test* con una pregunta concreta.

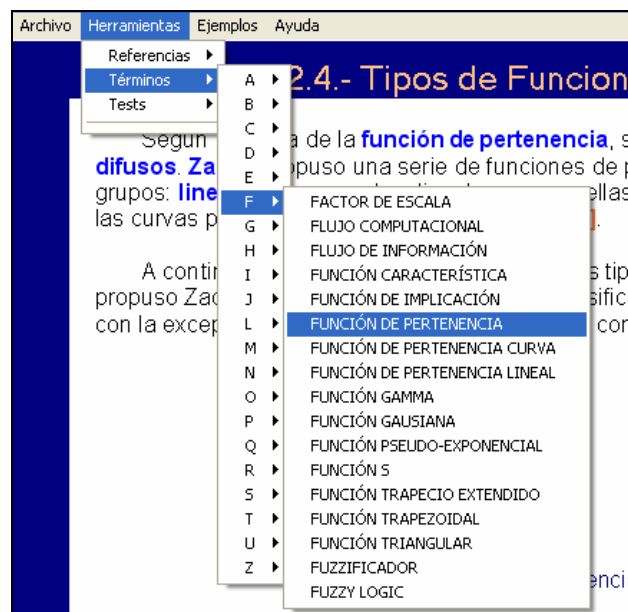


Figura 1: Menú *Herramientas*.

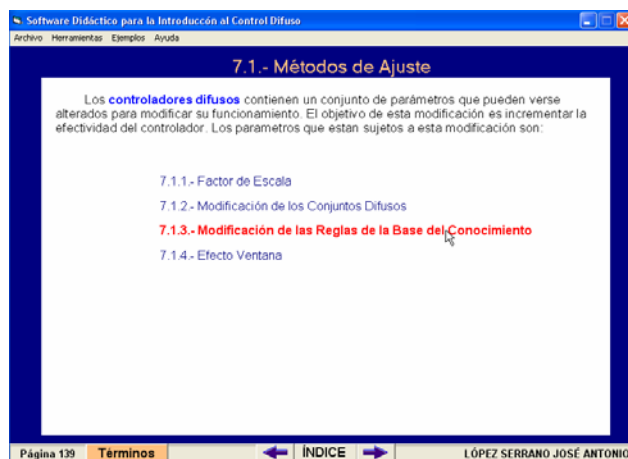


Figura 2: Movilidad en SDICD.

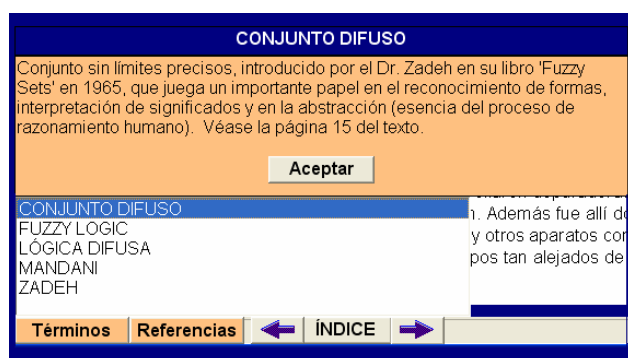


Figura 3: Términos y Referencias.

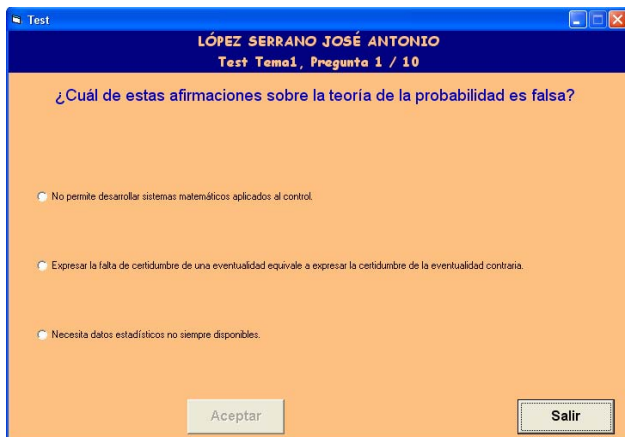


Figura 4: Test.

En el momento en que se han realizado todas las preguntas se muestra al usuario la puntuación obtenida, dándole la posibilidad de *Revisar Test*, opción por la que el alumno puede comprobar las respuestas marcadas, las respuestas correctas, y las incorrectas.

Una vez que se han realizado los test correspondientes a cada uno de los temas, SDICD dará al usuario la posibilidad de realizar un test general que contendrá 20 preguntas de entre todas las contenidas en el temario.

3.7 IMÁGENES Y REPRESENTACIÓN DINÁMICA

A lo largo de las *páginas* de SDICD encontramos cientos de imágenes y formulas. SDICD da la posibilidad a los alumnos de conocer de un modo claro y conciso el significado de las imágenes. Para ello solo han de situar el ratón sobre la imagen en cuestión y aparecerá en pantalla un recuadro amarillento con un texto breve con información relativa a la misma, tal y como se muestra en la Figura 5.

Otra herramienta relacionada con las imágenes y que se puede encontrar en SDICD es el botón *Representación*, situado junto a algunas imágenes y/o formulas, tal y como se muestra en la Figura 6. Este botón resulta útil para ayudar a la comprensión de algunas de las imágenes.

Siempre que nos encontremos con esta situación el programa nos está mostrando la imagen final definitiva. En caso de que se pulse el botón *Representación*, se pasa a visualizar una sucesión de imágenes superpuestas que crean un efecto animado con la intención de aclarar como se ha llegado a la imagen principal. En la Figura 6 se muestra una imagen que representa, a la izquierda, la medida de compatibilidad entre los conjuntos difusos B y A, de la derecha. El cálculo de esta medida de compatibilidad no es demasiado intuitivo por lo que pulsando el botón *Representación*, podremos obtener “paso a paso” cómo ha sido obtenido.

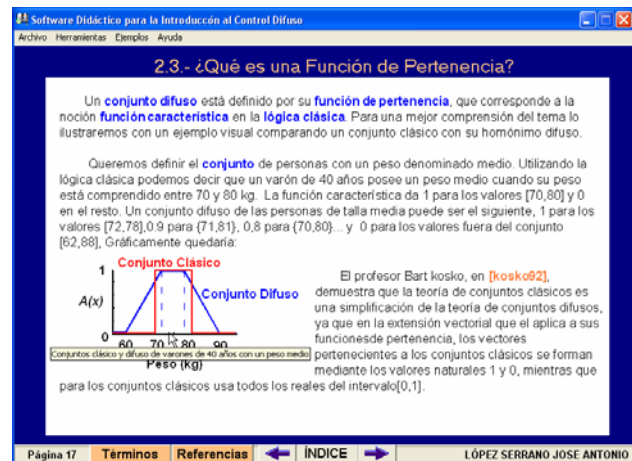


Figura 5: Imágenes en SCICD.

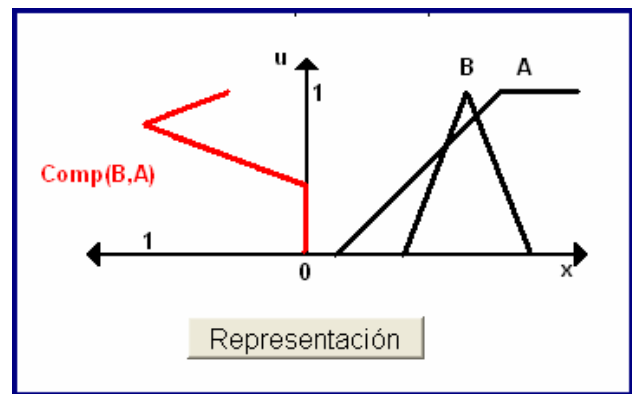


Figura 6: Imágenes en SCICD.

3.8 EJEMPLOS INTERACTIVOS

SDICD incluye dos ejemplos que se presentan como fundamentales para el entendimiento del Control Difuso. Estos ejemplos son la *Simulación del Control Difuso de un Cruce de Semáforos* y la *Simulación del Control difuso de un Invernadero* [4][5].

En el ejemplo *Simulación del Control Difuso de un cruce de Semáforos* (Figura 7) se comparará de un modo estadístico el rendimiento obtenido por un cruce de semáforos controlados mediante control difuso con respecto a la utilización de semáforos de tiempo fijo, y está basado los desarrollos de Mandani y Pappis [8].

Aquí pueden modificarse las variables de entrada y la configuración del controlador difuso. El valor de salida es el tiempo en el que el semáforo está en cada estado, mientras que las variables de entrada controlan, para cada una de las dos calles, el ritmo de llegada de vehículos y la cola de vehículos esperando.

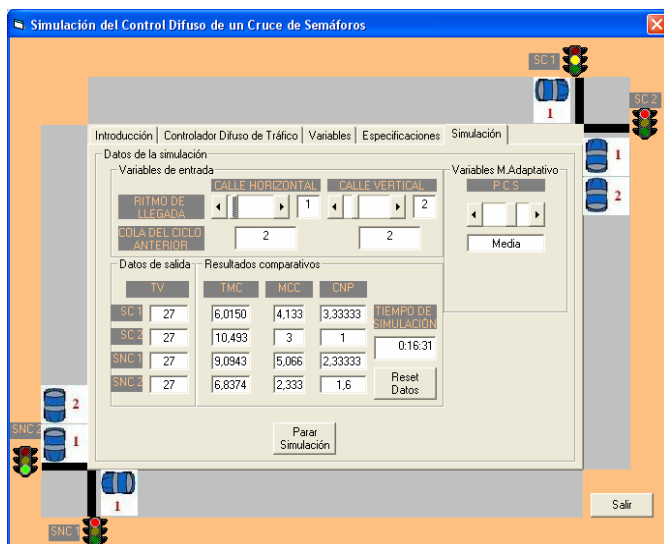


Figura 7: Simulación del control difuso de un cruce de semáforos.

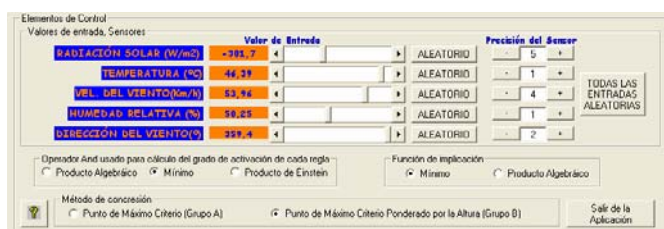


Figura 8: Variables de entrada al invernadero.

Este ejemplo facilita comprender cómo se desarrolla un controlador difuso a partir de unas especificaciones previas, además da la posibilidad de comparar los resultados obtenidos en función de los distintos valores que pueden tomar tanto las variables de entrada como los parámetros del módulo adaptativo incluidas en el control.

En la Figura 7 pueden observarse que tenemos dos cruces en sendas esquinas de la ventana. Ambos cruces tienen sus dos semáforos y a ellos llegan automóviles numerados. Uno tiene un control difuso y otro un control clásico. En la parte central de la ventana tenemos distintas lengüetas por las que podemos acceder a explicaciones sobre el controlador, sus variables, sus especificaciones o el apartado para simulación, que es el que se muestra en dicha figura. Ahí podemos iniciar y detener la simulación y controlar los valores de las variables.

El otro ejemplo interactivo, *Simulación del Control Difuso de un Invernadero* [3][4][5], muestra un sistema automático para un invernadero de verduras, considerando variables como la temperatura, la humedad, la radiación solar y la dirección y velocidad del viento.

Con esas variables de entrada el controlador establece el valor óptimo para las variables de salida, que son el grado de apertura de las distintas ventanas del invernadero, y si se activa o no se activa la pantalla térmica (para reducir la radiación solar) y la nebulización (para aumentar la humedad y reducir la temperatura).

De modo sencillo, la aplicación muestra el desarrollo de los cálculos que se producen dentro del controlador difuso en función de unos valores puntuales de entrada y de una serie de opciones incluidas en el controlador. Esta simulación es un desarrollo simplificado del ejemplo incluido en Software para el Control Difuso (SCD) [3][4].

En la Figura 8 podemos observar que en este ejemplo la aplicación permite modificar los valores de 5 variables de entrada: *Radiación Solar*, *Temperatura*, *Humedad Relativa*, *Velocidad del Viento* y *Dirección del Viento*. Además permite al usuario modificar una serie de aspectos de configuración del controlador difuso, como son: *Operador And* usado para el cálculo del grado de activación de cada regla, *Función de Implicación* y *Método de Concreción*. El programa permite observar los cálculos en cada una de las partes del controlador.

En la pestaña *Entradas* (Figura 9) se simula el módulo difuminador de un controlador difuso, obteniendo los valores para cada una de las variables en función de la precisión que tengamos asignada al sensor correspondiente para cada variable. La representación gráfica de cada valor obtenido resulta de gran utilidad para seguir el proceso del controlador difuso.

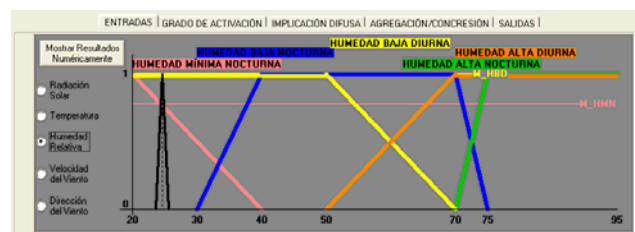


Figura 9: Difuminación de entradas del invernadero.

En la pestaña *Grado de Activación* se pueden ver los antecedentes y consecuentes de cada una de las reglas existentes, así como el grado de activación calculado para cada regla en función de los valores de entrada y del *Operador And* elegido por el usuario.

En el apartado *Implicación Difusa* (Figura 10) se muestra como se calculan los conjuntos difusos de salida para cada regla. Para esto se tienen en cuenta los consecuentes activados con cada regla, el valor del grado de activación de cada regla y la función de implicación elegida por el usuario.



Figura 10: Implicación difusa para el controlador del invernadero.

En la ventana *Agregación/Concreción* se muestra el valor de todos los conjuntos obtenidos para las diferentes variables de salida en cada una de las reglas activadas, y se muestra el valor *crisp* de salida obtenido en función de los conjuntos resultantes de la implicación difusa y del método de concreción elegido.

Al pulsar sobre la pestaña *Salidas* (Figura 11) podemos observar de un modo gráfico sencillo el estado del invernadero en cada momento



Figura 11: Gráfico del invernadero mostrando el estado de las variables de salida.

4. CONCLUSIONES

Desde un punto de vista funcional SDICD ha sido desarrollado con un entorno lo más ameno posible, de forma que permita un cómodo manejo de las distintas opciones implementadas.

Desde el punto de vista teórico se ha intentado realizar un desarrollo lo más claro posible de los principios de la lógica difusa y el control difuso. Se han plasmado conceptos matemáticos, definiciones, principios... que permitan alumno introducirse en la lógica difusa en general y en el interior de un controlador difuso en particular.

En todo momento el alumno tiene disponibilidad de consultar referencias y términos que se puede encontrar durante el estudio de la materia de un modo accesible y rápido. Así como examinarse de la materia y poder medir sus conocimientos de un modo objetivo y claro.

Un punto fuerte y muy útil es la inclusión de los dos ejemplos explicados, que permiten comprender el

desarrollo de un controlador difuso y simular Sistemas Basados en Reglas (SBR) con lógica difusa.

Esta aplicación se ha desarrollado para facilitar, a los alumnos de ingeniería y de cualquier estudio relacionado, el acercamiento al mundo de la lógica difusa y su aplicación al control.

Agradecimientos

Este trabajo ha sido parcialmente apoyado por el Gobierno de España (proyectos TIN2006-14285 y TIN2006-07262) y el Gobierno de Andalucía (proyecto P06-TIC-01570).

Referencias

- [1] Bermúdez, A., Barriga, A., Baturone, I., Sánchez-Solano, S. FLEB: A Fuzzy Logic E-Book. Proc. European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, pp. 549-554, Tenerife, 2001.
- [2] Ceballos Sierra, Francisco J. *Enciclopedia de Microsoft Visual 6*. Editorial Ra-Ma. 1999.
- [3] Escobar Rodríguez, Calixto. *Software para Control Difuso de todo tipo de Sistemas (SCD): Aplicación al Control de Invernaderos Industriales*. Proyecto Fin de Carrera, Ingeniería Técnica Industrial, Universidad de Málaga. 2003.
- [4] Escobar, C., Galindo, J. Software Genérico de Control Difuso: Aplicación en Agricultura Industrial. XII Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF'2004), pp. 551-556. Jaén (Spain), 2004.
- [5] Escobar, C., Galindo, J. Fuzzy Control in Agriculture: Simulation Software. Industrial Simulation Conference 2004 (ISC-2004), pp. 45-49. Ed.: J. Marín and V. Koncar. Málaga (Spain), June 2004 (www.lcc.uma.es).
- [6] Galindo, J. *Introduction and Trends to Fuzzy Logic and Fuzzy Databases*. In Handbook of Research on Fuzzy Information Processing in Databases, Vol. I, pp. 1-33. Hershey, PA, USA: Information Science Reference (<http://www.info-sci-ref.com>), 2008.
- [7] Galindo, J. Curso Introductorio de Conjuntos y Sistemas Difusos (Lógica Difusa y Aplicaciones). <http://www.lcc.uma.es/~ppgg/FSS> (accedido 2010).
- [8] Mandani, E., Pappis, C. *A Fuzzy Logic Controller for a Traffic Intersection*. IEEE Trans. on Systems, Man and Cybernetics, SMC-7, pp. 707-717, 1977.
- [9] Moreno-Velo, F.J., Baturone, I., Sánchez-Solano, S., Barriga, A. Rapid Design of Fuzzy Systems with XFUZZY. Proc. IEEE Int. Conference on Fuzzy Systems, pp. 342-347, St. Louis, 2003.

N_ip_{1.5}: UNA HERRAMIENTA SOFTWARE PARA LA GENERACIÓN DE CONJUNTOS DE DATOS CON IMPERFECCIÓN PARA MINERÍA DE DATOS

José M. Cadenas

Juan V. Carrillo

M. Carmen Garrido

Enrique Muñoz

Departamento de Ingeniería de la Información y las Comunicaciones

Universidad de Murcia. Murcia. Spain

jcadenas@um.es

jvcarrillo@um.es

carmengarrido@um.es

enriquemuba@um.es

Resumen

Este trabajo introduce una herramienta software llamada N_ip_{1.5} la cual ayuda a generar y tratar conjuntos de datos con imperfección para el proceso de minería de datos y aprendizaje. Esta herramienta software ha sido concebida para su uso en entornos de investigación e incluye la posibilidad de distintos formatos de entrada/salida standard y definidos por el usuario; gestión de los atributos; generar e introducir valores desconocidos, ruido y etiquetas lingüísticas, etc.

Palabras Clave: Herramienta software, Datos con imperfección, Soft Computing.

1 INTRODUCCIÓN

En el proceso de descubrimiento de conocimiento en conjuntos de datos se debe tener en cuenta la imperfección de la información para poder extraer modelos más cercanos a la realidad. Sin embargo, a pesar de la gran cantidad de técnicas existentes para dicho proceso, sólo unas pocas tienen en cuenta la imperfección, entre otros motivos, por la falta de conjuntos de datos imperfectos con las que probar y comparar los algoritmos. Aunque el objetivo de las técnicas es ser usadas con conjuntos de datos con imperfección real, la manipulación de forma controlada de conjuntos de datos “perfectos” permite experimentar y comparar resultados de los algoritmos que gestionen datos imperfectos, ofreciendo un marco de trabajo común; este es el propósito de N_ip_{1.5} [4].

Este trabajo presenta la herramienta N_ip_{1.5}, un software de tratamiento de datos que permite crear conjuntos de datos imperfectos, introduciendo ruido e imprecisión entre otras características. Además, N_ip_{1.5}

es capaz de generar etiquetas lingüísticas fuzzy mediante un algoritmo de particionamiento. Para ello el trabajo será estructurado de la siguiente forma: en la Sección 2 se hará una introducción a la imperfección que nos podemos encontrar en los datos, en la Sección 3 se presentarán algunas herramientas usadas para el tratamiento de datos. Posteriormente, en la Sección 4, se presentará la herramienta N_ip_{1.5}. Finalmente en la Sección 5 se presentan las conclusiones.

2 IMPERFECCIÓN EN CONJUNTOS DE DATOS

La información imperfecta aparece de manera general en dominios y situaciones reales. Los errores instrumentales o la corrupción debida al ruido durante la recogida de datos puede dar lugar a información incompleta para atributos específicos. En otros casos, la extracción de información exacta puede ser excesivamente costosa. En ese caso, puede ser útil aumentar el conjunto de datos con información adicional de un experto, que suele trabajar con términos lingüísticos fuzzy como *pequeño*, *más o menos*, *cercano a*, etc.

2.1 RUIDO Y VALORES DESCONOCIDOS

El ruido es un problema común en el análisis de datos, y como consecuencia debe ser gestionado y tratado debidamente. Este se define, [7], como cualquier cosa que oscurece la relación entre los atributos y la clase. De esta definición se obtienen las fuentes de ruido que aparecen en los datos: atributos erróneos, atributos desconocidos, atributos incompletos y atributos redundantes. El tratamiento de los atributos erróneos es complejo debido a la dificultad de distinguirlos de atributos correctos. Para gestionar los atributos desconocidos existen distintos enfoques: eliminar las instancias con atributos desconocidos, sustituir dichos atributos por su media o moda, usar fuzzy sets o intervalos, y otros métodos más complejos.

2.2 IMPRECISIÓN E INCERTIDUMBRE

Según la Teoría de la Posibilidad [6], un ítem de información puede ser impreciso e incierto. La imprecisión se manifiesta en el contenido de un ítem de información, mientras que la incertidumbre se refiere a su parecido con la realidad.

En [6] se define un ítem de información como una 4-tupla (atributo, objeto, valor, confianza) que representa el valor tomado por un objeto y la confianza sobre dicho valor. Un ítem de información es preciso cuando el conjunto correspondiente a su componente valor no se puede subdividir; en caso contrario hablamos de información imprecisa. Hay otros términos para referirse a la información imprecisa como vago, fuzzy, general o ambiguo. Un ítem de información es ambiguo cuando se refiere a contextos diferentes o conjuntos diferentes de referencia; es general si se nombra a una clase de objetos de los cuales se expresa una propiedad común; es vago si hay ausencia de límites claros en el conjuntos de valores asignados al objeto.

Algunos autores hablan de una “escala de conocimiento” que va desde la *certeza* absoluta a la *falta de conocimiento*, pasando por la incertidumbre [8], aunque no hay acuerdo sobre una terminología común sobre la incertidumbre.

2.3 REPRESENTAR LA INCERTIDUMBRE

Como se ha comentado, la incertidumbre y la imprecisión son conceptos complementarios, ya que la incertidumbre se debe normalmente a la imprecisión en los datos. Los datos pueden ser representados junto a su imprecisión y/o incertidumbre usando intervalos, modelos estocásticos o fuzzy sets, utilizando un conjunto de valores en lugar de uno único. La teoría de la probabilidad, el análisis de intervalos y la teoría de la posibilidad son los marcos matemáticos de trabajo más usados que nos permiten conceptualizar la incertidumbre y la imprecisión.

La primera aproximación para representar la incertidumbre es el análisis de intervalos, representando un valor con un intervalo. Se utilizan conjuntos clásicos en lugar de un número, pero el grado de pertenencia sigue siendo un valor binario. La teoría de la probabilidad es otra manera de representar la incertidumbre, haciendo un análisis probabilístico para representar la incertidumbre mediante la probabilidad asociada con los eventos. Las incertidumbre asociadas a las entradas del modelo se describen mediante distribuciones de probabilidad, y el objetivo es estimar la salida de dichas distribuciones. La teoría de los Fuzzy Sets y la teoría de la posibilidad [6], como extensión de la

noción de conjuntos clásicos, reemplazan la función bivaluada de pertenencia al conjuntos por una función con valor real; esto es, un fuzzy set representa el concepto de número aproximados, y la pertenencia al conjunto viene dada por $\alpha \in [0, 1]$.

3 HERRAMIENTAS Y CONJUNTOS DE DATOS

3.1 CONJUNTOS DE DATOS

Para poder comprobar la eficacia de una técnica de aprendizaje se utilizan conjuntos de datos obtenidas de diferentes entornos reales, algunas de las cuales se pueden obtener de repositorios públicos [1, 2]. Estos conjuntos de datos no suelen contener información imperfecta, a excepción de algún valor desconocido o, algunas de ellas, valores en intervalos. Como se comentó anteriormente, los problemas reales suelen tener algún tipo de información imperfecta que, de alguna manera, se ignora al publicar los conjuntos de datos en los repositorios. Por otro lado, cuando se diseña una técnica capaz de gestionar información imperfecta, los autores tienen que crear sus propios conjuntos de datos para probar y ajustar su método. Con N_{ip1.5} [4], se facilita dicho proceso de una manera más formal y rigurosa para simplificar asimismo la comparación de resultados con otros autores. En la siguiente sección veremos algunas de las herramientas software de minería de datos junto con sus principales características desde este punto de vista.

3.2 HERRAMIENTAS SOFTWARE

En esta subsección se verán algunas de las herramientas más usadas para el tratamiento de datos, especialmente aquellas destinadas a la minería de datos. Además, se comparan sus principales características con las de N_{ip1.5}.

- *Sodas2* [5]: Es una herramienta que soporta el análisis simbólico de datos. Trata de generalizar el proceso de minería de datos y estadística a un nivel superior, descrito por datos simbólicos. De esta manera, se transforman los datos en datos más manejables y más complejos. El funcionamiento de *Sodas2* consiste en construir un conjunto de datos simbólicos que resume la información del conjunto de datos inicial y, tras ello, realizar el *análisis simbólico*.

El uso de datos simbólicos permite introducir diferentes tipos de imperfección en los datos: - Atributos multivaluados, - intervalos y - Atributos multivaluados con pesos. Este tipo de atributos permiten representar otros tipos de imperfección,

Tabla 1: Características de las herramientas Software con respecto al tratamiento de datos.

(S: Sí, N: No, I: Medio)		Sodas2	Weka	Keel	RapidMiner	MiningMart	NiP1.5
Gestión de datos		S	S	S	S	S	S
Formato	Standard	I	S	S	S	I	I
	Usuario	N	N	N	N	N	S
Discretización	Crisp	S	S	S	S	S	S
	Fuzzy	N	N	N	N	N	S
Valores Desconocidos	Introducción	N	S	S	N	N	S
	Imputación	S	S	S	I	I	I
Ruido en atributo	Nominal	N	S	N	S	N	S
	N Numérico	N	N	N	S	N	S

como datos fuzzy, imprecisos e inciertos. No obstante, *Sodas2* no permite el uso directo de tecnología fuzzy ni la introducción de valores desconocidos, ruido o la modificación de los datos para introducir ningún tipo de imperfección.

- *Weka* [13]: Proporciona un conjunto de herramientas para el preproceso de datos, clasificación, regresión, clustering, reglas de asociación y visualización. Para el preproceso de datos, Weka ofrece una amplia variedad de técnicas para preproceso de atributos e instancias. En el caso de datos con imperfección, el número de técnicas disminuye, ofreciendo pocas posibilidades. Weka únicamente es capaz de gestionar valores desconocidos, y proporciona herramientas para sustituir dichos valores por la moda o media. Además, permite la adición de ruido en atributos nominales.
- *Keel* [1]: Es un software para evaluar algoritmos evolutivos para problemas de minería de datos. Contiene una gran colección de algoritmos clásicos de extracción de conocimiento, técnicas de preproceso, algoritmos de aprendizaje basados en inteligencia computacional, modelos híbridos, y un módulo de tests estadísticos para comparación. KEEL permite el uso de diferentes formatos de entrada y salida como CSV, XML o ARFF.

Su módulo de *Gestión de Datos* permite realizar la fase de preproceso proporcionando una amplia variedad de técnicas para selección de instancias, selección de características, discretización, etc. Una vez más, las técnicas proporcionadas para gestión de datos imperfectos no son muchas, ya que sólo se permite el uso de valores desconocidos mediante varios métodos de imputación (borrado de instancias con desconocidos, imputación de k-vecinos más cercanos, imputación de k-medias, etc).

- *Rapid miner* [10]: Antiguamente llamado YALE, es un entorno para aprendizaje computacional y minería de datos destinado a soportar el paradigma de prototipado rápido. Ofrece la posibilidad de usar diferentes formatos de entrada/salida.

El número de técnicas de preproceso ofrecidas por la herramienta es grande pero, de nuevo, disminuye si nos centramos en técnicas que gestionen datos imperfectos. Rapid miner permite gestionar valores desconocidos usando varios métodos de imputación: reemplazo con la media, moda, máximo, mínimo o el valor predicho mediante una técnica de minería de datos definida por el usuario. También permite la introducción de ruido en atributos numéricos y/o nominales.

- *MiningMart* [11]: Se desarrolla con el propósito de reutilizar técnicas con éxito en el preproceso de grandes volúmenes de datos. MiningMart no se centra en todo el proceso de descubrimiento de conocimiento, sino que sólo trata el preproceso. Ofrece diversas técnicas de agregación, discretización, limpieza de datos, tratamiento de valores nulos y selección de atributos relevantes. El único soporte que ofrece para datos imperfectos es sobre los valores desconocidos, eliminando las instancias que los contienen o bien usando algún método de imputación, como reemplazar el valor por la moda, media, un valor estocástico, etc., o usando un algoritmo para predecir el valor.
- *Otras herramientas*: Existen otras herramientas centradas en el proceso de extracción de conocimiento, como *ADaM* [12], *D2K* [9] or *KNIME* [3], entre otras, pero no prestan mucha atención al tratamiento de datos imperfectos.

En la Tabla 1 podemos ver un resumen de las características principales de las herramientas examinadas. La mayoría centran su funcionalidad en el tratamiento de los datos, teniendo sólo en cuenta la imperfección de tipo valor desconocido. En cuanto a la introducción de imperfección, son sólo algunas herramientas las que permiten introducir valores desconocidos, menos las que introducen ruido, y no hay ninguna que trabaje con conjuntos fuzzy.

NiP1.5 surge para paliar esta falta en el estado del arte de herramientas de ayuda a la investigación en el

campo de descubrimiento de conocimiento. Este software es capaz de generar conjuntos de datos con esos tipos de imperfección que el resto de herramientas no soporta o soporta parcialmente.

4 NIP 1.5

N_ip1.5 [4] es una herramienta que permite generar y gestionar conjuntos de datos con imperfección, concebida para su uso en entornos de investigación debido a la carencia de un software de características similares que ayude a establecer un marco común de trabajo con datos imperfectos. En esta sección veremos las principales funcionalidades del sistema, describiendo el uso de las diversas opciones. La figura 1 muestra, de manera muy general, las funcionalidades del software.

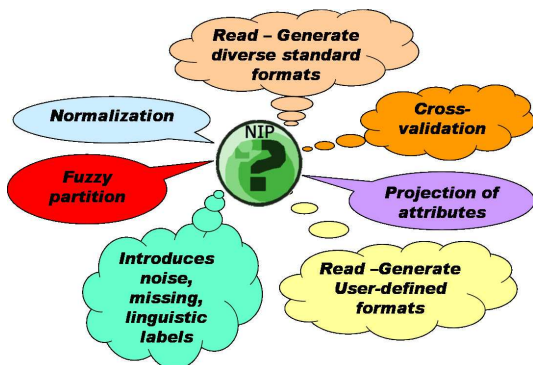


Figura 1: Funcionalidades de N_ip1.5

4.1 FORMATOS DE ENTRADA/SALIDA

N_ip1.5 tiene cuatro formatos de entrada y salida predefinidos o standard: WEKA, KEEL, UCI, CSV. Estos formatos son los más usados en la literatura y en los repositorios de conjuntos de datos [1, 2, 13]. Por otra parte, dado que se trata de una herramienta para la preparación de conjuntos de datos para su posterior uso con herramientas de tratamiento, el usuario puede definir su propio formato utilizando: separadores, representación de valores desconocidos, peso de los ejemplos, etc. En cuanto a los formatos de salida, también son personalizables. Además, N_ip1.5 permite generar conjuntos de datos, según el formato elegido, para realizar una validación cruzada (Figura 2).

4.2 GESTIÓN DE ATRIBUTOS

Cuando N_ip1.5 analiza el conjunto de datos, asigna a cada atributo un tipo, numérico o nominal. Así mismo, N_ip1.5 ofrece información útil sobre las características del conjunto de datos, como el número de ejemplos, porcentaje de valores desconocidos, y valores estadísticos (Figura 3). Además, el usuario puede realizar las siguientes acciones:

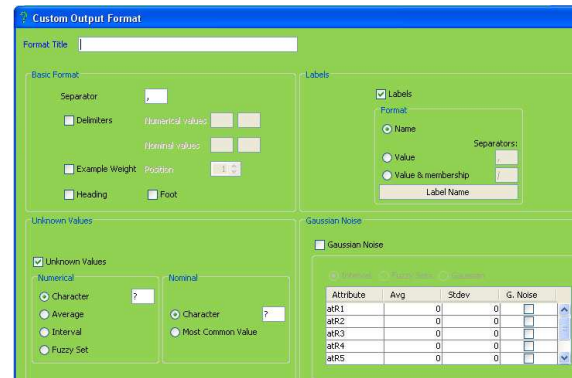


Figura 2: Formato entrada/salida

1. Cambiar el nombre de un atributo.
2. Cambiar el orden de los atributos.
3. Borrar atributos.
4. Para atributos numéricos, normalizar a $[0, 1]$.
5. Cambiar el tipo del atributo de numérico a numérico y viceversa (siempre que el nominal tenga valores numéricos).

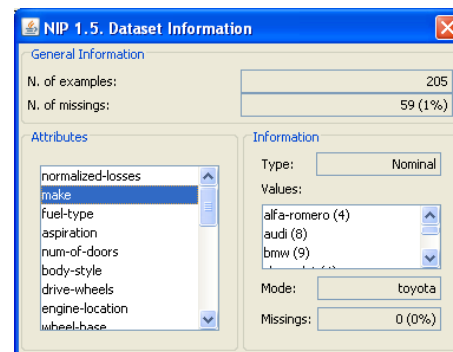


Figura 3: Información del conjunto de datos

4.3 INTRODUCCIÓN DE IMPERFECCIÓN

En esta subsección veremos las distintas opciones que ofrece N_ip1.5 para introducir imperfección en los conjuntos de datos.

4.3.1 Valores desconocidos

N_ip1.5 permite introducir imperfección al conjunto de datos en forma de valores desconocidos. Dichos valores se pueden introducir para todos los atributos o a un subconjunto de ellos. El número de valores desconocidos a añadir depende de un porcentaje establecido por el usuario, que puede ser distinto para cada atributo (Figura 4). La introducción de valores desconocidos es útil si se quiere corromper el conjunto de datos de manera aleatoria; no obstante, hay que tener en cuenta

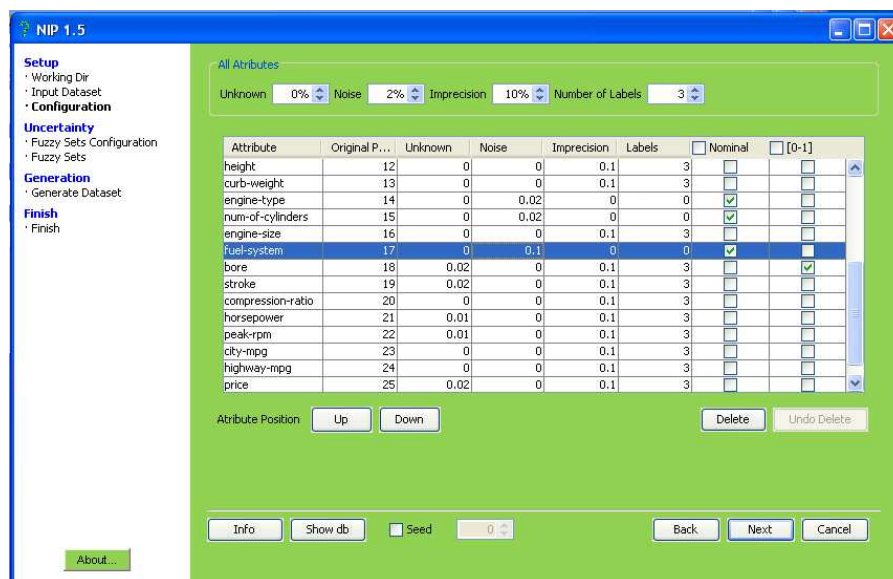


Figura 4: Pantalla de configuración

que en un conjunto de datos real con imperfección, los valores desconocidos pueden seguir un patrón concreto que no es posible simular.

Por otro lado, N_ip1.5 no sólo introduce valores desconocidos en el conjunto de datos, sino que también aplica un tratamiento a valores desconocidos ya existentes. Si el algoritmo a usar no soporta valores desconocidos, N_ip1.5 puede reemplazar dichos valores por la moda, media, intervalos o fuzzy sets (Figura 5).

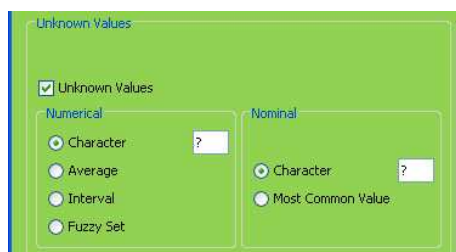


Figura 5: Gestión valores desconocidos

puede indicar el usuario, mediante un formato predefinido, o bien se puede dejar que N_ip1.5 seleccione la mejor partición para el atributo basándose en un algoritmo de Árbol de Decisión Fuzzy (Figura 6). Esta imprecisión se puede representar de diversas formas:

- Con el nombre de la etiqueta lingüística.
- Con los valores del fuzzy set trapezoidal (V_1, V_2, V_3, V_4).
- Con los valores y el valor asociado ($V_1/\mu_1(V_1), V_2/\mu_2(V_2), V_3/\mu_3(V_3), V_4/\mu_4(V_4)$).

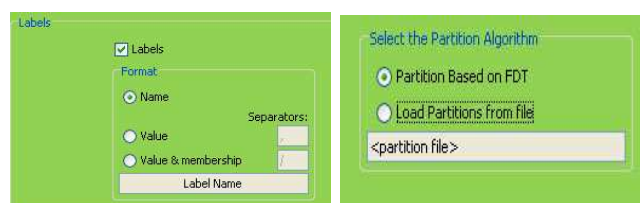


Figura 6: Introduciendo fuzzy sets

4.3.2 Imprecisión

Como hemos visto en la sección 2.2, la teoría de la posibilidad [6] y la teoría de intervalos sirven para representar información imprecisa. N_ip1.5 permite añadir imprecisión al conjunto de datos sustituyendo los valores de un atributo numérico por un fuzzy set que contenga dicho valor. El usuario decide qué atributos incluyen este tipo de imprecisión y en qué porcentaje (Figura 4). Para realizar esta transformación, N_ip1.5 necesita de una partición fuzzy para cada atributo al que se vaya a aplicar dicha alteración. La partición la

4.3.3 Ruido

El ruido existente en los datos viene determinado tanto por el tipo de valores de los atributos como por la exactitud en la medida de dichos valores. Con N_ip1.5 se puede introducir ruido para todos los atributos nominales o para un subconjunto de ellos. El porcentaje de ruido depende del valor indicado por el usuario, que puede ser distinto para cada atributo (Figura 4). Este tipo de ruido cambia el valor original del atributo por otro valor que esté en el conjunto de posibles valores de dicho atributo.

En el caso de los atributos numéricos, N_ip1.5 permite introducir ruido gaussiano indicando la media μ y la desviación típica σ de la distribución para cada atributo. La representación en el conjunto de datos generada puede variar entre intervalos, fuzzy sets o funciones gaussianas (Figura 7).

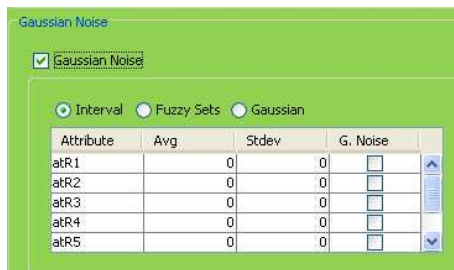


Figura 7: Introduciendo ruido en atributo numérico

5 CONCLUSIONES

La falta de conjuntos de datos con representación de la imperfección inherente a cualquier recogida de datos hace difícil el desarrollo de técnicas de extracción de conocimiento que gestionen datos imperfectos. En este trabajo hemos descrito una herramienta software Java, llamada N_ip1.5, diseñada como una herramienta de ayuda a la generación y gestión de conjuntos de datos que sirva como marco común para la prueba y comparación de técnicas y algoritmos de minería de datos y aprendizaje.

Hemos presentado las principales características de esta herramienta software y hemos destacado que N_ip1.5 es capaz de incluir en un conjunto de datos más tipos de imperfección que las herramientas actuales. Además N_ip1.5 acepta formatos de entrada/salida predefinidos y personalizados, con lo cual la convierte en un software más flexible.

Agradecimientos

Los autores agradecen la financiación proporcionada por el proyecto TIN2008-06872-C04-03 del MICINN, y por el proyecto 04552/GERM/06 y al programa de FPI de la “Agencia de Ciencia y Tecnología” de la Región de Murcia, de España.

Referencias

- [1] J. Alcalá-Fdez, L. Sánchez, S. García, M.J. del Jesus, S. Ventura, J.M. Garrell, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, J.C. Fernández and F. Herrera, KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems. *Soft Computing* **13**(3) (2009) 307–318.
- [2] A. Asuncion, and D.J. Newman, UCI Machine Learning Repository, <http://www.ics.uci.edu/~mllearn/MLRepository.html>, Irvine, CA: University of California, School of Information and Computer Science, 2007.
- [3] M.R Berthold, N. Cebron, F. Dill, T.R. Gabriel, T. Kötter, T. Meinl, P. Ohl, C. Sieb, K. Thiel and B. Wiswedel, KNIME: The Konstanz Information Miner, in *Data Analysis, Machine Learning and Applications*, eds. C. Preisach, H. Burkhardt, L. Schmidt-Thieme and R. Decker (Springer, Berlin, 2008), pp. 319–326.
- [4] J.M. Cadenas, J.V. Carrillo, M.C. Garrido, R. Martínez-España, E. Muñoz, NIP 1.5 - A tool to handle imperfect information on data-sets. Murcia University, 2008. <http://heurimind.inf.um.es/NIP/index.htm>
- [5] E. Diday and M. Noirhomme-Fraiture, *Symbolic Data Analysis and the SODAS Software*, (Wiley-Interscience, New York, 2008).
- [6] D. Dubois and H. Prade. *Possibility Theory: An Approach to Computerized Processing of Uncertainty* (Plenum Press, New York, 1988)
- [7] R. Hickey, Noise Modeling and Evaluating Learning from Examples, *Artificial Intelligence* **82**(1-2) (1996) 157–179.
- [8] B. Klauer and J.D. Brown, Conceptualising imperfect knowledge in public decision making: ignorance, uncertainty, error and risk situations, *Environmental Research, Engineering and Management*, **27**(1) (2004) 124–128.
- [9] X. Llorà, E2K: Evolution to knowledge, *SIGEVolution* **1**(3) (2006) 10–16.
- [10] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz and T. Euler, YALE: Rapid Prototyping for Complex Data Mining Tasks, in *Proc. 12th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining (KDD-06)* (2006) 935–940.
- [11] K. Morik, M. Scholz, The MiningMart Approach to Knowledge Discovery in Databases, in *Intelligent Tech. for Information Analysis*, eds. N. Zhong, J. Liu (Springer, Berlin, 2004) pp. 47–65.
- [12] J. Rushing, R. Ramachandran, U. Nair, S. Graves, R. Welch and H. Lin, ADaM: a data mining toolkit for scientists and engineers, *Computers & geosciences* **31** (5) (2005) 607–618.
- [13] I.H. Witten, E. Frank *Data Mining: Practical machine learning tools and techniques, 2nd Edition*, (Morgan Kaufmann, San Francisco, 2005).

KEEL: UNA HERRAMIENTA SOFTWARE PARA EL ANÁLISIS DE SISTEMAS DIFUSOS EVOLUTIVOS

Joaquín Derrac¹ Alberto Fernández¹ Julián Luengo¹ Salvador García²
Luciano Sánchez³ Jesús Alcalá¹ Francisco Herrera¹

¹ Dept. de Ciencias de la Computación e Inteligencia Artificial, CITIC-UGR, Universidad de Granada,
{alberto,julianlm,jderrac,jalcala,herrera}@decsai.ugr.es

² Dept. de Informática, Universidad de Jaén, sglopez@ujaen.es

³ Dept. de Informática, Universidad de Oviedo, luciano@uniovi.es

Resumen

En este trabajo se presenta *KEEL*, una herramienta software no comercial, capaz de abordar una amplia gama de problemas de minería de datos (regresión, clasificación, asociación, agrupamiento ...) mediante la generación de experimentos estándar y educativos. Dispone de una amplia librería de algoritmos de Sistemas Difusos Evolutivos basados en diferentes esquemas: Michigan, Pittsburgh, IRL y GCCL. También dispone de módulos de tratamiento de datos y análisis estadísticos. A modo de ejemplo, mostramos un caso de estudio sobre el análisis de dos propuestas diferentes de aprendizaje de Sistemas Difusos.

Palabras Clave: Herramienta software, Minería de Datos, Java, Sistemas Difusos Evolutivos.

1 INTRODUCCIÓN

La definición automática de los Sistemas Difusos (SD) puede ser considerada como un proceso de optimización o de búsqueda donde, actualmente, los Algoritmos Evolutivos (AE)[10] (en particular los Algoritmos Genéticos (AG)) son considerados la técnica de búsqueda global más conocida y utilizada. Además, la codificación genética que utilizan les permite introducir conocimiento *a priori* a partir del cual iniciar la búsqueda. Por esta razón, los Algoritmos Evolutivos han sido aplicados satisfactoriamente en los últimos años, dando lugar a la aparición de los llamados Sistemas Difusos Evolutivos (SDEs) [7].

Los SDEs han sido ampliamente utilizados en numerosos campos [3, 18]. Sin embargo, su empleo requiere cierta experiencia en programación, y una gran cantidad de esfuerzo y tiempo para escribir un programa que los implemente (suelen ser algoritmos

sofisticados). Este tedioso trabajo debe ser realizado antes de poder centrar la atención en los aspectos importantes del problema. En los últimos años se han desarrollado muchas herramientas software para aliviar esta tarea. Aunque muchas se distribuyen comercialmente (p.ej. Clementine¹, SPSS, varias *toolbox* de MATLAB²), algunas, como Weka [26] o Java-ML [1] están disponibles como software libre (recomendamos visitar el Directorio de Software KDnuggets³ como referencia). Estas últimas han adquirido una especial relevancia recientemente [20].

En esta contribución presentamos *KEEL* (*Knowledge Extraction based on Evolutionary Learning*)⁴, una herramienta software no comercial escrita en Java. *KEEL* permite al usuario emplear AEs en diferentes tipos de problemas de minería de datos: Regresión, clasificación, agrupamiento, asociación, etc., incluyendo una gran recopilación de los SDEs existentes. Además, ofrece varias ventajas:

- Primero, reduce el trabajo de programación. Incluye una gran librería con algoritmos de SDEs basados en diferentes paradigmas (Pittsburgh, Michigan, IRL y GCCL) y simplifica su integración con diferentes técnicas de preprocesamiento. Libera a los investigadores del esfuerzo de programación, permitiéndoles centrarse en el análisis de sus nuevos modelos de aprendizaje, en comparación con los ya existentes.
- Segundo, amplía el rango de posibles usuarios de los SDEs. Es un software fácil de usar, con una gran cantidad de propuestas ya implementadas, por lo que reduce considerablemente el nivel de conocimientos y experiencia requeridos a la hora de realizar un estudio que incluya SDEs, incluso para investigadores con menor experiencia.

¹<http://www.spss.com/clementine>

²<http://www.mathworks.com>

³<http://www.kdnuggets.com/software>

⁴<http://www.keel.es>

- Tercero, gracias al empleo de un paradigma estricto de orientación a objetos tanto en la librería como en la herramienta software, ambos pueden ser utilizados en cualquier máquina Java, con independencia del sistema operativo existente. Esto simplifica considerablemente el empleo de la herramienta por parte de cualquier investigador.

El presente trabajo se organiza de la siguiente forma: La Sección 2 describe las principales propuestas de aprendizaje con respecto a la forma de codificar la base de reglas. La Sección 3 presenta los principales SDEs existentes en KEEL. La Sección 4 describe las principales características y módulos de KEEL. En la Sección 5, se muestra un caso de estudio para ilustrar el empleo de KEEL. Finalmente, la Sección 6 muestra algunas conclusiones y trabajo futuro.

2 APRENDIZAJE EVOLUTIVO

Aunque los AEs no fueron diseñados específicamente para aprendizaje, sino como algoritmos de búsqueda global, su empleo ofrece varias ventajas. Muchas metodologías están basadas en la búsqueda de un buen modelo dentro del espacio de modelos posibles. En este sentido, los AEs son muy flexibles, ya que pueden manejar diferentes representaciones. Los procesos de aprendizaje evolutivo cubren diferentes niveles de complejidad con respecto a los cambios estructurales del SDE, desde el caso más simple de optimización de parámetros hasta el más alto nivel de complejidad de aprender una base de reglas completa, mediante la co-operación o competición entre cromosomas y la codificación empleada.

Existe una amplia gama de posibilidades para la tarea de aprender una base de reglas. Los métodos de aprendizaje evolutivo siguen dos enfoques distintos para codificar reglas en una población de individuos:

- “Cromosoma = Conjunto de reglas”, o enfoque Pittsburgh, en el que cada individuo representa un conjunto de reglas [19]. En este caso, los cromosomas evolucionan bases de reglas completas, y compiten entre sí durante la búsqueda. GABIL es una propuesta que sigue este enfoque [17].
- “Cromosoma = Regla”, en el que cada individuo representa una única regla, y el conjunto de reglas completo se obtiene combinando varios individuos de una población (cooperación) o a partir de varias ejecuciones (competición).

Dentro el enfoque “Cromosoma = Regla”, existen:

- El enfoque Michigan, en el que cada individuo codifica una única regla. Este tipo de sistemas suelen denominarse sistemas de aprendizaje de clasificadores [14]. Son sistemas de paso de mensajes, basados en reglas

que emplean aprendizaje por refuerzo y un AG para aprender reglas que guíen el aprendizaje en un problema dado. El AG detecta nuevas reglas que reemplazan a las peores mediante competición entre cromosomas en el proceso evolutivo. XCS [25] es un ejemplo de este enfoque.

- El enfoque IRL (*Iterative Rule Learning*, aprendizaje iterativo de reglas), en el que cada cromosoma representa una regla. Los cromosomas compiten entre sí, tomándose el mejor en cada ejecución del AG. La solución global se forma a partir de ellos, ejecutando múltiples veces el algoritmo. SIA [23] es una propuesta que sigue este enfoque.
- El enfoque GCCL (*Genetic Cooperative-Competitive Learning*, aprendizaje genético cooperativo-competitivo), en el que la base de reglas se codifica a partir de la población completa o de un subconjunto de ella. En este modelo, los cromosomas compiten y cooperan simultáneamente. LOGENPRO [27] es un ejemplo de este enfoque.

Todos ellos han sido empleados para el aprendizaje de base de reglas. Es posible encontrar varios ejemplos empleando reglas difusas o intervalares en KEEL⁵.

3 ALGORITMOS EVOLUTIVOS PARA SISTEMAS DIFUSOS EN KEEL

En los últimos años, el número de artículos publicados en el área los SDEs se ha incrementado, debido a su potencial. Al contrario que las redes neuronales, el agrupamiento, la inducción de reglas y otros muchos enfoques, los AGs ofrecen un mecanismo para codificar y evolucionar operadores de agregación para los antecedentes de las reglas, diferentes semánticas de reglas, operadores de agregación de base de reglas y métodos de *defuzzificación*. Por ello, los AGs siguen siendo hoy una de las pocas técnicas disponibles para diseñar y optimizar por completo los SDs basados en reglas, permitiendo a los investigadores decidir qué componentes son fijos y cuáles deben evolucionar durante el proceso de búsqueda.

KEEL divide las propuestas de SDEs en dos partes, ajuste (*tuning*) y aprendizaje:

- Ajuste evolutivo. Se aplica un proceso de ajuste *a posteriori* sobre la base de conocimiento (BC) para mejorar el rendimiento del SD.
- Aprendizaje evolutivo. Consiste en aprender componentes de la BC (incluyendo aspectos como obtener un mecanismo de inferencia adaptativo).

⁵<http://www.keel.es/algorithms.php>

Es posible clasificar los SDEs existentes en KEEL⁵ con respecto a estas partes o a los componentes del SD involucrados en el proceso evolutivo [12]:

- Ajuste evolutivo de parámetros de la BC: P.ej. un método de ajuste para la obtención de reglas de control difusas de alto rendimiento por medio de AGs [13].
- Aprendizaje evolutivo de pesos para reglas: P.ej. un AG para aprendizaje de pesos y selección de reglas [4].
- Aprendizaje evolutivo de la base de reglas: P.ej. el método Pittsburgh pionero propuesto por Thrift [21], SLAVE [11], y otros.
- Aprendizaje evolutivo de SDs TSK: P.ej. un proceso evolutivo con identificación local de prototipos para obtener un conjunto inicial de reglas TSK locales basadas en semántica [2].
- Aprendizaje simultáneo de componentes de la BC: P.ej. un proceso multi-etapa híbrido de AG y estrategias de evolución para el diseño aproximado de SDs [6].
- La primera contribución en el área de selección evolutiva de reglas, propuesta por Ishibuchi [15].

4 DESCRIPCIÓN DE KEEL

KEEL es una herramienta software para la preparación de AEs para problemas de minería de datos. Permite ejecutarlos dentro del propio entorno (ejecución *on-line*) o generarlos para una ejecución posterior en distintas máquinas (ejecución *off-line*). La versión actual de KEEL está compuesta por los siguientes módulos (ver Figura 1):



Figura 1: Pantalla principal de KEEL

- *Tratamiento de datos* (Data Management): Este módulo contiene una serie de herramientas de tratamiento de datos: Importación, exportación, edición y visualización de datos, aplicación de transformaciones, etc.

- *Experimentos* (Experiments): Este módulo genera procedimientos de análisis y evaluación automática de algoritmos *off-line*, proporcionando numerosas opciones: Tipo de validación, tipo de aprendizaje (clasificación, regresión, aprendizaje no-supervisado), etc.
- *Educacional* (Educational): Este módulo realiza la ejecución de algoritmos *on-line*. Tiene una estructura similar al módulo anterior, pero permite diseñar experimentos para ser ejecutados paso a paso, con propósitos educativos.

Esta estructura hace que KEEL sea interesante para distintos tipos de usuarios, dependiendo de sus necesidades. A nivel general, las principales características de KEEL son las siguientes:

- Incluye algoritmos de aprendizaje de modelos predictivos, de preprocesamiento (transformación de datos, discretización, selección de instancias y selección de características) y postprocesamiento, con especial atención a las propuestas evolutivas.
- Posee una librería estadística para analizar resultados de algoritmos. Dicha librería contiene tests estadísticos para analizar la bondad de los resultados obtenidos y también para realizar comparaciones paramétricas y no paramétricas.
- Incluye una Librería de Programación de Algoritmos Evolutivos en Java, *Java Class Library for Evolutionary Computation* (JCLEC) [22].
- Ofrece al usuario una interfaz amigable, orientada al análisis de algoritmos.
- Permite crear experimentaciones conteniendo múltiples conjuntos de datos y algoritmos conectados entre sí. Los experimentos son generados mediante scripts independientes de la interfaz de usuario, para permitir una ejecución *off-line* en la misma u otras máquinas.
- KEEL también permite ejecutar experimentos en modo *on-line*, dando un soporte educativo a la tarea de comprender el funcionamiento de los algoritmos incluidos.

A continuación describimos brevemente las principales características del módulo *Experimentos*. Este módulo es el encargado de generar los ficheros y la estructura de directorios requerida para ejecutarlo en cualquier máquina con Java (ver Figura 2).

Los experimentos son modelados gráficamente, basados en flujos de datos y representados mediante grafos y conexiones arco-nodo. Permiten escoger el tipo de validación a emplear (validación cruzada de k-particiones o validación cruzada 5x2) y el tipo de aprendizaje (regresión, clasificación o no supervisado).

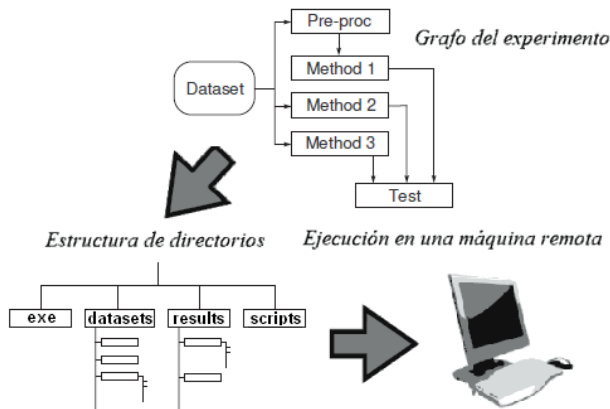


Figura 2: Diseño de experimentos

Después se seleccionan los conjuntos de datos, se arrastran los métodos seleccionados al espacio de trabajo y se establecen conexiones entre métodos y datos (incluyendo, si se desea, test estadísticos de análisis de resultados). En cualquier momento, cualquier componente puede ser configurado haciendo doble click en su nodo correspondiente. La Figura 3 muestra un ejemplo de experimento de clasificación, siguiendo la metodología MOGUL [8] y empleando un método de informe para obtener un resumen de resultados. También se muestra la ventana de configuración del método MOGUL en dicha figura.

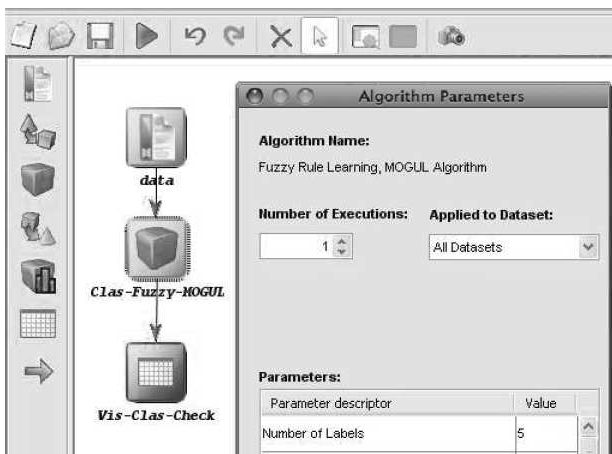


Figura 3: Ejemplo de experimento y de ventana de configuración de un método.

Cuando el experimento ha sido diseñado, el usuario puede escoger entre guardarlo en un fichero XML u obtener un fichero .zip. Éste último contendrá los ficheros requeridos para ejecutar el experimento en una máquina independiente con Java (los conjuntos de datos, ficheros .jar de los algoritmos, ficheros XML de configuración y la herramienta *RunKeel*, un entorno

simple de scripting que interpreta scripts de configuración de KEEL en formato XML).

5 CASO DE ESTUDIO

En esta sección presentamos un ejemplo de creación de experimentos en KEEL. Este estudio está centrado en la comparación de dos algoritmos de aprendizaje basados en reglas difusas sobre 12 problemas de clasificación. La Tabla 1 muestra dichos problemas.

Tabla 1: Problemas empleados.

Problema	#Ejemplos	#Atr.	#Clases
Bupa	345	6	2
Cleveland	297	13	5
Ecoli	336	7	8
Glass	214	9	7
Haberman	306	3	2
Iris	150	4	3
Monk-2	432	6	2
New-thyroid	215	5	3
Pima	768	8	2
Vehicle	846	18	4
Wine	178	13	3
Wisconsin	683	9	2

Los métodos considerados son el algoritmo SLAVE [11] y el algoritmo Chi et al. [5] con pesos en la reglas [16] (denominado algoritmo Chi-RW). El algoritmo SLAVE es un algoritmo de aprendizaje inductivo de la base de reglas, basado en el enfoque IRL y el algoritmo Chi-RW es una adaptación del algoritmo de Wang-Mendel [24] a problemas de clasificación.

Para desarrollar el experimento se ha considerado un esquema de validación cruzada de 10 particiones. Debido a que SLAVE es un método probabilístico, ha sido ejecutado 5 veces por cada partición (un total de 50 ejecuciones).

Las particiones lingüísticas iniciales están compuestas por *tres términos lingüísticos*, con funciones de pertenencia triangulares uniformemente distribuidas. Los valores considerados para los parámetros de entrada de cada método son:

- SLAVE: *Tamaño de la población* = 100, *Iteraciones sin cambios* = 500, *Probabilidad de mutación* = 0.01, *Probabilidad de cruce* = 1.0.
- Chi-RW: *Cálculo del grado de compatibilidad*: T-norma producto, *Pesos en reglas*: Factor de certeza penalizado, *Combinación del grado de compatibilidad y de pesos en reglas*: T-norma producto, *Método de inferencia*: mejor regla.

Para realizar este experimento en KEEL, primero seleccionamos la opción *Experiments* del menú principal de KEEL, definimos el experimento como problema de clasificación e indicamos el procedimiento de validación cruzada de 10 particiones. Después, el primer paso de configuración del grafo del experimento consiste en seleccionar los conjuntos de datos de la Tabla

1. Las particiones generadas por KEEL son estáticas, por lo que permiten ejecutar experimentos similares posteriormente, sin alterar las particiones realizadas. Opcionalmente, el usuario puede crear sus propias particiones mediante el módulo de *Tratamiento de datos*.

El grafo de la Figura 4 representa el flujo de datos y resultados entre algoritmos y técnicas estadísticas. Un nodo puede representar un flujo inicial de datos, un algoritmo de preprocesamiento o postprocesamiento, un algoritmo de aprendizaje, un test o un módulo de visualización de resultados. Todos sus parámetros pueden ser ajustados haciendo doble click en el nodo (ver Sección 4), incluyendo el número de ejecuciones para algoritmos probabilísticos (5 en el caso de SLAVE). Por otro lado, los arcos que conectan dos nodos representan una relación entre ellos (intercambio de datos o de resultados en entrenamiento y test).

Las salidas de los algoritmos de aprendizaje se emplean como entradas para los métodos de visualización y test. El módulo *Vis-Clas-Tabular* recibe dichos resultados y genera ficheros de salida con varias medidas de rendimiento calculadas a partir de ellos. La Figura 4 también muestra el nodo *Stat-Clas-Wilcoxon*, que realiza una comparación estadística entre resultados obtenidos sobre múltiples conjuntos de datos, el test de Wilcoxon [9].

Una vez que el grafo ha sido definido, podemos guardar el experimento en un fichero .zip para su ejecución *off-line*. Dicho experimento estará compuesto por un conjunto de scripts XML y un programa .jar que los ejecuta. En el directorio *results* se almacenarán los resultados de cada método durante su ejecución. P. ej., los ficheros situados en el directorio asociado al algoritmo de aprendizaje difuso contendrán BCs. En el caso de los métodos de visualización, sus directorios contendrán los ficheros de resultados. Los resultados obtenidos del análisis de los métodos se muestran en la Tabla 2, donde *#Reg* es el número medio de reglas y *AccEnt* y *AccTst* son, respectivamente, el porcentaje de acierto medio obtenido sobre los datos de entrenamiento y test.

En el caso de los módulos de test, el directorio *Stat-*

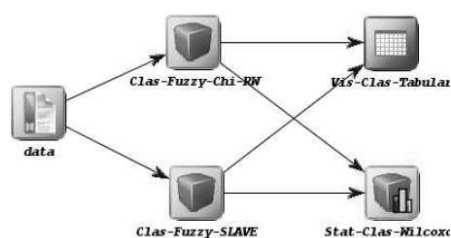


Figura 4: Grafo del experimento del caso de estudio

Tabla 2: Resultados obtenidos.

Problema	CHI-RW			SLAVE		
	<i>AccEnt</i>	<i>AccTst</i>	<i>#Reg</i>	<i>AccEnt</i>	<i>AccTst</i>	<i>#Reg</i>
Bupa	59.87	57.87	43.3	60.60	58.28	3.9
Cleveland	91.25	39.09	230.5	79.05	53.52	39.6
Ecoli	79.53	78.33	43.5	82.75	79.10	11.7
Glass	65.99	60.04	27.1	71.70	62.16	12.8
Haberman	74.26	73.19	16.7	74.90	74.35	2.9
Iris	93.78	94.00	14.7	96.98	94.86	3.3
Monk-2	100.0	48.84	301.8	67.36	67.23	1.3
New-thyroid	85.94	84.24	18.4	89.82	87.99	3.9
Pima	75.62	72.40	105.2	75.45	74.44	4.7
Vehicle	65.92	60.77	227.8	66.31	60.18	20.8
Wine	98.75	92.68	121.1	94.60	90.42	4.3
Wisconsin	98.08	91.21	224	97.16	95.72	5.1
Average	82.42	71.06	114.51	79.72	74.85	9.53

Clas-Wilcoxon contendrá los resultados del test estadístico. Dichos módulos producen su salida en formato de texto y $\text{\LaTeX}$, generando las tablas asociadas al test. Para este estudio se ha obtenido la Tabla 3, donde los rangos positivos corresponden a SLAVE y los negativos a Chi-RW.

Tabla 3: Test estadístico.

Comparación	R^+	R^-	p-valor
SLAVE vs. Chi-RW	69.0	9.0	0.019607

Analizando los resultados de las Tablas 2 y 3 podemos destacar que, aunque el algoritmo SLAVE no obtiene la mejor precisión en entrenamiento en todos los problemas, su precisión en test es mejor en 11 de los 12 conjuntos, y los modelos obtenidos tienen un menor número de reglas. Además, el análisis estadístico (test de Wilcoxon de pares) obtiene que el algoritmo SLAVE mejora claramente al algoritmo Chi-RW, asumiendo un alto nivel de significancia $p = 0.0196$.

6 CONCLUSIONES

KEEL es una herramienta software capaz de preparar adecuadamente AEs para problemas de minería de datos, incluyendo algoritmos de SDEs. Esta herramienta libera a los investigadores de la mayoría del trabajo técnico y les permite centrarse en el análisis de sus nuevos algoritmos en comparación con los ya existentes, permitiendo que incluso investigadores con conocimientos básicos en Lógica Difusa y Computación Evolutiva puedan emplear los SDEs en su trabajo.

Hemos mostrado un caso de estudio para ilustrar el proceso de generación de experimentos en KEEL, incluyendo el empleo de tests estadísticos.

La herramienta KEEL está siendo continuamente actualizada y mejorada. Actualmente, estamos desarrollando herramientas de visualización para los módulos on-line y off-line. Además, se está trabajando en el desarrollo de un repositorio de conjuntos de datos que

incluya particiones y resultados de la aplicación de algoritmos sobre dichos conjuntos, *KEEL-dataset*⁶.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio Español de Educación y Ciencia bajo los proyectos TIN2008-06681-C06-01 y TIN2008-06681-C06-04.

Referencias

- [1] T. Abeel, Y.V. de Peer, Y. Saeys. Java-ML: A machine learning library. *Journal of Machine Learning Research*, 10, Pág. 931-934, 2009.
- [2] R. Alcalá, J. Alcalá-Fdez, J. Casillas, O. Cordon, F. Herrera. Local Identification of Prototypes for Genetic Learning of Accurate TSK Fuzzy Rule-Based Systems. *International Journal of Intelligent Systems*, 22:9, Pág. 909-941, 2007.
- [3] R. Alcalá, J. Casillas, O. Cordon, A. González, F. Herrera. A Genetic Rule Weighting and Selection Process for Fuzzy Control of Heating, Ventilating and Air Conditioning Systems. *Engineering Applications of Artificial Intelligence*, 18:3, Pág. 279-296, 2005.
- [4] R. Alcalá, O. Cordon, F. Herrera. Combining Rule Weight Learning and Rule Selection to Obtain Simpler and More Accurate Linguistic Fuzzy Models. *in Modelling with Words*, LNCS Vol. 2873, Pág. 44-63, 2003.
- [5] Z. Chi, J. Wu, H. Yan. Handwritten Numeral Recognition Using Self-Organizing Maps and Fuzzy Rules. *Pattern Recognition*, 28:1, Pág. 59-66, 1995.
- [6] O. Cordon, F. Herrera. Hybridizing Genetic Algorithms with Sharing Scheme and Evolution Strategies for Designing Approximate Fuzzy Rule-Based Systems. *Fuzzy Sets and Systems*, 118:2, Pág. 235-255, 2001.
- [7] O. Cordon, F. Herrera, F. Hoffmann, L. Magdalena. *Genetic fuzzy systems. Evolutionary tuning and learning of fuzzy knowledge bases*. World Scientific, Singapore, 2001.
- [8] O. Cordon, M.J. del Jesus, F. Herrera, M. Lozano. MOGUL: A Methodology to Obtain Genetic fuzzy rule-based systems Under the iterative rule Learning approach. *International Journal of Intelligent Systems*, 14:11, Pág. 1123-1153, 1999.
- [9] J. Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7, Pág. 1-30, 2006.
- [10] A.E. Eiben, J.E. Smith. *Introduction to Evolutionary Computing*. Springer-Verlag, Berlin, 2003.
- [11] A. González, R. Perez. SLAVE: A genetic learning system based on an iterative approach. *IEEE Transactions on Fuzzy Systems*, 7:2, Pág. 176-191, 1999.
- [12] F. Herrera. Genetic Fuzzy Systems: Taxonomy, Current Research Trends and Prospects. *Evolutionary Intelligence*, 1, Pág. 27-46, 2008.
- [13] F. Herrera, M. Lozano, J.L. Verdegay. Fuzzy Logic Controllers by Genetic Algorithms. *International Journal of Approximate Reasoning*, 12, Pág. 299-315, 1995.
- [14] J.H. Holland, J.S. Reitman. Cognitive systems based on adaptive algorithms. *SIGART Bull*, 63, Pág. 49-49, 1977.
- [15] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka. Selecting Fuzzy If-Then Rules for Classification Problems Using Genetic Algorithms. *IEEE Transactions on Fuzzy Systems*, 3:3, Pág. 260-270, 1995.
- [16] H. Ishibuchi, T. Yamamoto. Rule weight specification in fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, 13:4, Pág. 428-435, 2005.
- [17] K.A. de Jong, W.M. Spears, D.F. Gordon. Using genetic algorithms for concept learning. *Machine Learning*, 13, Pág. 161-188, 1993.
- [18] J.S. Shieh, M.H. Kao, C.C. Liu. Genetic fuzzy modelling and control of bispectral index (BIS) for general intravenous anaesthesia. *Medical Engineering and Physics*, 28:2, Pág. 134-148, 2006.
- [19] S. Smith. *A learning system based on genetic algorithms*. Ph.D. thesis, University of Pittsburgh, 1980.
- [20] S. Sonnenburg, M.L. Braun, Ch.S. Ong, S. Bengio, L. Bottou, G. Holmes, Y. LeCun, K-R. Müller, F. Pereira, C.E. Rasmussen, G. Rätsch, B. Schölkopf, A. Smola, P. Vincent, J. Weston, R.C. Williamson. The Need for Open Source Software in Machine Learning. *Journal of Machine Learning Research*, 8, Pág. 2443-2466, 2007.
- [21] P. Thrift. Fuzzy logic synthesis with genetic algorithms. *in 4th International Conference on Genetic Algorithms (ICGA91)*, Pág. 509-513, 1991.
- [22] S. Ventura, C. Romero, A. Zafra, J.A. Delgado, C. Hervás. JCLEC: A Java framework for Evolutionary Computation. *Soft Computing*, 12:4, Pág. 381-392, 2008.
- [23] G. Venturini. SIA: a supervised inductive algorithm with genetic search for learning attribute based concepts. *in European Conference on Machine Learning*, LNCS Vol. 669, Pág. 280-296, 1993.
- [24] L. X. Wang, J. M. Mendel. Generating Fuzzy Rules by Learning from Examples. *IEEE Transactions on Systems, Man and Cybernetics*, 22:6, Pág. 1414-1427, 1992.
- [25] S. W. Wilson. Classifier Fitness Based on Accuracy. *Evolutionary Computation*, 3:2, Pág. 149-175, 1995.
- [26] I. H. Witten, E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Second edition, Morgan Kaufmann, 2005.
- [27] M.L. Wong, K.S. Leung. *Data mining using grammar based genetic programming and applications*. Kluwer Academic Publishers, Norwell, 2000.

⁶<http://www.keel.es/datasets.php>

XFHL: UNA HERRAMIENTA DE INDUCCIÓN DE SISTEMAS DIFUSOS JERÁRQUICOS

Sergio Cala Cordero¹ Francisco José Moreno Velo²

¹Departamento de Tecnologías de la Información, Universidad de Huelva, sergio.cala@alu.uhu.es

²Departamento de Tecnologías de la Información, Universidad de Huelva, francisco.moreno@dti.uhu.es

Resumen

Este trabajo presenta una herramienta de generación automática de sistemas difusos jerárquicos llamada *Xfhl*. La herramienta toma como entrada un conjunto de datos de entrenamiento y busca la mejor descomposición jerárquica en términos de módulos difusos de dos entradas y una salida. El funcionamiento de *Xfhl* se basa en una búsqueda exhaustiva entre todas las posibles descomposiciones modulares. Para evaluar cada estructura se utiliza el error cometido por la estructura tras un proceso de optimización paramétrica. La herramienta está integrada en el entorno Xfuzzy de desarrollo de sistemas difusos.

Palabras Clave: Sistemas Difusos Jerárquicos, Software para SoftComputing.

1 INTRODUCCIÓN

A medida que los sistemas difusos se han afianzado como forma de modelar comportamientos no lineales el grado de complejidad de los problemas tratados ha ido aumentando. En los sistemas difusos estándar, para un problema con n variables de entrada descritas por m etiquetas lingüísticas, el número de reglas del sistema difuso resulta m^n . Este crecimiento exponencial provoca, en la práctica, que para un número de variables alto el número de reglas es tan grande que la interpretabilidad del sistema se hace imposible. Este problema, que no es exclusivo de los sistemas difusos, se conoce como *la maldición de la dimensionalidad* [2].

Una de las formas de disminuir el número de reglas y, por tanto, de aumentar la interpretabilidad, es descomponer el sistema difuso en módulos más simples, lo que se conoce como *sistemas difusos jerárquicos* (SDJ). Existen numerosas propuestas de diseño de este tipo de sistemas [13]. Algunas de ellas consisten en identificar partes comunes del conjunto de reglas y crear módulos que generen estas

partes comunes, lo que se conoce como SDJ limpios [1][9]. En otras propuestas el nivel de jerarquía de cada módulo se refiere a un aumento de la granularidad de las variables [6].

El tipo más tradicional de SDJ es aquel en el que cada módulo es un sistema difuso completo que relaciona a un conjunto reducido de variables, que pueden ser variables de entrada del sistema global o variables internas generadas como salidas de otros módulos [11]. En este tipo de SDJ el número de reglas es

$$\text{Nº de reglas} = \sum_{i=1}^L m^{c_i} \quad (1)$$

donde L es el número de módulos y c_i el número de entradas del módulo i -ésimo. Si los módulos no comparten entradas el SDJ tiene forma de árbol. Si los módulos pueden compartir entradas el SDJ tiene forma de grafo dirigido acíclico. Centrándonos en sistemas con una única salida (MISO) y con forma de árbol, el número de entradas de los módulos debe cumplir

$$\sum_{i=1}^L c_i = n + L - 1 \quad (2)$$

Si todos los módulos tienen el mismo número de entradas, c , el número de módulos del SDJ es

$$L = \frac{n-1}{c-1} \quad (3)$$

y el número de reglas total del SDJ es entonces

$$\text{Nº de reglas} = \left(\frac{n-1}{c-1} \right) \cdot m^c \quad (4)$$

lo que representa un crecimiento lineal del número de reglas en función del número de variables de entrada del

sistema frente al crecimiento exponencial que sufren los sistemas difusos estándar. El menor número de reglas totales se consigue para $c=2$, es decir, utilizando módulos de dos entradas. Es por eso que los SDJs más utilizados son los formados por este tipo de módulos, que se denominan habitualmente *fuzzy logic units* (FLUs). El uso de módulos de dos entradas tiene una ventaja adicional y es que su funcionamiento puede representarse mediante una superficie, lo que permite no sólo una interpretabilidad lingüística del módulo sino también una interpretabilidad gráfica del mismo.

Un aspecto muy importante de los SDJs es su capacidad de aproximación. Como es bien sabido, los sistemas difusos estándar son aproximadores universales [4]. Aunque existen propuestas de SDJs que han demostrado ser aproximadores universales [8][15], estas propuestas se refieren a sistemas difusos de tipo Takagi-Sugeno de orden muy alto (equivalente al número de entradas), por lo que el problema de la dimensionalidad no se resuelve sino que se desplaza del antecedente de las reglas a unos consecuentes de un alto grado de complejidad [16]. Por su parte, los sistemas difusos jerárquicos de tipo Mamdani no son aproximadores universales. En este caso el estudio se dirige a conocer cuales son las funciones que pueden ser aproximadas por un sistema difuso jerárquico de tipo Mamdani. En [17] se define el concepto de *función continua con estructura jerárquica separable* y se demuestra que un SDJ de tipo Mamdani puede aproximar estas funciones siempre que su estructura jerárquica corresponda a la de la función.

Para encontrar la estructura de un SDJ que permita aproximar una cierta función continua algunos autores han propuesto el uso de algoritmos genéticos [3][5]. En este trabajo se presenta una herramienta que realiza una búsqueda exhaustiva entre todas las descomposiciones modulares que puedan realizarse en base a módulos de dos entradas (FLUs). La estructura jerárquica generada por la herramienta es aquella que tras un proceso de optimización obtenga el menor error cuadrático sobre el conjunto de datos de entrenamiento.

2 ALGORITMO DE DESCOMPOSICIÓN MODULAR

La metodología propuesta se basa en realizar una búsqueda exhaustiva entre todas las posibles estructuras jerárquicas con n entradas formadas por módulos de dos entradas y una salida. Para generar todas estas estructuras jerárquicas se utiliza un algoritmo recursivo en el que las estructuras de n entradas se generan a partir de todas las estructuras de $n-1$ entradas.

El caso base del algoritmo es el formado por 2 entradas. En este caso sólo existe una estructura que es la formada

por un único módulo, tal y como se muestra en la Figura 1.a.

Dada una estructura jerárquica con n entradas, es posible generar un conjunto de estructuras con $n+1$ entradas incluyendo una nueva FLU en cada variable de la estructura. La Figura 1 muestra un ejemplo de las estructuras que pueden generarse a partir del caso base. La estructura de la Figura 1.b se obtiene incluyendo la nueva FLU en la posición de la variable $i0$. La Figura 1.c se obtiene incluyendo la FLU en la posición de la variable $i1$. La estructura mostrada en la Figura 1.d se obtiene incluyendo la nueva FLU en la posición de la variable de salida.

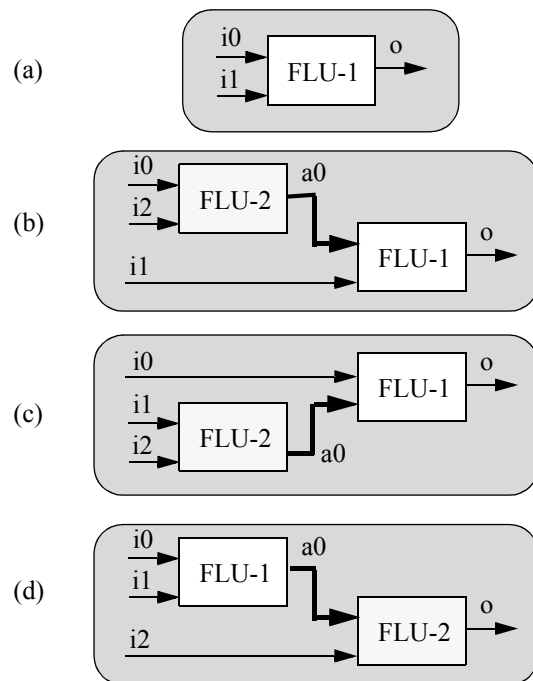


Figura 1: Algoritmo de descomposición modular para el caso de tres variables.

Teniendo en cuenta que cada estructura jerárquica con n entradas contiene $n-1$ FLUs, el número total de variables de la estructura es de $2n-1$. Por tanto, el número de estructuras con $n+1$ entradas es

$$S(n+1) = (2n-1) \cdot S(n) = \prod_{i=1}^n (2i-1) \quad (5)$$

La Tabla 1 muestra los valores del número de estructuras jerárquicas diferentes que se pueden generar utilizando módulos de dos entradas. Como se puede apreciar, este número crece exponencialmente con respecto al número de entradas. Para problemas con un número de entradas menor de 7 u 8 la búsqueda exhaustiva de la mejor estructura resulta factible. A partir de este tamaño el número de estructuras resulta demasiado elevado, lo que conduce a la utilización de técnicas de búsqueda aleatoria.

Tabla 1: Número de estructuras jerárquicas

Número de entradas	Número de estructuras
2	1
3	3
4	15
5	105
6	945
7	10.395
8	135.135
9	2.027.025

3 OPTIMIZACIÓN PARAMÉTRICA DE SISTEMAS DIFUSOS JERÁRQUICOS

De entre todas las estructuras jerárquicas en las que se puede descomponer un sistema de n entradas, la metodología propuesta selecciona aquella que mejor consigue aproximarse al conjunto de datos de entrenamiento. Para evaluar la capacidad de aproximación de cada estructura se utilizan algoritmos de optimización paramétrica basados en descenso por gradiente. El uso de algoritmos de descenso por gradiente para ajustar sistemas difusos jerárquicos ya ha sido propuesto en [14]. En nuestro caso la propuesta es utilizar los algoritmos RPROP [12] y Levenberg-Marquardt [7], que presentan una velocidad de convergencia mucho mayor que el utilizado en dicho trabajo. Estos algoritmos y muchos otros se encuentran ya integrados en el entorno Xfuzzy [10], lo que ha facilitado en gran medida el desarrollo de la herramienta *Xfhl* presentada en este trabajo.

Para seleccionar la mejor estructura jerárquica se evalúa el error cuadrático medio (ECM) que produce la estructura sobre el conjunto de datos de entrenamiento, que viene dado por la expresión

$$ECM = \frac{1}{E} \cdot \sum_{e=1}^E (\tilde{y}_e - y(\tilde{x}_e))^2 \quad (6)$$

donde E es el número de instancias del conjunto de datos, $\tilde{x}_e$ es el valor de las entradas de la instancia e -ésima, $\tilde{y}_e$ es el valor de la salida de la instancia e -ésima y la función $y(x)$ representa el comportamiento del sistema jerárquico.

Los parámetros de los que depende el comportamiento del sistema jerárquico corresponde al conjunto de parámetros que definen el comportamiento de cada FLU. Estos pará-

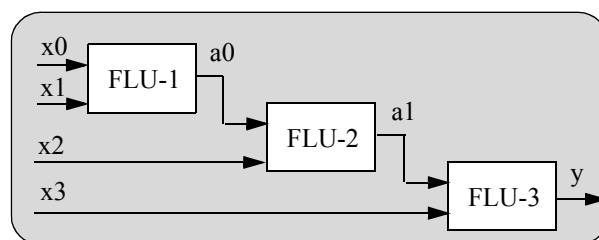


Figura 2: Ejemplo de estructura jerárquica

metros son los que definen las funciones de pertenencia de las variables de entrada del módulo y los relacionados con el cálculo de la salida. Para calcular la derivada del ECM respecto de cada uno de estos parámetros se utiliza la regla de la cadena. Por ejemplo, la derivada del ECM respecto de los parámetros del módulo FLU-1 de la Figura 2 viene dada por la expresión:

$$\frac{\partial ECM}{\partial p} = \sum_e \left(\frac{\partial ECM}{\partial y} \right) \cdot \left(\frac{\partial y}{\partial a1} \right) \cdot \left(\frac{\partial a1}{\partial a0} \right) \cdot \left(\frac{\partial a0}{\partial p} \right) \quad (7)$$

La derivada del ECM respecto de la salida del sistema para el ejemplo e -ésimo es la siguiente

$$\frac{\partial ECM}{\partial y_e} = \frac{2}{E} \cdot (y_e - \tilde{y}_e) \quad (8)$$

Para calcular el resto de derivadas parciales es necesario conocer el funcionamiento de las FLUs. Si el operador de conjunción es el producto y el método de concreción es la media difusa, el principio de transformación de la FLU es el siguiente:

$$y(x_1, x_2) = \frac{\sum_{i=1}^m \sum_{j=1}^m \mu_{1i}(x_1) \cdot \mu_{2j}(x_2) \cdot c_{ij}}{\sum_{i=1}^m \sum_{j=1}^m \mu_{1i}(x_1) \cdot \mu_{2j}(x_2)} \quad (9)$$

donde $\mu_{1i}(x_1)$ son las funciones de pertenencia asociadas a la variable x_1 , $\mu_{2j}(x_2)$ son las funciones de pertenencia asociadas a la variable x_2 , c_{ij} es el centroide de la conclusión de la regla “ $x_1 = \text{label-}i$ AND $x_2 = \text{label-}j$ ” y m es el número de etiquetas lingüísticas asociadas a cada variable. A partir de la expresión anterior se puede calcular la derivada parcial de la salida de una FLU respecto de los centroides de las reglas:

$$\frac{\partial y(x_1, x_2)}{\partial c_{ij}} = \frac{\mu_{1i}(x_1) \cdot \mu_{2j}(x_2)}{\sum_{i=1}^m \sum_{j=1}^m \mu_{1i}(x_1) \cdot \mu_{2j}(x_2)} \quad (10)$$

La ec. 9 permite también calcular la derivada parcial de la salida de una FLU respecto al grado de activación de cada función de pertenencia.

$$\frac{\partial y(x_1, x_2)}{\partial \mu_{1i}(x_1)} = \frac{\left(\sum_{j=1}^m \mu_{2j}(x_2) \cdot c_{ij} \right)}{\left(\sum_{i=1}^m \sum_{j=1}^m \mu_{1i}(x_1) \cdot \mu_{2j}(x_2) \right)} \cdot \frac{\left(\sum_{j=1}^m \mu_{2j}(x_2) \right) \cdot \left(\sum_{i=1}^m \sum_{j=1}^m \mu_{1i}(x_1) \cdot \mu_{2j}(x_2) \cdot c_{ij} \right)}{\left(\sum_{i=1}^m \sum_{j=1}^m \mu_{1i}(x_1) \cdot \mu_{2j}(x_2) \right)^2} \quad (11)$$

Por último hay que considerar la derivada parcial de la función de pertenencia respecto de sus parámetros de definición. Por ejemplo, para una función gaussiana

$$\mu(x, a, b) = \exp\left(-\left(\frac{x-a}{b}\right)^2\right) \quad (12)$$

las derivadas quedan

$$\begin{aligned} \frac{\partial \mu(x, a, b)}{\partial x} &= -\frac{2}{b} \cdot \left(\frac{x-a}{b}\right) \cdot \exp\left(-\left(\frac{x-a}{b}\right)^2\right) \\ \frac{\partial \mu(x, a, b)}{\partial a} &= \frac{2}{b} \cdot \left(\frac{x-a}{b}\right) \cdot \exp\left(-\left(\frac{x-a}{b}\right)^2\right) \\ \frac{\partial \mu(x, a, b)}{\partial b} &= \frac{2}{b} \cdot \left(\frac{x-a}{b}\right)^2 \cdot \exp\left(-\left(\frac{x-a}{b}\right)^2\right) \end{aligned} \quad (13)$$

Con todas estas ecuaciones se puede calcular el gradiente del error cuadrático medio, tal y como se indica en la ec. 7 y utilizarlo en el algoritmo de optimización paramétrica seleccionado. En el caso de que el comportamiento de las FLUs se basen en otro operador de conjunción, otro método de concreción u otro tipo de funciones de pertenencia habría que recalcular las ecuaciones 8 a 13.

4 LA HERRAMIENTA XFHL

Xfhl es una nueva herramienta integrada en el entorno Xfuzzy de desarrollo de sistemas difusos. La herramienta *Xfhl* implementa la metodología de inducción de sistemas difusos jerárquicos comentada en los apartados anteriores. El funcionamiento de la herramienta se basa en generar todas las estructuras jerárquicas de n entradas y una salida que se pueden construir en base a módulos de dos entradas

y una salida. La estructura seleccionada es la que presenta un menor error cuadrático medio tras un proceso de optimización paramétrica basado en un algoritmo de descenso por gradiente (RProp o Levenberg-Marquardt).

La generación del conjunto de estructuras jerárquicas para n variables de entrada se basa en el algoritmo presentado en la segunda sección. Para ello se utiliza un iterador basado en una pila de estructuras. Cada nivel de la pila corresponde a un número de entradas comenzado desde la estructura base con dos entradas. La cima de la pila contiene la estructura de n entradas que genera el iterador. En cada iteración se desapila la última estructura generada y se genera la siguiente estructura a partir de la estructura de $n-1$ entradas almacenada en la pila. Cuando se han generado todas las estructuras de n entradas correspondientes, se desapila la estructura de $n-1$ entradas y se genera una nueva estructura de $n-1$ entradas en función de la estructura de $n-2$ entradas almacenada en la pila, y así sucesivamente. Gracias a este iterador se evita tener que almacenar en memoria todas las estructuras jerárquicas a la vez.

Para evaluar cada estructura se realiza un proceso de optimización paramétrica sobre todos los parámetros de la estructura. Para ello se calcula el gradiente del error cuadrático medio tal y como se explica en la sección 3 y se aplica el algoritmo de descenso por gradiente elegido.

Puesto que la optimización y evaluación de una estructura se puede realizar de manera independiente del de las demás, el proceso de búsqueda es fácilmente paralelizable. La herramienta *Xfhl* ha sido programada para aprovechar las capacidades de paralelización de los sistemas actuales (arquitecturas multicore, multiprocesador y tecnologías de hyperthreading) de manera que el proceso de estudio de las diferentes estructuras jerárquicas se realiza en paralelo en varios hilos de ejecución.

La ventana principal de la herramienta se muestra en la Figura 3. En la parte superior de la ventana se encuentra el menú de configuración que permite seleccionar las dife-

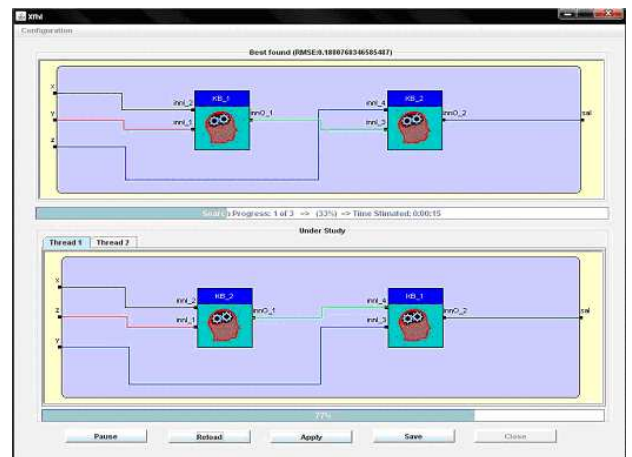


Figura 3: Ventana principal de la herramienta Xfhl

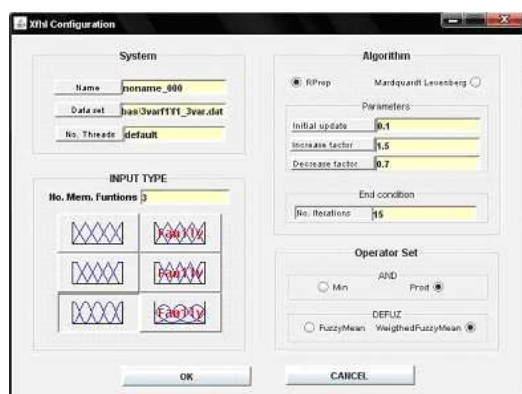


Figura 4: Ventana de configuración de la herramienta

rentes opciones de ejecución de la herramienta. Bajo la barra de menú se encuentra un área con el mejor sistema difuso encontrado hasta el momento y una barra de progreso que indica el porcentaje de estructuras estudiadas hasta el momento. Bajo esta barra se encuentra un panel en el que se muestra el estado de cada hilo de ejecución de la herramienta, que incluye para cada hilo la representación de la estructura en estudio y una barra de progreso del proceso de optimización. En la parte inferior de la ventana se encuentran los botones de control que permiten lanzar la ejecución, detenerla, almacenar el mejor resultado obtenido hasta el momento y cerrar la herramienta.

La configuración de la herramienta *Xfhl* se realiza por medio de la ventana mostrada en la Figura 4. Esta configuración requiere de la selección de los valores de los siguientes campos:

- Conjunto de datos de entrenamiento.
- Número de hilos concurrentes a ejecutar. El valor "Default" especifica el número de hilos igual al número de núcleos del sistema.
- Número de funciones de pertenencia de cada variable de entrada
- Estilo de las funciones de pertenencia de las variables de entrada. Pueden ser de los siguientes tipos: Free Triangles, Free Shouldered Triangles, Free Gaussians, Triangles Family, Shouldered-Triangular Family y B-Splines Family.
- Función asociada al operador de conjunción. Puede ser el mínimo o el producto.
- Algoritmo de concreción utilizado en los módulos. Puede ser la media difusa o la media difusa ponderada.
- El algoritmo de optimización paramétrica a usar para evaluar cada estructura. Se puede elegir entre RProp y Levenberg-Marquardt y se deben configurar sus parámetros de control y el número de iteraciones a realizar en el proceso de ajuste.

5 PRUEBAS

Para demostrar el funcionamiento de la herramienta *Xfhl* se ha considerado la siguiente función continua con 4 variables de entrada y estructura jerárquica separable:

$$f(w, x, y, z) = (w^2 + z^2) \cdot \cos\left(\frac{\pi}{3} \cdot (x - 2y)\right) \quad (14)$$

Para generar el conjunto de datos se ha considerado que las 4 variables están definidas en el intervalo $[-1.0, 1.0]$ y se han generado todas las instancias posibles barriendo los valores de las variables en intervalos de 0.25, lo que produce un conjunto de datos con un total de 6561 instancias.

El proceso de ejecución de la herramienta *Xfhl* se ha desarrollado considerando una división de las variables en tres funciones de pertenencia gaussianas y seleccionando el producto como operador de conjunción y la media difusa ponderada como método de concreción. El ajuste paramétrico se ha llevado a cabo por medio de 40 iteraciones del algoritmo Rprop. La ejecución se ha realizado con dos hilos sobre una máquina con procesador Intel Core2 Duo de 2.0 GHz, mostrando una duración aproximada de 3 minutos. De entre las 15 posibles descomposiciones la mejor estructura encontrada, con un RMSE = 2.9%, es la mostrada en la Figura 5.a, que corresponde a la estructura jerárquica de la función objetivo. Un ajuste paramétrico posterior de nuevo con RProp consigue una mejora de la aproximación hasta un RMSE=1.4%.

La Figura 5 muestra la representación gráfica del comportamiento de los diferentes módulos de la estructura jerárquica obtenida. Como se puede observar en la Figura 5.b, el comportamiento del módulo FLU-1 se aproxima al de la función coseno sobre una relación lineal de las variables x e y . (En realidad la función está desplazada ya que el rango de salida se encuentra entre 0 y 1). Un estudio más detallado permite incluso deducir esta relación. Con respecto al comportamiento del módulo FLU-2, la Figura 5.c permite también deducir que la relación entre las variables w y z corresponde a una parábola de revolución, es decir, a la función w^2+z^2 multiplicada por un factor negativo. En el caso del módulo FLU-3 el comportamiento es algo más difícil de interpretar ya que este módulo se encarga no sólo de calcular el producto entre las dos partes de la función, sino también de compensar el desplazamiento y los factores de escala de los módulos anteriores.

Agradecimientos

Este trabajo ha sido financiado parcialmente por los Proyectos de Investigación TEC2008-04920/TEC y TIN2008-06681-C06-06 de la CICYT y los Proyectos de Excelencia P07-TIC-03179 y P08-TIC-03674 de la Junta de Andalucía.

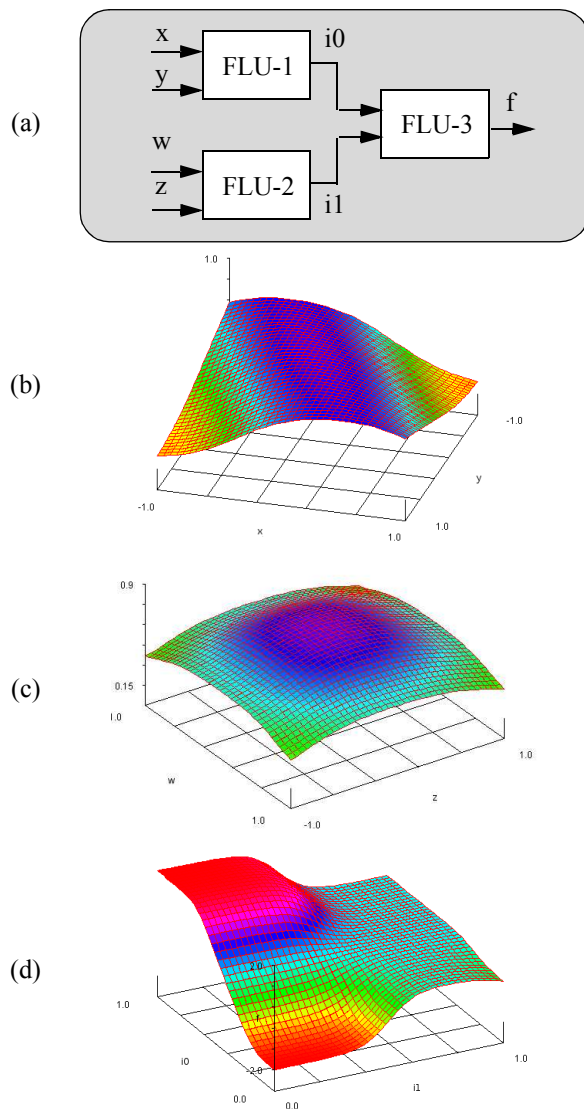


Figura 5: Resultado de *Xfhl* sobre la función de prueba.
(a) estructura jerárquica encontrada; (b,c,d) comportamiento de los módulos FLU-1, FLU-2 y FLU-3.

Referencias

- [1] S. Aja-Fernández, C. Arbeloa-López. Matrix modeling of hierarchical fuzzy systems, *IEEE Transactions on Fuzzy Systems*, Vol. 16, N. 3, Pág. 585-599, 2008.
- [2] R. Bellman, *Adaptive Control Processes*, Princeton University Press, 1966.
- [3] A.D. Benítez, J. casillas. Aprendizaje Evolutivo de Sistemas Difusos Jerárquicos en Serie, *Actas del VI Congreso Español sobre Metaheurística, Algoritmos Evolutivos y Bioinspirados (MAEB'09)*, Pág. 255-262, 2009.
- [4] J.L. Castro. Fuzzy logic controllers are universal approximators, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 25, N. 4, Pág. 629-635, 1995.
- [5] Y. Chen, B. Yang, A. Abraham, L. Peng, Automatic Design of Hierarchical Takagi-Sugeno Type Fuzzy Systems Using Evolutionary Algorithms, *IEEE Transactions on Fuzzy Systems*, Vol. 15; N. 3, Pág. 385-397, 2007.
- [6] O. Cordón, F. Herrera, I. Zwir. A hierarchical knowledge-based environment for linguistic modeling: models and iterative methodology, *Fuzzy Sets and Systems*, Vol. 138, Pág 307-341, 2003.
- [7] R. Fletcher. *Practical Methods of Optimization*, John Wiley & Sons, Ltd., 1986
- [8] M.G. Joo, J.S. Lee, Universal approximation by hierarchical fuzzy system with constraints on the fuzzy rule, *Fuzzy Sets and Systems*, Vol. 130, N. 2, Pág. 175-188, 2002.
- [9] M.L. Lee, H.Y. Chung, F.M. Yu. Modeling of hierarchical fuzzy systems, *Fuzzy Sets and Systems*, Vol. 138, Pág. 343-361, 2003.
- [10] F.J. Moreno-Velo, I. Baturone, A. Barriga, S. Sanchez-Solano. Automatic tuning of complex fuzzy systems with Xfuzzy, *Fuzzy Sets and Systems*, Vol. 158, Pág. 2026-2038, 2007.
- [11] G.V.S. Raju, J. Zhou, R.A. Kisner. Hierarchical fuzzy control, *International Journal of Control*, Vol. 54, Pág. 1201-1216, 1991.
- [12] M. Riedmiller, H. Braun. RPROP: A fast and robust backpropagation learning strategy, *Proc. 4th Australian Conference on Neural Networks*, Pág 169-72, 1993.
- [13] V. Torra. A Review of the Construction of hierarchical Fuzzy Systems, *International Journal of Intelligent Systems*, Vol. 17, Pág. 531-543, 2002.
- [14] D. Wang, X.J. Zeng, J. Keane. Learning for Hierarchical Fuzzy Systems Based on the Gradient-Descent Method, *Proc. 2006 IEEE International Conference on Fuzzy Systems*, Pág. 92-99, 2006.
- [15] L.X. Wang. Universal approximation by hierarchical fuzzy systems, *Fuzzy Sets and Systems*, Vol. 93, Pág. 223-230, 1998.
- [16] L.X. Wang. Analysis and Design of Hierarchical Fuzzy Systems, *IEEE Transactions on Fuzzy Systems*, Vol. 7, N. 5, 1999.
- [17] X.J. Zeng, J.A. Keane. Approximation Capabilities of Hierarchical Fuzzy Systems, *IEEE Transactions on Fuzzy Systems*, Vol. 13, N. 5, Pág. 659-672, 2005.

MODELADO Y CONTROL NEURO-DIFUSO DE PROCESOS DE TALADRADO DE ALTO RENDIMIENTO

Agustín Gajate¹ Pastora Vega² Rodolfo Haber¹

¹ Instituto de Automática Industrial
Consejo Superior de Investigaciones Científicas
{agajate,rhaber}@iai.csic.es

² Departamento de Informática y Automática
Universidad de Salamanca
pvega@usal.es

Resumen

Este trabajo presenta el diseño y la implementación de un sistema neuro-difuso para el modelado y control del proceso de taladrado de alto rendimiento. Primeramente, un sistema neuro-difuso se entrena para aprender la dinámica directa del proceso a través de datos entrada/salida. A su vez, otro sistema neuro-difuso se entrena para modelar la dinámica inversa. Posteriormente, estos modelos generados (directo e inverso) se introducen en un esquema de control por modelo interno para controlar la fuerza de corte del proceso a través de la modificación de la velocidad de avance de la herramienta. El sistema neuro-difuso considerado en este estudio es el conocido como *Adaptive Network based Fuzzy Inference System* (ANFIS). Los resultados obtenidos son positivos tanto en la simulación como en la aplicación al control de la fuerza de corte.

Palabras Clave: Sistemas neuro-difusos, Modelado, Control, Control por modelo interno, Taladrado de alto rendimiento.

1 INTRODUCCIÓN

La obtención de un modelo matemático que sea capaz de representar un sistema real es un tema fundamental en multitud de áreas relacionadas con la ingeniería y la ciencia. Los modelos pueden ser utilizados para realizar simulaciones, para analizar el comportamiento del sistema, para entender mejor los mecanismos subyacentes del sistema, para diseñar nuevos procesos y para controlar los propios sistemas. Ciertamente no se trata de una tarea

sencilla y tiene una importancia vital para el cumplimiento de objetivos posteriores.

En el campo de la ingeniería de control, el modelado y la identificación son pasos muy importantes en el diseño de sistemas de control, sistemas de supervisión y sistemas de detección de fallos. Para satisfacer los estrictos requerimientos de calidad, los sistemas de control deben garantizar un buen funcionamiento sobre todo el amplio rango de condiciones de operación. Bajo estas condiciones, frecuentemente el modelado de procesos constituye el principal cuello de botella en la aplicación de técnicas avanzadas de control basadas en modelo [7].

Muchos sistemas no son susceptibles de ser modelados a través de técnicas tradicionales debido a la pérdida de precisión, a la no disponibilidad de un conocimiento técnico exacto acerca del sistema, al comportamiento fuertemente no lineal, al alto grado de incertidumbre, o a sus características variantes en el tiempo del proceso [1]. Existen ejemplos muy variados de estos sistemas en la industria de procesos, manufacturera, aeronáutica, en el campo de la bioingeniería, etc. En un primer análisis a cada uno de estos campos, se puede extraer que en todos se desarrollan procesos de mecanizado. Los procesos de mecanizado tienen una gran importancia dentro de los sectores anteriormente citados. Estos procesos involucran maquinaria y materiales muy costosos así como operadores expertos. De ahí que cualquier modelo que sea capaz de describir de forma eficiente los procesos físicos que tienen lugar en ellos, es esencial para la mejora y optimización de estos sistemas. La fuerte competencia a nivel internacional en el sector del mecanizado obliga a desarrollar e implantar sistemas de control que mejoren los procesos productivos en todas sus facetas [6].

Los procesos de mecanizado se enmarcan en la serie de procesos de difícil modelado a través de técnicas clásicas. La razón fundamental es que son procesos extremadamente complejos, fuertemente no-lineales,

cargados de incertidumbre y con comportamientos imposibles de describir exactamente por herramientas de modelado convencionales. En este caso, las aproximaciones basadas en técnicas de Inteligencia Artificial (IA) constituyen una alternativa útil y viable para tratar con estos problemas [10]. El uso masivo de las técnicas de IA en el modelado de procesos, y sobre todo en el control, ha sido consecuencia de que, en buena medida, las técnicas clásicas no han producido, en algunos casos, los resultados deseados y se han mostrado efectivas sólo en las condiciones teóricas de diseño. Además, las prestaciones de los reguladores clásicos se deterioran considerablemente en presencia de incertidumbre y de no linealidades.

Por estas razones, el presente trabajo trata de modelar, a través de técnicas de IA, la dinámica de una de las operaciones de mecanizado más frecuente, más compleja y que ha recibido menos atención: el taladrado de alto rendimiento. Una vez se haya modelado la dinámica del proceso, se procederá a controlar el proceso utilizando estrategias de control que incluyan controladores neuro-difusos con el objetivo de mejorar la eficiencia del proceso alcanzando mayores tasa de arranque de material y considerando la vida útil de la herramienta de corte.

2 EL PROCESO DE TALADRADO DE ALTO RENDIMIENTO

El taladrado representa uno de las operaciones de mecanizado más comunes en la industria manufacturera. Se llama taladrar a la operación de mecanizado que tiene por objeto producir agujeros cilíndricos en una pieza cualquiera, utilizando como herramienta una broca. La operación de taladrar se puede hacer con un taladro portátil, con una máquina taladradora o en un centro de mecanizado equipado con un Control Numérico Computerizado (CNC). Esta última opción es la más contemporánea y, además, es la opción utilizada en este trabajo.

Al igual que en los procesos de mecanizado en general, en el proceso de taladrado, el término “alto rendimiento” se refiere a operaciones de taladrado que utilizan parámetros de corte muy superiores a los utilizados convencionalmente. Sin embargo, no existe una velocidad determinada como umbral, sino que consiste en una determinada combinación de pieza, máquina herramienta, herramienta y velocidades óptimas de corte.

Pese a su importancia y sus múltiples estudios, el proceso de taladrado no se ha beneficiado suficientemente de las posibilidades que ofrecen el control y la automatización de procesos. Los desarrollos en los procesos de taladrado pueden tener repercusiones muy positivas en la calidad del acabado, en la productividad y en el coste.

Examinando las operaciones de taladrado se observa que a medida que la profundidad del taladro aumenta, la viruta ya no se desaloja tan fácilmente del agujero y comienza a causar un incremento en la fricción entre la broca y la pieza. Como resultado, la fuerza de corte y la temperatura de corte aumentan continuamente, y si no existe una acción apropiada, se producen vibraciones en la herramienta, desgaste excesivo, perjudiciales roturas de herramienta y quizás fallos catastróficos en el cabezal. Todas estas consecuencias son evidentemente muy indeseadas desde cualquier punto de vista productivo. Por ello, el control en tiempo real de la fuerza de corte es una solución adecuada para evitar el incremento indeseado en la fricción pieza-herramienta. El incremento en la tasa de arranque de material también guarda relación directa con el mantenimiento de la fuerza constante en toda la operación. Maximizar la fuerza de corte en la operación produce un incremento en la tasa de arranque de material pero acelera el desgaste de la herramienta de corte, su posible rotura, y se pueden producir daños en la calidad de la pieza entre otros. De este modo, no solo es importante mantener la fuerza de corte constante, también hay que obtener un buen comportamiento dinámico a lazo cerrado (sin oscilaciones ni sobrepaso). Así se alarga la vida útil de la herramienta de corte.

Sin embargo, antes de llegar a realizar cualquier tipo de control sobre la fuerza de corte, es necesario obtener un modelo que relacione esta variable con el resto de fenómenos físicos que ocurren durante la operación. Numerosos autores han obtenido modelos lineales del proceso de taladrado, fundamentalmente a través de técnicas clásicas, considerándolo como un sistema de una entrada y una salida [5,9]. Generalmente como entrada se considera la velocidad de avance (f), y como salida, la fuerza media de corte (F). El proceso de taladrado está considerado un proceso no lineal. Sin embargo, la representación mediante un modelo lineal proporciona una descripción aproximada del proceso, que ayuda a investigar y analizar las posibles vías de mejoras mediante sistemas de control. En este trabajo se ha utilizado el modelo propuesto por Del Toro *et al.* para la realización de simulaciones [3]. Tanto este modelo lineal como los demás modelos lineales encontrados en la literatura suponen una aproximación al proceso de taladrado. Es necesario aclarar que los modelos lineales obtenidos son aproximaciones imprecisas dado que existen no linealidades y comportamientos variantes en el tiempo del proceso (desgaste de herramienta, etc.) que no reflejan esta serie de modelos. También es necesario mencionar que se han realizado controles en la literatura en base a alguna de estas aproximaciones lineales, pero precisamente no han aportado los resultados esperados (dado que el proceso es no-lineal). Este es el motivo fundamental de emplear técnicas neuro-difusas para el modelado del proceso de taladrado de alto rendimiento.

2.1. MODELADO A TRAVÉS DE ANFIS

Debido a las múltiples ventajas que aporta el uso de las técnicas neuro-difusas y a que su empleo tiene novedad y utilidad en el campo de investigación objeto de este estudio, se decide modelar la dinámica del proceso a través del primer sistema neuro-difuso conocido y el más establecido, ANFIS. Además de ser uno de los trabajos pioneros, resulta uno de los más sencillos computacionalmente hablando [4].

El principio de ANFIS se basa en la extracción de reglas difusas en cada nivel de una red neuronal. Una vez obtenidas las reglas, éstas deben proporcionar la información necesaria del comportamiento global del proceso. ANFIS implementa el modelo de Takagi-Sugeno para la estructura de las reglas *If-Then* del sistema difuso. La arquitectura de ANFIS dispone de cinco capas, tal y como se muestra en la Fig. 1. Los nodos representados con cuadrados son nodos cuyos parámetros son ajustables, mientras que los nodos representados por círculos son nodos fijos.

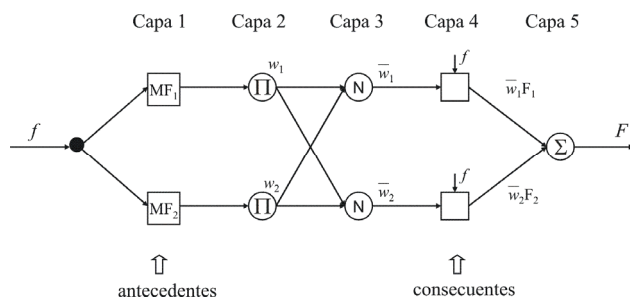


Figura 1: Arquitectura del sistema ANFIS.

Al igual que en los modelos clásicos estudiados, aquí se decide también reducir el sistema a una entrada y una salida, siendo la velocidad de avance de la herramienta de corte (f) la variable de entrada y la fuerza media de corte (F) la variable de salida. A esta representación del sistema se la conoce como modelo directo. Una vez definida la estructura del modelo, el sistema neuro-difuso ANFIS se entrena para que aprenda la dinámica del proceso por medio de datos entrada-salida.

En este trabajo se han utilizado datos reales de fuerza de corte y velocidad de avance obtenidos experimentalmente por los autores, en la máquina en la que posteriormente se efectuará el control del proceso. Los datos se obtuvieron de pruebas realizadas sobre probetas de fundición nodular GGG40, material muy utilizado en la industria aeroespacial. Las condiciones nominales de operación fueron velocidad de giro de 870 rpm, velocidad de avance inicial de 100 mm/min y diámetro de la herramienta de 10 mm. Los datos se tomaron de una operación con profundidad de corte de 15 mm.

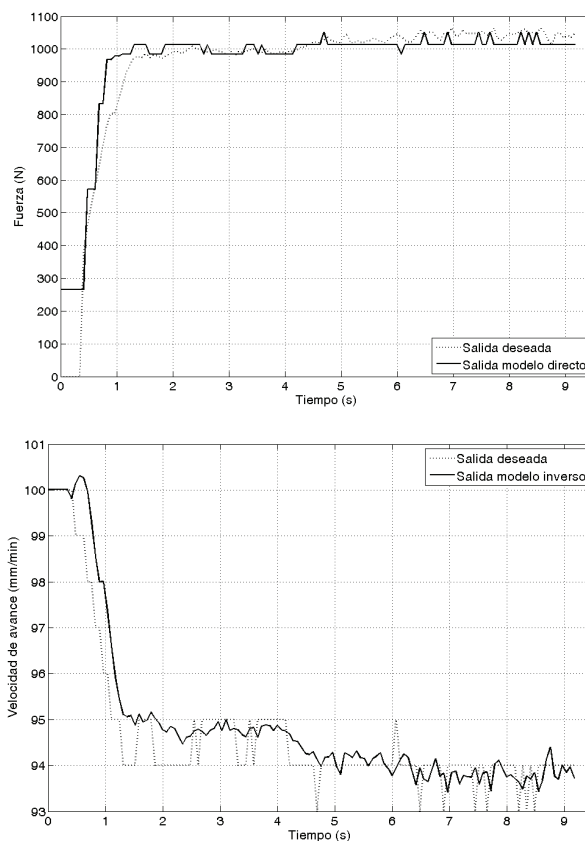


Figura 2: a) Respuesta del modelo neuro-difuso directo y b) respuesta del modelo neuro-difuso inverso.

En la fase de obtención de los modelos son modificables ciertos parámetros del sistema ANFIS tales como el número de funciones de pertenencia en el emborronado, la clase o tipo de dichas funciones, así como el orden de las reglas Takagi-Sugeno de desemborronado. Asimismo, se puede mejorar la precisión del modelo cambiando los parámetros del proceso de aprendizaje (método, número de iteraciones, tamaño del paso, etc.). La elección de las variables correctas y los parámetros óptimos se ha realizado a partir de un compromiso entre el criterio de error de la raíz cuadrada del error cuadrático medio (1) y la respuesta dinámica del modelo (sin oscilaciones ni sobrepasos).

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (F_i - \bar{F}_i)^2} \quad (1)$$

En ocasiones resulta también conveniente modelar la dinámica inversa del modelo obteniéndose el denominado modelo inverso. Esta técnica es especialmente útil a la hora de diseñar controladores del propio proceso, ya que son los encargados de modificar, en este caso, la velocidad de corte de la herramienta. Por ello nos interesa también conocer qué velocidad de avance se obtiene

cuando existe una determinada fuerza de corte. Para obtener el modelo inverso, en vez de invertir el modelo obtenido anteriormente (ya sea por técnicas clásicas o neuro-difusas), realizamos otro entrenamiento del sistema pero esta vez con la fuerza media como entrada y la velocidad de avance como salida. Nuevamente utilizaremos un sistema neuro-difuso ANFIS para esta labor. En este caso, también se analizó la modificación de los parámetros del sistema para modelar la dinámica inversa, resultando finalmente como parámetros óptimos los mismos parámetros utilizados para el caso del modelo directo.

Los parámetros utilizados en el sistema ANFIS se muestran en la Tabla 1. Asimismo, la Fig. 2 muestra la respuesta (salida) de los modelos neuro-difusos creados ante la entrada de un conjunto de datos de validación (datos experimentales en los cuales se conoce la salida que ha proporcionado el proceso).

Tabla 1: Resumen de los parámetros.

ANFIS	PARÁMETROS
Sistema	SISO
Tipo de función de pertenencia	Gaussiana
Nº de funciones de pertenencia	2
Tipo de Inferencia	Takagi-Sugeno
Número de reglas	2
Iteraciones	100
Ratio de aprendizaje	0.01
Algoritmo de entrenamiento	Retropropagación + Mín. cuadrados
Conjunto de datos de entrenamiento	133 datos
Conjunto de datos de validación	128 datos

3 SISTEMA DE CONTROL NEURO-DIFUSO

La aplicación al control de las técnicas de inteligencia artificial, en general, y de los sistemas neuro-difusos en particular, pasa necesariamente por la utilización de un método de diseño que satisfaga los requisitos del mismo. Varios han sido los esquemas de control que han utilizado modelos o controladores neuro-difusos en su estructura.

El esquema de control por modelo interno es uno de los paradigmas de la teoría de control que incluye la utilización explícita de modelos, y que surge a principios de los años ochenta de la mano de Morari y Zafiriou [8]. Este paradigma se concreta en un esquema de control en lazo cerrado en el que intervienen tanto un modelo directo

(G_M) del proceso a controlar (G_P) situado en paralelo con éste, así como un modelo inverso (G_M') del mismo. Desde un punto de vista general, podríamos decir que el modelo directo del proceso actúa como un predictor, mientras que el modelo inverso haría las veces de controlador. Las perturbaciones externas al proceso vienen representadas por d en la Fig. 3. F_R representaría, en este caso particular, la fuerza de corte deseada. El filtro (G_F) se incluye en el sistema de control con el objetivo de reducir la ganancia de alta frecuencia y mejorar la robustez del sistema. También sirve para suavizar los cambios rápidos y bruscos en las señales, mejorando la respuesta del controlador. El bloque L representa el retardo inherente a la red de comunicación encargada de enviar la señal de control al proceso.

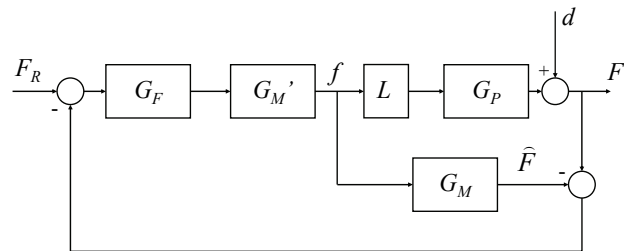


Figura 3: Esquema de control por modelo interno.

El control por modelo interno (CMI) ha ganado gran aceptación desde sus orígenes debido a su capacidad para hacer frente a las perturbaciones y su comportamiento robusto. Además, este paradigma constituye un enfoque bien establecido para el diseño de controladores. El diseño del controlador es sencillo y su sintonización simple, además, se encuentra bien fundamentado a través de diversos trabajos teóricos para sistemas lineales. En la práctica, sin embargo, la mayoría de los sistemas presentan algún tipo de no linealidad, razón por la que surge la necesidad de extender el CMI a sistemas no lineales. Uno de los mayores problemas aparece debido a que, en general, la inversión de modelos no lineales no es una tarea fácil, y pueden no existir soluciones analíticas, de modo que las soluciones tienen que calcularse numéricamente. Otro problema es que la inversión del modelo del proceso puede conducir a controladores inestables [2].

Para solventar este tipo de dificultades se combina este tipo de esquema de control con técnicas de modelado no lineales. Algunas técnicas ampliamente utilizadas para obtener representaciones de sistemas no lineales son las redes neuronales y los sistemas difusos. Varios son los trabajos que relacionan el control por modelo interno con estas técnicas. De ahí que este tipo de paradigma pueda ser combinado con modelos neuro-difusos (como modelos directo e inverso) tal y como se propone y aplica en este trabajo.

4 RESULTADOS EXPERIMENTALES

Una vez creados los modelos neuro-difusos directo e inverso, y habiéndose diseñado la estrategia de control utilizando el paradigma del control por modelo interno, se estudió el comportamiento del sistema de control mediante simulación. El principal objetivo es obtener una respuesta transitoria sin sobrepaso o máximo pico (Ovt. = 0%), con un tiempo de establecimiento inferior a 2 segundos utilizando como condiciones iniciales las proporcionadas por el manual del fabricante de la herramienta para la combinación dada de material y broca.

En la fase de simulación se consideró un retardo entre la acción de control y el proceso, tal y como sucede en el proceso de taladrado real, dado que la conexión entre el proceso y el sistema de control se hace a través de una red Profibus (por restricciones del fabricante de CNC). Asimismo, se incluyó una perturbación o ruido aditivo a la salida con el fin de emular la influencia de las perturbaciones sobre el sistema de control.

Después de verificar el buen comportamiento del sistema de control en simulación se deciden realizar ensayos en un entorno industrial real. Los ensayos se realizaron en una fresadora Kondia HS1000 equipada con un CNC abierto Sinumerik 840D. En los experimentos se utilizó una broca de diámetro 10 mm Sandvik R840-1000-30-A0A de metal duro integral con recubrimiento de TiN/TiAlN. El control se ha realizado a través de un PC que se encuentra conectado al CNC a través de una red Profibus (Fig. 4).

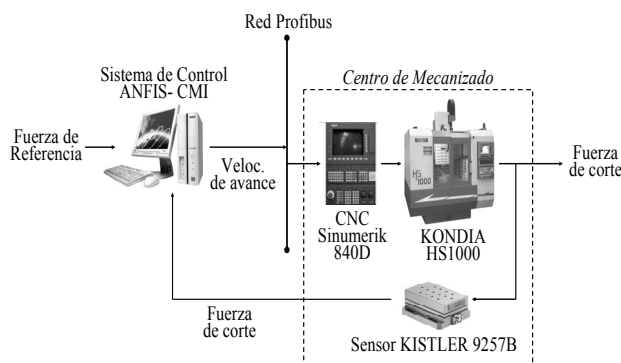


Figura 4: Arquitectura del control ANFIS-CMI en un proceso real.

Los modelos y el esquema de control fueron implementados en la *toolbox* RTWT de Simulink. Para la lectura y escritura de datos a través del bus se desarrolló una aplicación en lenguaje C. En caso de fallo o pérdida de información de la red, el sistema tiene programado un mecanismo de seguridad en el CNC que mantiene la velocidad de avance constante sobre el proceso.

Para comprobar el funcionamiento del sistema de control se realizaron varios ensayos con las condiciones nominales de velocidad de avance $f = 100$ mm/min, velocidad de giro $n = 870$ rpm y 14 mm en la profundidad de corte recomendadas para el GGG40, material con el que se hicieron las pruebas. La fuerza de corte se mide a través de una plataforma dinamométrica Kistler 9257B. Los parámetros del CNC y la acción de control (velocidad de avance) se transmiten a través de la red.

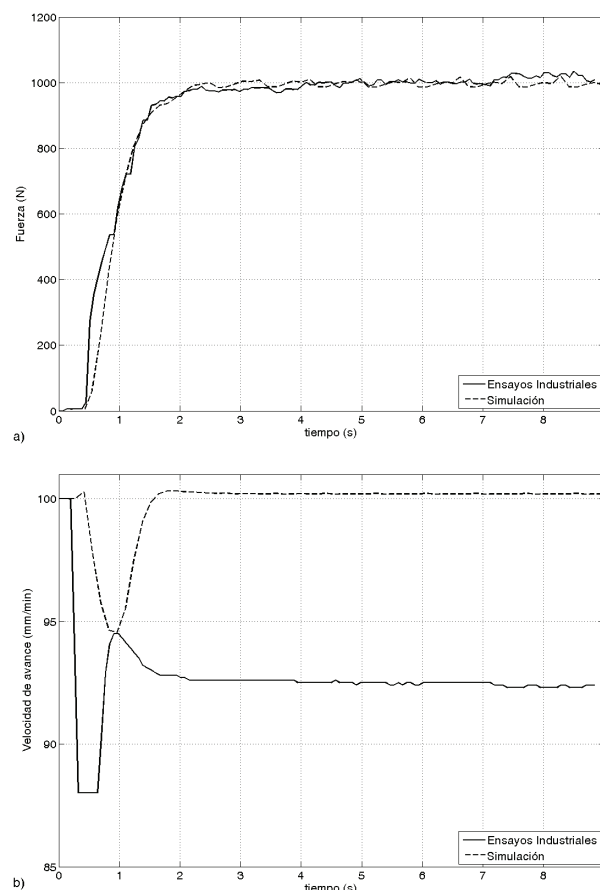


Figura 5: a) Comportamiento de la fuerza de corte y b) Acción de control en el taladrado del material GGG40 (Simulación y Ensayos industriales reales).

Los resultados de los ensayos experimentales en el taladrado del material GGG40 (dureza de 233HB) se muestran en la Fig. 5. Se puede observar como a pesar de la exigente condición inicial y del retardo, el sistema de control neuro-difuso garantiza una respuesta dinámica rápida (tiempo de establecimiento de 2 segundos) y sin sobrepaso. Con el sistema de control se consigue aumentar la tasa de arranque de material. Además, la calidad en la respuesta transitoria y la no existencia de sobrepaso y oscilaciones en la respuesta contribuyen desde el punto de vista industrial a un mejor aprovechamiento de la vida útil de la herramienta.

Desde el punto de vista práctico la existencia de una condición inicial en la velocidad de avance (100 mm/min para el acero GGG40), añade a los requisitos a cumplir en la respuesta transitoria y en el tiempo de establecimiento una mayor dificultad.

Tabla 2: Cifras de mérito del sistema de control.

CONTROL	ITAE	ITSE	Ovt. (%)
ANFIS-CMI (Ensayo Industrial)	0.90	0.20	0.00
ANFIS-CMI (Simulación)	0.54	0.96	0.00

4 CONCLUSIONES

Este trabajo propone una metodología para el diseño y aplicación de un sistema neuro-difuso de control al proceso de taladrado utilizando el paradigma de control por modelo interno. Mediante la utilización de dos sistemas neuro-difusos se obtuvieron los modelos directo e inverso del proceso para incorporarlos después al mencionado esquema de control. Una vez se concluyó la etapa de diseño, el sistema neuro-difuso de control se simuló y posteriormente se aplicó a la regulación de la fuerza de corte en un proceso de taladrado de alto rendimiento.

Desde el punto de vista científico, se ha corroborado la viabilidad de principio de control por modelo interno y la capacidad de un sistema neuro-difuso para capturar la dinámica directa e inversa de un proceso complejo, no lineal, plagado de incertidumbres y variable en el tiempo. Desde el punto de vista técnico, el sistema ha permitido incrementar en la tasa de arranque de material y una disminución en el tiempo de operación. Igualmente, la calidad de la respuesta transitoria sin máximo pico o sobrepaso permitió un aprovechamiento efectivo de la vida útil de la herramienta de corte, evitando el desgaste prematuro o acelerado de la broca, así como el fallo catastrófico de la misma. Del mismo modo, esta buena respuesta dinámica contribuye a un buen acabado de la pieza y a proteger los mecanismos de la propia máquina herramienta.

Las principales ventajas del método aplicado es que no se requiere un modelo matemático exacto del proceso para el diseño del sistema de control. Los datos experimentales de entrada/salida son usados para obtener los modelos neuro-difusos directo e inverso. Además, se trata de un procedimiento simple y computacionalmente eficiente que puede ser implementado en aplicaciones de tiempo real.

El desarrollo de nuevos métodos de inferencia y la incorporación de más variables relacionadas con el proceso serán objeto de trabajos futuros.

Agradecimientos

Este trabajo ha sido realizado en el marco del Proyecto COGNETCON (DPI 2008-01978). Los autores desean agradecer a F. Martínez su ayuda en realización de los experimentos industriales.

Referencias

- [1] R. Babuska y H. B. Verbruggen. An overview of fuzzy modeling for control. *Control Engineering Practice*, vol. 4, Pág. 1593-1606, 1996.
- [2] R. Carotenuto. Iterative system inversion technique. *International Journal of Adaptive Control and Signal Processing*, vol. 15, Pág. 85-91, 2001.
- [3] R. Del Toro, M. Schmittiel, R. Haber-Guerra, y R. Haber-Haber. System identification of the high performance drilling process for network-based control. In *Proc. of ASME IDETC 2007*, Las Vegas, NV, USA, 2007.
- [4] J.-S. R. Jang. ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, Pág. 665-685, 1993.
- [5] J. B. Kim, S. J. Lee, y Y. P. Park. Stable and efficient drilling process by active control of the thrust force. *Mechanical Systems and Signal Processing*, vol. 8, Pág. 585-595, 1994.
- [6] S. Y. Liang, R. L. Hecker, y R. G. Landers. Machining process monitoring and control: The state-of-the-art. *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 126, Pág. 297-310, 2004.
- [7] L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, Upper Saddle River, NJ, 1999.
- [8] M. Morari y E. Zafiriou. *Robust Process Control*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [9] Y. T. Oh, G. D. Kim, y C. N. Chu. Design of a Drilling Torque Controller for a Machining Center. *International Journal of Advanced Manufacturing Technology*, vol. 22, Pág. 329-335, 2003.
- [10] Y. Sheng y M. Tomizuka. Intelligent modeling of thrust force in drilling process. *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 128, Pág. 846-855, 2006.

HIBRIDACIÓN DE PARADIGMAS DE CLASIFICACIÓN EN EL CONTEXTO DEL RECONOCIMIENTO AUTOMÁTICO DE ACTIVIDADES FÍSICAS COTIDIANAS

Oresti Baños, Héctor Pomares, Ignacio Rojas

Departamento de Arquitectura y Tecnología de Computadores
Universidad de Granada, España
oresti@correo.ugr.es; hector@ugr.es; irojas@ugr.es

Resumen

Son de actualidad los trabajos dirigidos al reconocimiento de movimientos, gestos o ejercicios a partir del análisis de señales obtenidas mediante la monitorización de actividades físicas comunes. La gran mayoría de los estudios realizados hasta la fecha ofrecen buenos resultados en el ámbito supervisado, siendo más discretos para el caso no supervisado o naturalístico. La eficiencia de los sistemas de discriminación es extremadamente dependiente de la forma en que se implementan cada una de las etapas del proceso de reconocimiento (análisis de la información, extracción de características, etc.), con especial trascendencia para la fase de clasificación. En este trabajo se presenta una posible alternativa basada en la hibridación ponderada de máquinas de soporte vectorial y árboles de decisión que permita mejorar los ratios de acierto obtenidos de forma independiente, potenciando su rendimiento a través de la optimización de las metodologías empleadas en las etapas precedentes.

Palabras Clave: Reconocimiento de Actividades, Hibridación, SVM, Árboles de Decisión.

1 INTRODUCCIÓN

El reconocimiento de actividades físicas cotidianas es una compleja tarea de rigurosa actualidad que está siendo aplicada en muy diversos contextos [5, 12, 16]. La dificultad estriba fundamentalmente en hacer al sistema independiente de quién realiza el ejercicio y de cómo lo hace en particular (robustez). Elementos de distorsión asociados al sistema de monitorización y procesado, así como la imprevisibilidad en la ejecución de las tareas

incrementan la complejidad de éste. Campos como la teleasistencia, la telemedicina, la robótica, la industria del videojuego y un largo etcétera que completan una lista casi inagotable, están apostando en la actualidad por esta línea de investigación, siendo relevantes los avances y resultados obtenidos hasta la fecha [3, 6, 7, 16].

La mayoría de los estudios realizados trabajan sobre datos obtenidos en sesiones de laboratorio supervisadas. Algunos como [2, 8, 13, 17] ofrecen resultados bastante buenos en esta dirección, siendo más discretos en el marco seminaturalístico o semisupervisado. Ver que no se habla de modelos de monitorización naturalísticos, ya que éstos son más cercanos a un concepto idealista cuyas bases son la ausencia de supervisión por parte del investigador y el desconocimiento absoluto del sujeto acerca de su participación¹ en el proceso de monitorización. En la práctica se trabaja con modelos en los que se invita al sujeto a realizar una actividad concreta (por ejemplo, “mirar el correo electrónico usando un PC”), obteniendo el investigador la información deseada (en el ejemplo anterior, podría referirse a la actividad “permanecer sentado”) y manteniéndose ésta enmascarada para el usuario (su atención estaría centrada en la interacción con el ordenador, no en el hecho de que ha sido necesario sentarse para utilizarlo). En cualquier caso, frente al modelo supervisado aparece esta vertiente, de mayor interés por tratarse de una aproximación más cercana al comportamiento natural con el que solemos desempeñar las actividades. Lo cierto es que si ya de por sí el reconocimiento sobre datos obtenidos en sesiones de laboratorio es manifiestamente complejo, las connotaciones y particularidades asociadas a la monitorización no supervisada (las cuales tienen su raíz en la aleatoriedad y universalidad de los movimientos realizados para ejecutar una actividad concreta)

¹ El conocimiento del ejercicio que está siendo monitorizado o las indicaciones/correcciones por parte del supervisor influyen en gran medida sobre la forma en que el sujeto realiza los movimientos, alterando el carácter natural de éstos. Los problemas de reconocimiento aparecen cuando el sistema entrenado con estos datos es probado sobre actividades naturales cotidianas.

incrementan de forma significativa la dificultad de este proceso. Ello determina la necesidad de un enfoque distinto, especialmente cuando se trabaja en contextos relacionados con redes o entornos multisensados que requieren de un alto grado de confianza, el cual traslade la capacidad de extracción de conocimiento clásica basada en sistemas monolíticos (un solo clasificador [1, 10, 14]) a otros constituidos por la combinación de los anteriores [6, 9]. En este trabajo se presenta una posible alternativa de clasificación basada en la hibridación de dos paradigmas conceptualmente distintos que procura aprovechar los beneficios ofrecidos por cada uno de los modelos constituyentes, buscando un modelo conjunto (*ensemble model*) con un alto grado de robustez.

De este modo el artículo se estructura en un primer epígrafe en el que se resume la metodología seguida para llevar a cabo el proceso de reconocimiento. A continuación se detalla el modelo de clasificación híbrido basado en máquinas de soporte vectorial (SVM) y árboles de decisión ideado. Finalmente se plantean posibles alternativas en términos de trabajo futuro.

2 RECONOCIMIENTO DE ACTIVIDADES

En este trabajo se parte de un conjunto de señales² correspondientes a la aceleración registrada por una serie de sensores dispuestos estratégicamente en diferentes partes del cuerpo (ver figura 1), para un conjunto de actividades de práctica cotidiana. En cualquier caso, la filosofía que aquí se describe es generalizable a otro tipo de estudios basados en el reconocimiento de señales independientemente del origen de las mismas.



Figura 1: Disposición de los acelerómetros en el individuo.

La información de partida contiene una serie de artefactos y ruido inherente al proceso de adquisición de los datos.

² Base de datos facilitada en [2] por el Profesor Stephen Intille (Massachusetts Institute of Technology).

Se emplea para corregir dichas irregularidades un filtrado paso bajo y un filtrado paso alto, los cuales permiten eliminar respectivamente ruido electrónico de alta frecuencia y cambios bruscos en la aceleración. Se han valorado asimismo otras técnicas basadas en filtros de mediana, media y descomposición wavelets, si bien finalmente se ha optado por trasladar su uso al proceso de extracción de características (caracterización de las señales).

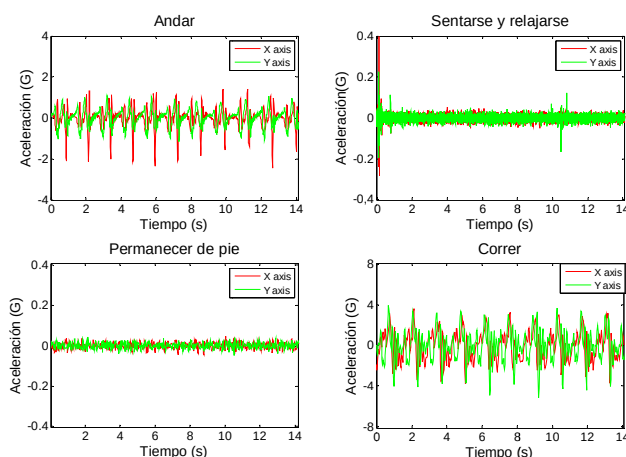


Figura 2: Señales correspondientes a 4 actividades físicas cotidianas (acelerómetro del tobillo).

A continuación definimos un conjunto de parámetros formado por un total de 861 características correspondientes a la combinación de una serie de operadores estadísticos como son la media, la curtosis, la moda, la varianza, etc., sobre magnitudes correspondientes a la transformación de la señal de aceleración a diferentes dominios, como son la densidad espectral de energía, la coherencia espectral o la descomposición wavelets, entre otras. Dichas características son calculadas sobre la señal completa, si bien fueron estudiadas otras alternativas originales de este trabajo basadas en métodos de *windowing* y procesos de selección del subsegmento más característico.

El hecho de definir un conjunto de características tan amplio se sustenta en la idea de trabajar sobre variables que han ofrecido buenos resultados en otros trabajos como son la media, la entropía o la energía [1, 2, 14], otras sugeridas en éste por el interés suscitado durante la fase de análisis de los datos y un subconjunto que en principio quizás no tengan un interés directo aparente en este trabajo, pero que son propuestas teniendo en cuenta que algunas de ellas pudieran ofrecer información ciertamente relevante y discriminante. La responsabilidad de decidir acerca de qué variables son las más interesantes recae sobre el proceso de selección de características, método de filtraje propio definido a través de una técnica de ranking basada en el análisis de la capacidad discriminativa de éstas. Brevemente, este método se fundamenta en el análisis comparativo de las diferentes

características a partir de la evaluación estadística de su potencial discriminante (ver figura 3), siendo entendido éste como la probabilidad de que muestras correspondientes a clases diferentes no sean confundidas (esto es, el error de predicción sea mínimo).

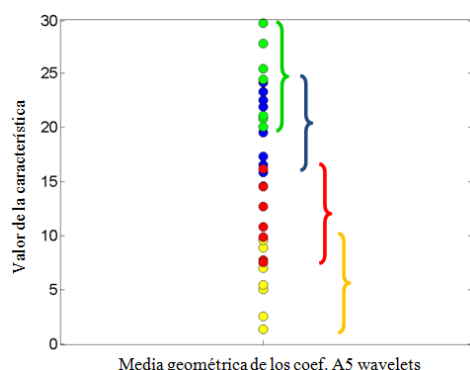


Figura 3: Valores extraídos a partir de la información obtenida a través del acelerómetro del tobillo. Cada círculo representa el valor (muestra) de la característica “media geométrica de los coeficientes A5 wavelets” para un sujeto específico. La capacidad discriminante de la característica analizada para una clase (actividad) concreta se obtiene a partir del número de muestras correspondientes a otras clases (actividades) que caen dentro del conjunto muestral de ésta.

Dado un subconjunto de características estimado como el más adecuado para la categoría de datos con los que se trabaja (las más destacables resultan ser la media geométrica de la señal amplitud, autocorrelación y algunos de los coeficientes wavelets obtenidos a través de una descomposición Daubechies de nivel 4), se evalúan diferentes metodologías de extracción de conocimiento y reconocimiento de patrones de forma individualizada, como paso previo a la introducción del modelo compuesto. Es cometido del siguiente apartado describir el modelo de clasificación ideado para dicha tarea.

3 CLASIFICADOR JERÁRQUICO HÍBRIDO PROPUESTO

El concepto de hibridación entraña una complejidad superior a la de la suma de los paradigmas que lo componen, ya que no se basa en una aplicación separada de cada modelo, sino que se requiere la interacción entre éstos. Es por ello que se suelen definir versiones adaptadas a las condiciones del problema en cuestión que tratan de minimizar los requerimientos exigidos, tal y como sucede en nuestro caso.

El sistema ideado trata de aprovechar las ventajas ofrecidas por los modelos de clasificación utilizados. Las máquinas de soporte vectorial se caracterizan por un bajo

número de parámetros a ajustar, la capacidad de generalización debido a la minimización del riesgo estructurado y la simplicidad en la solución (un número pequeño de vectores de entrada determina todo el modelo final). Los árboles de decisión facilitan la interpretación de la decisión adoptada, se adecuan a la existencia de datos incompletos, reducen el número de variables independientes y son rápidos de calcular. Todo ello sumado a la velocidad computacional que los caracteriza los ha determinado como métodos muy interesantes en este contexto [2, 6, 11, 14].

Conceptualmente el clasificador híbrido que se plantea puede entenderse como una máquina de estados en el que la probabilidad de paso de un estado a otro está definida a través de unos pesos cuantificados en el proceso de entrenamiento (parecido a los pesos que se definen en las entradas de los perceptrones de las redes neuronales). Dichos estados se materializan a través de los diferentes sistemas de clasificación que constituyen el modelo. Tal y como fue mencionado al principio, cuando se acomete un problema de hibridación se procura adaptarlo al problema en cuestión para el que se define. En nuestro caso partimos de la definición de clasificadores SVM y árboles de decisión que serán vinculados por las conexiones anteriormente definidas.

Si bien existen diferentes aproximaciones para la aplicación multiclase de SVM binarios, en este trabajo se ha apostado por una estrategia *uno contra todos* que simplifica la decisión a la selección entre una clase concreta o el resto, entendiendo el resto como una clase propiamente dicha. Hemos optado por un núcleo de base radial (RBF) que ofrece muy buenos resultados en la búsqueda del hiperplano óptimo tanto en modelos de datos de carácter lineal como para el caso no lineal [15]. Lo mencionado anteriormente se refleja en el ejemplo de la figura 4.

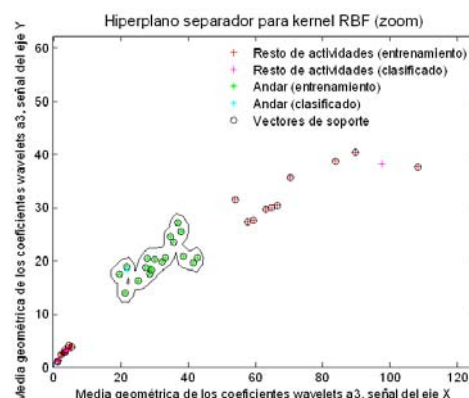


Figura 4: Hiperplano obtenido a través del método SVM-RBF. Clases correspondientes a las actividades mostradas en la figura 2.

Dado que se desea una decisión concreta acerca de la clase correspondiente al evento clasificado, se constituye

un modelo de decisión jerárquico que integre para el análisis de cada clase una serie de clasificadores basados en lo visto anteriormente. Una posible realización para 4 clases se muestra en la figura 5. El orden en la jerarquía viene dado por los ratios de acierto medidos para cada clasificador de forma independiente. Véase el ejemplo para la clase A (figura 5.b): se definen un total de k clasificadores SVM cuya influencia en la decisión final vendrá dada por el peso w_i . La obtención de dichos parámetros de ponderación se realiza tras un estudio individualizado de los ratios de acierto de cada uno de los clasificadores constituyentes, fijando sus valores en base a la comparación con el resto (por ejemplo, dados tres clasificadores C1, C2 y C3 con ratios de acierto 59%, 75% y 83% respectivamente, es claro que se ha de cumplir que $w_1 < w_2 < w_3$ con valores proporcionales a las eficiencias de clasificación). Los clasificadores son diferentes, en cuanto a las variables o características de entrada en que basan su decisión. Este modelo es ciertamente interesante para mejorar la robustez del sistema, algo bastante necesario al trabajar sobre datos seminaturalísticos.

Con respecto a los clasificadores basados en árboles de decisión [4], éstos permiten trabajar sobre modelos multiobjetivo o multiclase de modo que no es necesario un modelo jerárquico como el presentado en la figura 5 para la decisión. En cualquier caso aquí se apuesta por un clasificador compuesto por subelementos que constituyen modelos de decisión unificados. Los árboles de decisión con los que se trabaja no superan un nivel 5 de profundidad, empleando no más de 4 variables o características de entrada (todo ello gracias a la optimización en el conjunto de selección de características realizado a través del método de filtraje basado en ranking citado en el apartado 2).

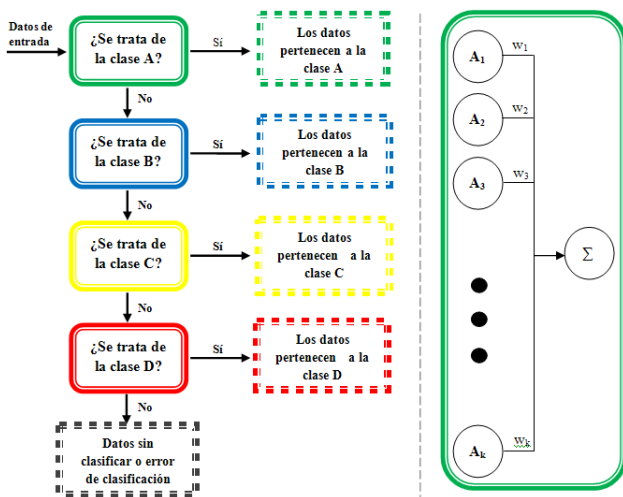


Figura 5: a) Sistema jerárquico de decisión basado en SVMs. b) Ejemplo de composición de la decisión para la clase A.

Combinando ambos paradigmas de clasificación se genera una estructura híbrida de modo que la decisión final depende de los resultados obtenidos en cada una de las unidades funcionales componentes. Al igual que para la definición de las entidades de decisión, se extrapola el concepto de peso en un nivel de abstracción mayor. Estos pesos se pueden aplicar directamente a través de una función matemática ponderada, si bien en este trabajo se sugiere la posibilidad de que influyan más bien desde un punto de vista de valor en la decisión adoptada. Así por ejemplo, si el peso de un clasificador (calculado a partir de los ratios de acierto individuales) dado es mayor que el de otro, ello supondrá que su decisión sea tenida en cuenta de forma preferente. En cualquier caso la decisión final vendrá dada por la suma ponderada de las decisiones tomadas por cada uno de los clasificadores componentes:

$$\begin{aligned} \text{Decisión} &= p \times \text{Decisión}_{SVM} + (1 - p) \times \text{Decisión}_{\text{Árboles}} = \\ &= p \times \sum_{k=1}^M w_k(d_k)_{SVM} + (1 - p) \times \sum_{k=1}^N v_k(d_k)_{\text{Árboles}} \quad (1) \end{aligned}$$

con M y N el número de clasificadores SVM (entendido como en la figura 5.a) y de árboles respectivamente que constituyen el sistema híbrido, w_i el peso (relevancia en la decisión) para el clasificador SVM i , v_i el peso para el clasificador basado en árbol i y p la ponderación en el primer nivel sobre la decisión ofrecida para el sistema SVM (evidentemente $1-p$ determina la ponderación para el otro bloque). Todo esto se esquematiza en la figura 6.

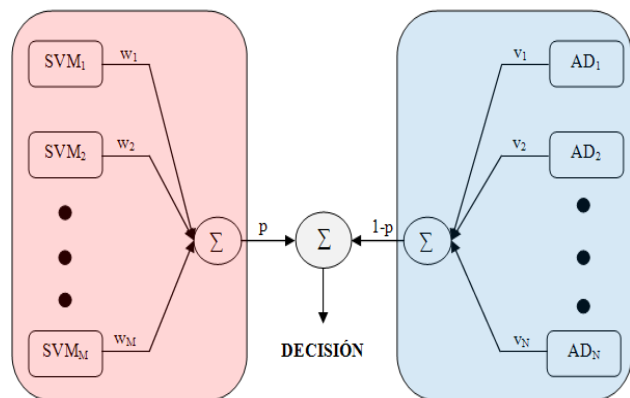


Figura 6: Sistema de decisión híbrido basado en SVM y Árboles de Decisión (AD). Las entradas de cada clasificador son diferentes puesto que las características empleadas en cada caso difieren. Teniendo en cuenta el ranking u ordenación de características obtenido en el proceso de selección, se generan tantos clasificadores como combinaciones de dos características en el caso de los basados en SVM y combinaciones de cuatro para los AD. Los clasificadores son entrenados individualmente con la información extraída a partir de las correspondientes características. El testeo permite definir los pesos de cada uno de ellos.

4 CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha presentado una aplicación directa de un modelo híbrido de clasificación dentro de una metodología original para el reconocimiento automático de actividades físicas. Partiendo de las ventajas que ofrecen los paradigmas SVM y los árboles de decisión se ha definido una posible aproximación basada en un modelo ponderado compuesto por subclasificadores que trata de mezclar eficientemente ambas metodologías.

Uno de los puntos más interesantes a trabajar radica en la idea de convertir este sistema en un esquema autoadaptativo y autoorganizado a través de la actualización de los pesos definidos, esto es, que la información que se obtenga de la utilización del sistema por usuarios independientes a los usados para entrenar los clasificadores sea empleada para un constante refinamiento de las ponderaciones, de forma parecida a como sucede en las redes neuronales adaptativas. Las ventajas en este sentido son grandes, sobretudo cuando el número de nodos constituyentes es alto. Por otra parte se ha pensado en la definición de una versión híbrida sustentada en un modelo bayesiano como la que se muestra en la figura 7.

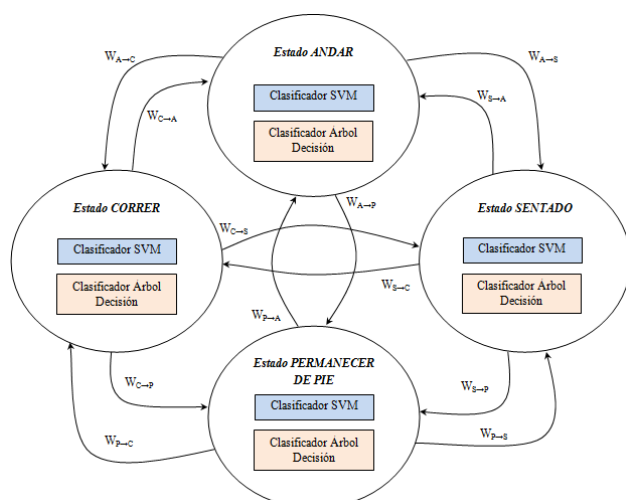


Figura 7: Sistema híbrido bayesiano basado en arquitecturas SVM y Árboles de Decisión. Los resultados del clasificador híbrido se combinan con probabilidades entre estados para generar la decisión final.

Agradecimientos

Este trabajo ha sido financiado por el Proyecto Nacional CICYT TIN2007-60587 y los Proyectos P07-TIC-02768 y P07-TIC-02906 de la Junta de Andalucía.

Queremos expresar nuestra gratitud al Profesor Stephen S. Intille, Director Tecnológico del Consorcio House_n del Departamento de Arquitectura de Computadores del MIT, por los datos experimentales cedidos.

Referencias

- [1] J. Baek, G. Lee, W. Park y B.J. Yun. Accelerometer Signal Processing for User Activity Detection. 2004.
- [2] L. Bao y S.S. Intille. Activity Recognition from User-Annotated Acceleration Data. 2004.
- [3] A.G. Bonomi, A.H.C. Goris, B. Yin y K. R. Westerterp. Detection of Type, Duration, and Intensity of Physical Activity Using an Accelerometer. *Medicine & Science in Sports & Exercise*. 2009.
- [4] L. Breiman, J. Friedman, R. Olshen y C. Stone. Classification and Regression Trees. 1984.
- [5] N. Crampton, K. Fox, H. Johnston y A. Whitehead. Dance, Dance Evolution: Accelerometer Sensor Networks as Input to Video Games. *IEEE International Workshop on Haptic Audio Visual Environments and their Applications*. 2007.
- [6] M. Ermes, J. Pärkka, J. Mantyjarvi y I. Korhonen. Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions. *IEEE Trans. Inf. Technol. Biomed.* 2008.
- [7] H. Koskimaki, V. Huikari, P. Siirtola, P. Laurinen y J. Roning. Activity recognition using a wrist-worn inertial measurement unit: A case study for industrial assembly lines. *17th Mediterranean Conference on Control and Automation*. 2009.
- [8] S.W. Lee y K. Mase. Activity and location recognition using wearable sensors. 2002.
- [9] J. Lester, T. Choudhury, N. Kern, G. Borriello y B. Hannaford. A Hybrid Discriminative/Generative Approach for Modeling Human Activities. 2005.
- [10] J. Mantyjarvi, J. Himberg y T. Seppanen. Recognizing human motion with multiple acceleration sensors. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*. 2001.
- [11] U. Maurer, A. Rower, A. Smailagic y D. Siwwiorek. Location and activity recognition using eWatch: a wearable sensor platform Ambient Intelligence in Everyday Life. 2006.
- [12] M. McCue, J. Hodgins y A. Bargteil. Telerehabilitation in Employment/Community Supports Using Videobased. Activity Recognition. *RERC on Telerehabilitation*. 2008.
- [13] E. Munguia, S.S. Intille, W. Haskell, K. Larson, J. Wright, A. King y R. Friedman. Real-Time Recognition of Physical Activities and Their Intensities Using Wireless Accelerometers and a Heart Rate Monitor. 2007.
- [14] N. Ravi, N. Dandekar, P. Mysore y M.L. Littman. Activity Recognition from Accelerometer Data. *American Association for Artificial Intelligence*. 2005.
- [15] B. Scholkopf y A.J. Smola. Learning with Kernels. 2002.

- [16] C. Zhu y W. Sheng. Multi-sensor Fusion for Human Daily Activity Recognition in Robot-assisted Living. 2009.
- [17] S. Zuolei, M. Xuchu, T. Weifeng y Z. Xiangfen. Activity classification and dead reckoning for pedestrian navigation with wearable sensors. 2008.

MEJORAS EN EL DISEÑO MULTI OBJETIVO DE REDES DE FUNCIONES DE BASE RADIAL

Pedro L. López¹ Antonio J. Rivera¹ Cristóbal J. Carmona¹ María Dolores Pérez-Godoy¹

¹ Departamento de Informática. Universidad de Jaén {plopez, arivera, ccarmona, lperez}@ujaen.es

Resumen

En este artículo se presenta un nuevo elemento poblacional a considerar en el diseño de algoritmos evolutivos multiobjetivo para la optimización de Redes de Funciones de Base Radial. Concretamente, se divide la población en subpoblaciones virtuales, donde cada subpoblación está compuesta por individuos o redes con el mismo número de neuronas. El objetivo será mantener el mejor individuo (individuo elite) de cada subpoblación entre generaciones, consiguiendo aumentar la diversidad de la población. Este elemento se implementa en el algoritmo EMORBFN, ya presentado por los autores, y se aplica a la tarea de clasificación. En los resultados se puede observar cómo la introducción de este nuevo elemento mejora el comportamiento general del algoritmo.

Palabras Clave: Diseño Multiobjetivo, Redes de Funciones de Base Radial, Elitismo.

1 INTRODUCCION

El diseño de cualquier modelo o componente en cualquier área de ciencia e ingeniería se ha convertido en un problema donde se trata de buscar una solución de compromiso entre todos los objetivos que se intentan cumplir. En general, estos objetivos entran en conflicto y desaparece el concepto de diseño óptimo convencional. Estos problemas, denominados problemas de optimización multiobjetivo, son aquellos en los que no existe una única solución sino un conjunto de soluciones, cada una de ellas con sus ventajas frente a las demás, pero donde ninguna es mejor en todos los aspectos. A partir de este abanico de modelos se elige la solución o soluciones finales en función del conocimiento experto que se maneje.

En nuestro caso, los modelos a diseñar son Redes de Funciones de Base Radial (RBFNs), uno de los paradigmas más importantes dentro del campo de las Redes Neuronales. Este modelo de cómputo ha demostrado su solvencia a la hora de abordar problemas de aproximación de funciones [9], clasificación [4] o predicción de series temporales [21]. Las Funciones de Base Radial (RBFs) se usaron inicialmente en interpolación numérica y aproximación funcional [18]. A finales de los 80 tienen lugar las primeras investigaciones de redes neuronales basadas en RBFs [3][15].

Son muchas las características que despiertan el interés por este tipo de redes, entre las cuales cabe destacar: su topología simple con tan solo una capa oculta; su capacidad de ser un aproximador universal [16]; el ser de los pocos modelos de red neuronal interpretable, ya que pueden extraerse reglas difusas a partir de ellos [13], etc.

El diseño de una RBFN se puede abordar desde el punto de vista de la computación evolutiva [1][4]. En la mayoría de estas propuestas, un individuo representa una RBFN completa, que evoluciona intentando siempre optimizar su precisión.

Contemplar sólo el objetivo de la precisión a la hora de optimizar RBFNs, como en la computación evolutiva tradicional, puede conducir a obtener redes con un número elevado de RBFs. Esto se debe a que es más fácil reducir el error con RBFNs que contienen muchas RBFs que con RBFNs que contienen pocas RBFs. Para solventar este problema aparece la Optimización Multiobjetivo Evolutiva (EMO) [5]. En nuestro caso, y como suele ser habitual, además de optimizar la precisión de la red, se optimizará también la complejidad o número de neuronas de ésta.

Dentro de la EMO no son muchos los trabajos que se pueden encontrar en la literatura especializada en el campo del diseño multiobjetivo evolutivo de RBFNs. Algunos de estos trabajos se enumeran en la siguiente sección. Esta carencia de trabajos está relacionada con la

dificultad que encuentran este tipo de algoritmos a la hora de lograr diversidad en la población.

Para solucionar este problema se propone introducir un elitismo especial en la población a nivel de subpoblaciones, estando cada subpoblación definida por el número de RBFs de la red. Esto implica que todas las RBFNs con un número igual de RBFs estarán en la misma subpoblación y que al mejor individuo de cada una de esas subpoblaciones virtuales se le mantendrá entre generaciones.

Concretamente, en este trabajo se ha implementado una mejora para EMORBFN [14], un algoritmo multiobjetivo para el diseño de RBFNs, en la cual se mantienen estos individuos élite por subpoblación, aportando diversidad y una mejora de los resultados finales.

En la Sección 2 se describen las RBFNs y su optimización evolutiva. En la Sección 3 se detalla el funcionamiento del algoritmo EMORBFN. En la Sección 4 se muestran y analizan los resultados obtenidos. Para terminar, en la Sección 5 se exponen las conclusiones del trabajo.

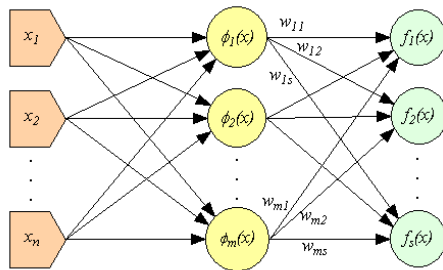


Figura 1: Topología de una RBFN

2 RBFNs Y SU DISEÑO

Una Red de Funciones de Base Radial es un tipo de red neuronal hacia delante con tres capas: la capa de entrada con n nodos, una capa oculta con m neuronas o RBFs, y una capa de salida con uno o varios nodos (Figura 1). Las m neuronas/RBFs de la capa oculta ofrecen una activación simétrica radial $\phi_i: R^n \rightarrow R$, que puede tomar diferentes formas. La más común es la función gaussiana, que viene dada por la expresión $\phi_i(\vec{x}) = \phi_i(e^{-\|\vec{x}-\vec{c}_i\|^2/d_i^2})$, donde $\vec{c}_i \in R^n$ es el centro de la función base ϕ_i , $d_i \in R$ es el radio y como $\|\cdot\|$ suele utilizarse la norma euclídea en R^n . Los nodos de salida implementan la siguiente ecuación:

$$f_j(\vec{x}) = \sum_{i=1}^m w_{ij} \phi_i(\vec{x}) \quad (1)$$

En un escenario de clasificación, la RBFN debe establecer una relación entre un espacio de entrada X^n hacia un conjunto finito de clases Y con k clases. De esta manera un típico conjunto de entrenamiento S sería:

$$S = \{(\vec{x}_u, y_u) \mid x_u \in X^n, y_u \in Y, u = 1, \dots, p\} \quad (2)$$

donde $\vec{x}_u$ es el vector de características e y_u es la clase a la que pertenece. Normalmente, en un problema de clasificación, el número de salidas de la RBFN se corresponde con el número de clases (k). Para entrenar la RBFN, la pertenencia a la clase y_u se codifica en un vector binario $\vec{z}_u \in \{0,1\}^k$ a través de la relación $\vec{z}_u^i = 1$ si $y_u = i$, y $\vec{z}_u^i = 0$ en otro caso. La clase obtenida por la red ante una determinada entrada será la salida con un mayor grado de activación.

En cualquier caso, el objetivo a la hora de diseñar una RBFN es determinar el centro y el radio que caracteriza cada RBF así como su peso para cada salida de la red. Las técnicas empleadas en el diseño de RBFNs son muy diversas.

El algoritmo típico de diseño de RBFNs tiene dos etapas. En la primera etapa se determinan los centros y los radios de las RBFs, mientras que en la segunda etapa se calculan sus pesos. Para determinar los centros y los radios se pueden emplear técnicas de clustering [17]. En la segunda etapa, para calcular los pesos, se pueden utilizar algoritmos como el LMS [22], SVD [8], etc.

Hoy en día la computación evolutiva es uno de los paradigmas más importantes en el diseño de RBFNs [4][11]. Estos algoritmos suelen utilizar una codificación de tipo Pittsburg, en la que un individuo representa una red completa. La mayoría de estos algoritmos sólo optimizan la precisión de los individuos sin tener en cuenta la complejidad de las redes obtenidas. Como se ha comentado, contemplar un único objetivo puede conducir a obtener redes con un alto número de neuronas, ya que es más fácil reducir el error con un mayor número de neuronas que con un menor número de éstas. Es por esto que hay que tener en cuenta los paradigmas que nos proporciona la EMO para abordar este tipo de problemas.

Brevemente definiremos los conceptos de dominancia y de frente de Pareto dentro de la optimización multiobjetivo. Así se dice que un individuo A domina a otro individuo B si A no es peor que B en ninguno de los objetivos y A es mejor que B en algún objetivo. Dado un espacio de búsqueda el frente de Pareto óptimo se define como el conjunto de individuos no dominados de ese espacio de búsqueda.

Los Algoritmos Evolutivos Multiobjetivo (MOEAs) han sido usados en el diseño de RBFNs. Por ejemplo los

conocidos Algoritmos Genéticos Multiobjetivo (MOGAs) se han utilizado en [9][20] para la estimación o aproximación de funciones. En [19] se hibrida NSGA-II [6] con “rough-sets” para alcanzar un algoritmo multiobjetivo más eficiente. [12] usa “surrogates” distribuidos espacialmente para implementar las RBFNs. En [10] se presenta una aplicación que hace estimaciones en la reducción de un mineral. Hay también MOEAs específicos: en [23] se utiliza la técnica HRDGA (Hierarchical Rank Density Genetic Algorithm) para diseñar la topología y los parámetros de la red.

3 DESCRIPCIÓN DE EMORBFN

En este trabajo se presenta EMORBFN, un Algoritmo Genético Multiobjetivo para el diseño de RBFNs, donde los objetivos a optimizar son la precisión del modelo (maximizar el porcentaje de aciertos) y su complejidad (minimizar el número de RBFs). La principal característica a destacar de EMORBFN es que va a definir subpoblaciones virtuales de individuos con igual número de neuronas y que conservará entre generaciones al mejor individuo de cada subpoblación.

La justificación de esta propuesta se encuentra en que, tras analizar la evolución de una población compuesta por individuos/RBFNs en el marco de un algoritmo evolutivo multiobjetivo, se puede observar que los individuos élite de las subpoblaciones de gran tamaño pueden ser dominados por individuos con menos neuronas, desapareciendo así de la población. Esto se debe a dos causas:

- Durante las primeras iteraciones del algoritmo, los parámetros de las redes no han sido ajustados aún y es de esperar que tengan un porcentaje de clasificación similar. Esto provoca que redes con pocas RBFs dominen rápidamente a los individuos de mayor tamaño, haciendo que estos últimos sean descartados durante el reemplazamiento de la población.
- Por otro lado, las RBFNs con un mayor número de neuronas son más difíciles de aprender, ya que contienen más parámetros a determinar. Por ello, se necesitan más iteraciones para obtener redes que sean competitivas en cuanto a precisión.

Estos dos factores implican una fuerte pérdida de diversidad en las primeras etapas del algoritmo, dificultando la obtención de un conjunto de soluciones adecuadamente distribuido a lo largo del frente de Pareto. El operador de elitismo garantiza que el individuo de cada número de RBFs con mayor precisión pase a formar parte de la población en la siguiente generación. Con ello se pretende evitar la mencionada pérdida de diversidad, dando lugar a un mejor comportamiento del método.

Concretamente, EMORBFN determina todos los parámetros del modelo: número de RBFs y centro y radio de cada una de ellas. Se utiliza codificación real y un enfoque Pittsburgh, donde cada individuo representa una RBFN al completo. El problema multiobjetivo se aborda en el paso 6 mediante un procedimiento de reemplazo basado en el algoritmo NSGA-II [6]. A dicho algoritmo se le añade un operador de elitismo que permite conservar al menos un individuo de cada número de RBFs; así se garantiza la evolución de todas las subpoblaciones. La Figura 2 muestra los pasos básicos del algoritmo, que se detallan a continuación.

-
1. $P_t = \text{Inicialización}(\text{Datos})$
 2. Entrenamiento/Evaluación(P_t)
 3. $C_t = \text{Cruce}(P_t)$
 4. $M_t = \text{Mutación}(P_t)$
 5. Entrenamiento/Evaluación(C_t, M_t)
 6. $P_{t+1} = \text{Reemplazo+Elitismo}(P_t, C_t, M_t)$
 7. Si no se cumple la condición de parada, ir a 3.
-

Figura 2: Pasos básicos de EMORBFN

Inicialización. Se generan λ (tamaño de la población) RBFNs de manera que haya la misma cantidad de redes para cada número de RBFs; así, cada subpoblación tiene un tamaño similar. El centro $\vec{c}_i$ de cada RBF se determina

asignando un ejemplo de entrenamiento aleatoriamente, teniendo en cuenta que las RBFs de una misma red deben estar igualmente distribuidas por todas las clases. Una vez asignados los centros, el valor del radio d_i es la mitad de la distancia media entre los centros de la red. Los pesos w_{ij} se inicializan a cero.

Entrenamiento/Evaluación. Se utiliza el algoritmo LMS para determinar los pesos w_{ij} de cada red. A continuación se evalúa cada RBFN y se obtiene su error de clasificación.

Cruce. Durante el cruce se genera un porcentaje pc de nuevas redes. Para ello se eligen dos RBFNs padre aleatoriamente y se combinan formando un único hijo, cuyo tamaño será un número aleatorio entre el mínimo número de RBFs de los padres y la suma del número de RBFs de los padres. El nuevo hijo se genera insertando RBFs alternativamente de cada padre. Cada vez que se selecciona una RBF se busca en el otro padre si existe una RBF que se solape con la seleccionada. En caso afirmativo se combinan sus valores (centros y radios) usando el cruce BLX- α [7]. En caso contrario, se introduce la RBF seleccionada aplicando una mutación (ver *ModRBF* a continuación).

Mutación. Se han implementado cinco operadores de mutación, distinguiendo entre mutaciones aleatorias y mutaciones con información. Las redes a mutar se eligen aleatoriamente. Los operadores aleatorios son:

- *EliRBFAleatorio*. Elimina un porcentaje pm de RBFs elegidas de forma aleatoria.
- *InsRBFAleatorio*. Inserta un porcentaje pm de RBFs generadas aleatoriamente.
- *ModRBF*. Modifica un porcentaje pm de las RBFs de una red. Para cada RBF a mutar, el centro y el radio se modifican en un porcentaje aleatorio entre 0 y pr del valor del radio actual.

Los operadores de mutación con información son:

- *InsRBFInf*. Inserta un porcentaje pm de RBFs de forma que no se solapen con el resto de RBFs de la red.
- *EliRBFInf*. Elimina un porcentaje pm de las RBFs con menor peso medio de la red.

Reemplazo+Elitismo. Durante esta fase se seleccionan sin reemplazo los λ individuos que formarán la nueva generación (P_{t+1}), considerando las RBFNs de la población actual (P_t) y las generadas mediante cruce (C_t) y mutación (M_t). La implementación se basa en el algoritmo NSGA-II y en los conceptos de dominancia de Pareto. Se construyen los frentes de no-dominancia $F=\{F_1, F_2, \dots\}$, donde cada frente F_i está formado por individuos no-dominados entre sí y que dominan a todos los individuos de los frentes $F_{k>i}$.

En un primer paso, se recorren los frentes de no-dominancia de menor a mayor hasta que se encuentre un individuo de cada número de RBFs. Dicho individuo es considerado el mejor de su subpoblación, ya que las demás RBFNs del mismo tamaño estarán en frentes dominados por él, por lo que es incluido en la nueva población P_{t+1} . Así se asegura que las subpoblaciones conservan durante toda la evolución al mejor individuo encontrado hasta ese momento (elitismo).

Para completar la nueva población se van insertando los frentes $F_1, F_2, \dots$ hasta conseguir λ (tamaño de la población) individuos. Si al insertar un frente, éste tiene más individuos de los necesarios para completar P_{t+1} , se calcula la distancia de crowding entre los individuos del frente y se insertan en orden de mayor a menor distancia.

A la hora de insertar una RBFN en la nueva población P_{t+1} se implementa un operador que impide la introducción de individuos equivalentes. Consideramos que dos RBFNs son equivalentes cuando, teniendo el mismo tamaño, la distancia entre los centros de las dos RBFs más cercanas de cada red es menor que los radios de las mismas.

4 EXPERIMENTACION Y RESULTADOS

El objetivo de los experimentos llevados a cabo es comprobar la influencia del elitismo en el comportamiento de EMORBFN a la hora de obtener RBFNs de diferentes complejidades con una alta capacidad de clasificación.

Para evaluar el efecto del elitismo se ha experimentado con dos versiones del algoritmo: *EMORBFNBasico* y *EMORBFNElitista*. Ambas versiones son idénticas exceptuando el paso 6 (Figura 2), donde la implementación básica no incluye elitismo y selecciona los individuos atendiendo únicamente al frente de dominancia en el que se encuentran.

Se han seleccionado tres problemas de clasificación estándar (pima, wbc y wine) obtenidos del UCI Machine Learning Repository [1]. Con el objetivo de acotar el espacio de búsqueda, se ha establecido el tamaño mínimo de una red al número de clases del problema y el tamaño máximo a cuatro veces dicho número de clases. Esta decisión se justifica porque tal y como se demuestra en los resultados, las RBFNs no necesitan un elevado número de neuronas para conseguir una precisión importante. Además permite mantener un número relativamente pequeño de individuos en la población. Para evaluar la precisión del método se ha utilizado la técnica de 10-fold cross validation. Cada experimento se ha repetido 5 veces, sumando 50 ejecuciones total.

La Tabla 1 muestra los parámetros utilizados en la configuración de EMORBFN. Los valores se han elegido entre los habituales para este tipo de algoritmos. Las Tablas 2-4 muestran los frentes obtenidos por ambos algoritmos en cada base de datos. Para cada número de RBFs se muestra la media del porcentaje de clasificación en test del mejor individuo de cada ejecución. Las Figuras 3-6 resumen gráficamente los resultados obtenidos.

Tabla 1. Parámetros de EMORBFN

Parámetro	Valor
Número de iteraciones	200
Tamaño de la población (λ)	40
Probabilidad de cruce	0,6
Probabilidad de mutación	0,1
pm	0,5
pr	0,5

Tabla 2. Resultados para el problema Pima.

Nº de RBFs	% de Clasificación	
	EMORBFNBásico	EMORBFNELitista
2	75,06	74,54
3	74,79	75,64
4	73,52	75,68
5	74,57	75,07
6	73,00	76,05
7	75,44	75,16
8	75,88	77,10

Tabla 3. Resultados para el problema Wbcd.

Nº de RBFs	% de Clasificación	
	EMORBFNBásico	EMORBFNELitista
2	95,16	96,28
3	96,37	96,79
4	95,84	96,61
5	96,50	97,23
6	96,18	96,30
7	95,84	96,86
8	95,87	96,90

Tabla 4. Resultados para el problema Wine.

Nº de RBFs	% de Clasificación	
	EMORBFNBásico	EMORBFNELitista
3	89,36	88,51
4	88,75	93,71
5	93,33	96,09
6	94,38	94,58
7	94,67	96,03
8	94,37	95,10
9	94,08	95,79
10	95,37	96,27
11	98,61	96,80
12	95,56	97,22

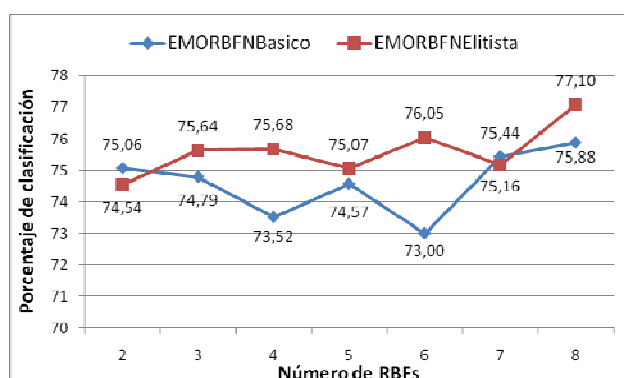


Figura 3. Resultados para el problema Pima.

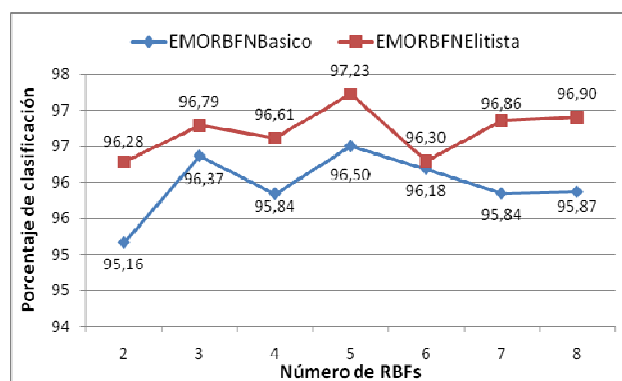


Figura 4. Resultados para el problema Wbcd.

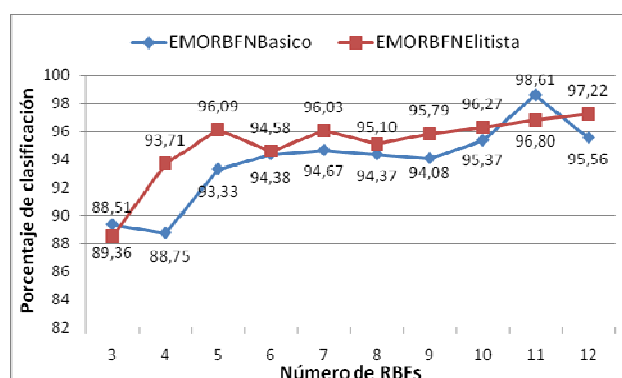


Figura 5. Resultados para el problema Wine.

En la evolución de las ejecuciones del algoritmo EMORBFNELitista hemos observado una mayor diversidad que para el algoritmo EMORBFNBásico. En las tablas de resultados observamos que EMORBFNELitista mejora con respecto a la versión básica. Exceptuando algunos casos puntuales, el operador de elitismo genera modelos con mayor precisión para redes con el mismo número de neuronas.

En general, los frentes obtenidos con EMORBFNELitista son más homogéneos, presentando altibajos menos pronunciados. Además, en los frentes de la versión elitista se observa de forma más clara la tendencia creciente en el porcentaje de clasificación a medida que aumentamos la complejidad de las RBFNs.

Si nos centramos únicamente en la precisión de las redes, EMORBFNELitista obtiene el mejor porcentaje de clasificación global tanto en Pima como en Wbcd. En cambio, en el problema Wine, EMORBFNBásico obtiene la mejor precisión global debido a la presencia de un pico en el caso de redes con 11 neuronas.

5 CONCLUSIONES

En este trabajo se presenta un nuevo elemento para aumentar la diversidad en el diseño multiobjetivo de RBFNs. Tras analizar el funcionamiento de estos

algoritmos se comprueba que, para la tarea de clasificación, sufren una fuerte pérdida de diversidad de individuos cuando el número de neuronas empieza a incrementarse. Una de las principales causas es que estos individuos necesitan un número mayor de generaciones para conseguir alcanzar un nivel adecuado de entrenamiento y eficiencia.

Para solventar este problema se ha propuesto definir un conjunto de subpoblaciones virtuales, donde cada subpoblación estaría compuesta por los individuos o redes de un determinado número de neuronas. El objetivo será mantener a los mejores individuos en eficiencia de cada subpoblación entre generaciones. Con esto hemos conseguido una mejora en la diversidad en la población y en la eficiencia general de los resultados alcanzados.

Agradecimientos

Este trabajo está soportado por el proyecto del Ministerio de Ciencia e Innovación TIN2008-06681-C06-02 y por el de la Junta de Andalucía TIC-3928.

Referencias

- [1] Asuncion, A.; Newman, DJ. UCI machine learning repository. School of Information and Computer Science, University of California, Irvine, CA. <http://www.ics.uci.edu/mllearn/MLRepository.htm> 1. 2007.
- [2] Bäck, T.; Hammel, U.; Schwefel, H. Evolutionary computation: comments on the history and current state. *IEEE Trans. on Evolutionary Computation*, 1(1):3-17, 1997.
- [3] Broomhead, D.; Lowe, D. Multivariable functional interpolation and adaptive networks. *Complex System*. 2:321-355, 1988.
- [4] Buchtala, O.; Klimek, M.; Sick, B. Evolutionary optimization of radial basis function classifiers for data mining applications. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 35(5):928-947, 2005.
- [5] Deb, K. Multi-objective optimization using evolutionary algorithms. Wiley, 1st ed. 2001
- [6] Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation* 6(2):182-197, 2002.
- [7] Eshelman, L.J.; Schaffer, JD. Real-coded genetic algorithms and interval-schemata. Whitley, L.D. ed. *Foundations of Genetic Algorithms* 21:187-202. Morgan Kaufmann. 1993.
- [8] Golub, G.; Van Loan, C. Matrix computations. J. Hopkins University Press, 3rd ed., 1996.
- [9] González, J., Rojas, I., Ortega, J., Pomares, H., Fernández, J., Fco, A. Multiobjective evolutionary optimization of the size, shape, and position parameters of radial basis function networks for function approximation. *IEEE Transactions on Neural Networks*, 14(6):1478-1495, 2003.
- [10] Guillén, A.; Pomares, H.; Rojas, I.; González, J.; Herrera, L.J.; Rojas, F.; Valenzuela, O.; Output value-based initialization for radial basis function neural networks. *Neural Process Letters*. (2007)
- [11] Harpham, C.; Dawson, C.; Brown, M.; A review of genetic algorithms applied to training radial basis function networks. *Neural Computation Applications* 13:193-201. 2004
- [12] Isaacs, A.; Ray, T.; Smith, W. An evolutionary algorithm with spatially distributed surrogates for multiobjective optimization. *LNAI* 4828:257-268. 2007
- [13] Jin, Y.; Sendhoff, B.; Extracting interpretable fuzzy rules from RBF networks. *Neural Process Letters* 17(2):149-164. 2003
- [14] López, P.; Rivera, A.; Pérez-Godoy, M.; del Jesus, M.; Carmona, C. EMORBFN: An Evolutionary Multiobjective Optimization algorithm for RBFN design. *Lecture Notes in Computer Science*. 5517: 752-759. 2009
- [15] Moody, J.; Darken, C, J. Fast learning in networks of locally-tuned processing units. *Neural Computing*. 1:281-294, 1989.
- [16] Park, J.; Sandberg, I. Universal approximation using radial-basis function networks. *Neural Comput.*, 3:246-257, 1991.
- [17] Pedrycz, W. Conditional fuzzy clustering in the design of radial basis function neural networks. *IEEE Transactions on Neural Networks* 9(4):601-612, 1998.
- [18] Powell, M. Radial basis functions for multivariable interpolation: A review. In *IMA. Conf. on Algorithms for the approximation of functions and data*, 143-167, 1985.
- [19] Santana-Quintero LV, Coello CA, Hernández-Díaz AG. Hybridizing surrogate techniques, rough sets and evolutionary algorithms to efficiently solve multi-objective optimization problems. *GECCO'08* 763-764, 2007
- [20] Teixeira, CA; Ruano MG, Ruano AE, Pereira WCA. A soft-computing methodology for non invasive time-spatial temperature estimation. *IEEE Transactions on Bio-Medical Engineering* 55:2:572-580, 2008
- [21] Whitehead, B; Choate, T. Cooperative-competitive genetic evolution of Radial Basis Function centers and widths for time series prediction. *IEEE Transactions on Neural Networks*, 7(4):869-880, 1996.
- [22] Widrow, B.; Lehr, M.A. 30 Years of adaptive neural networks: perceptron, madaline and backpropagation. *Proceedings of the IEEE*, 78(9):1415-1442, 1990.
- [23] Yen GG. Multi-Objective evolutionary algorithm for radial basis function neural network design. *Studies in Computational Intelligence* 16:221-239, 2006

SISTEMA NEURO-DIFUSO PARA EL MODELADO DE LA EVOLUCIÓN DEL COLOR. MODELADO DE UN PROCESO DE BLANQUEAMIENTO DENTAL

L. J. Herrera¹ H. Pomares¹ I. Rojas¹ J. Santana² R. Pulgar²
R. Ghinea³ M. M. Perez³

¹ Departamento de Arquitectura y Tecnología de los Computadores. Universidad de Granada, jherrera@atc.ugr.es

² Departamento de Estomatología. Universidad de Granada

³ Departamento del Óptica. Universidad de Granada

Resumen

El modelado del cambio de color es una problemática esencial en diferentes aplicaciones colorimétricas. En concreto, la aproximación al color esperado tras un proceso físico, químico o natural es de vital importancia en la industria y en la ciencia del color en general. Este trabajo propone un enfoque neuro difuso para la resolución del problema de modelado de cambio cromático. En concreto se aplicará el enfoque propuesto a un problema de gran importancia desde el punto de vista clínico en odontología tal y como es el modelado de la variación cromática del diente tras el proceso de blanqueamiento. La disposición de herramientas que ayuden a predecir dichos cambios, partiendo de la cromaticidad inicial del diente y el protocolo establecido, puede resolver el problema de falta de información sobre el color dental esperado tras un tratamiento concreto y ayudará al odontólogo en la toma de decisiones sobre el protocolo adecuado y su información al paciente.

Palabras Clave: Sistemas Neuro Difusos, Modelado Local y Global, Modelado del Color, Espacio de Color CIELAB.

1 INTRODUCCIÓN

El modelado de variaciones del color tras un proceso físico, químico o natural puede ser una importante herramienta en la práctica industrial. En concreto, la aproximación al color esperado tras un determinado proceso, es de vital importancia en cualquier aplicación colorimétrica y en la ciencia del color, en general [10] [11].

Dentro de las numerosas metodologías de modelado existentes en las áreas de la Estadística y el Aprendizaje Automático o Machine Learning, los sistemas neuro-difusos presentan grandes ventajas frente a otras alternativas, al combinar las capacidades aproximativas no lineales de las redes neuronales, junto con las propiedades interpretativas de los modelos difusos. Componen una metodología que proporciona una manera simple de obtener una conclusión a partir de información de entrada imprecisa. El diseño y optimización de estos modelos, en el que los expertos pueden intervenir, resulta en un conjunto de reglas a partir de un repertorio de mediciones disponibles, que facilita el análisis y utilización de dichos modelos para un estudio más claro y en profundidad del fenómeno físico que se estaba modelando.

Este trabajo presenta un modelo neuro-difuso orientado a la modelización del cambio de color en un sistema. Las mediciones de color en óptica suelen obtenerse y representarse utilizando un espacio de representación tridimensional. En este trabajo nos centramos en el espacio CIELAB y exponemos la aplicación de la metodología propuesta a un problema de modelado del cambio cromático producido tras un proceso de blanqueamiento dental.

El blanqueamiento dental es cada vez más popular entre pacientes y dentistas al ser una opción sencilla y no invasiva que proporciona blanqueado y abrillantado de los dientes. Su predicción puede ayudar a proporcionar guías sobre los resultados esperados tras dichos tratamientos. Estas guías deben contener shades/tonos que contemplen en gran medida la población. En este sentido, un aspecto de gran interés, y no abordado hasta la actualidad, es el estudio de cambio de color para pacientes con tinción de tetraciclina, lo cuales suelen recurrir a tratamientos de blanqueamiento para eliminar dicha tinción. Por tanto, en este trabajo se tratará con dos grupos de pacientes diferentes, (pacientes tetraciclínicos) y otros sin tinción por tetraciclina (pacientes “normales”) ya

que la respuesta al tratamiento de ambos grupos de pacientes, es decir el cambio cromático, puede diferir.

El resto del trabajo se organiza de la siguiente forma: La sección 2 describe el espacio CIELAB y los valores de mediciones obtenidos para el problema dado. La sección 3 presenta el modelo neuro-difuso diseñado para efectuar el modelado. La sección 4 presenta los resultados obtenidos y el conjunto de reglas diseñadas. La sección 5 presenta las conclusiones más relevantes de este estudio.

2 REPRESENTACIÓN DEL COLOR: ESPACIO CIELAB. COLOR DE LOS DIENTES

La localización de cualquier color en el sistema CIELAB está determinada por sus coordenadas L^* , a^* y b^* (Figura 1). L^* es la luminosidad y está cuantificada en una escala de 0 (negro) a 100 (blanco). La componente a^* representa la diferencia entre rojo ($+a^*$) y verde ($-a^*$) y la componente b^* representa la diferencia entre amarillo ($+b^*$) y azul ($-b^*$). Estos dos componentes están inspirados en la teoría de los colores oponentes.

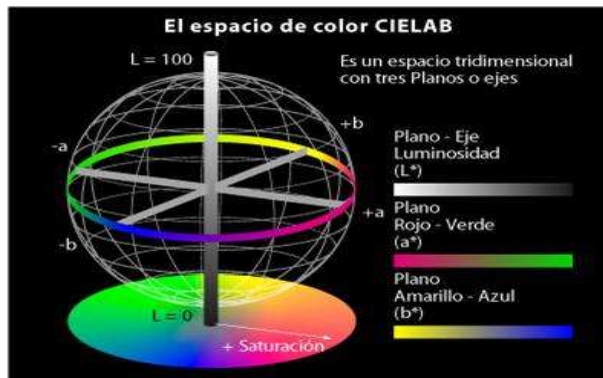


Figura 1: Espacio CIELAB de color.

En la actualidad, el sistema de representación CIELAB es el más empleado en la práctica industrial relacionada con alimentos, tejidos, etc. y es el recomendado por la CIE (Comisión Internacional de l'Eclaire) para la evaluación del color [2]. Además, debido a su uniformidad respecto a otros espacios, CIELAB es el sistema en el que están basadas las más actuales fórmulas de diferencia de color: CIELAB, CIE94 y CIEDE2000.

Por todo ello, en odontología y más en concreto en los trabajos de investigación relacionados, es usual el empleo del espacio CIELAB para la especificación del color del diente y de los materiales de restauración den-

tal. Más aún, la reciente guía clínica de color, VITA 3D-Máster, está basada en la estructura de CIELAB.

3 SISTEMA NEURO-DIFUSO PARA EL MODELADO DEL CAMBIO CROMÁTICO

La lógica difusa está adquiriendo cada vez mayor atención en la literatura de óptica y de color [1] [8]. Además, en el área de la odontología, algunos trabajos han utilizado lógica difusa para caracterizar la superficie del diente en la práctica parodontológica y la eliminación de cálculos dentales o para detectar y cuantificar la placa dental [7].

Este trabajo presenta una metodología para el estudio del cambio de color durante un tratamiento de blanqueamiento basado en un sistema neuro-difuso, enfoque no utilizado aún en la literatura de color dental o de cambio de color en general. Dado que el objetivo es encontrar el color tras el proceso estudiado, deberían estudiarse en un principio cuáles son las variables que pueden influir en dicho color final. En el enfoque propuesto en este trabajo, en ausencia de otra información, se restringirá el modelo para considerar tan solo el color inicial del diente para modelar el color final del mismo, de forma que el sistema difuso abstraiga el color final esperado según el color inicial del diente. El estudio de ese proceso de cambio de color en el espacio tridimensional CIELAB (dimensiones L^* , a^* , b^*), y considerando sistemas difusos de tipo TSK (con consecuentes numéricos), definiría por tanto reglas de la forma siguiente,

$$\begin{aligned} \text{SI } L_i^* \text{ es } \mu_{L^*}^k \text{ Y } a_i^* \text{ es } \mu_{a^*}^k \text{ Y } b_i^* \text{ es } \mu_{b^*}^k \\ \text{ENTONCES} \\ L_f^* = \tilde{L}_f^{*k} \text{ Y } a_f^* = \tilde{a}_f^{*k} \text{ Y } b_f^* = \tilde{b}_f^{*k} \end{aligned} \quad (1)$$

donde el sistema por tanto indicaría el valor de color tras el proceso estimado $\{\tilde{L}_f^*(\vec{x}), \tilde{a}_f^*(\vec{x}), \tilde{b}_f^*(\vec{x})\}$ para un valor de color inicial $\vec{x} = \{L_i^*, a_i^*, b_i^*\}$. El sistema de inferencia difuso utilizaría la siguiente función de salida del sistema difuso global $\tilde{y}(\vec{x})$ para cada una de las tres salidas estimadas correspondientes a las coordenadas CIELAB $\tilde{L}_f^*(\vec{x})$, $\tilde{a}_f^*(\vec{x})$ y $\tilde{b}_f^*(\vec{x})$:

$$\tilde{y}(\vec{x}) = \frac{\sum_{k=1}^K \mu^k(\vec{x}) \tilde{y}^k}{\sum_{k=1}^K \mu^k(\vec{x})} \quad (2)$$

La construcción de un sistema difuso óptimo implica la elección del número de reglas del sistema K , y la construcción adecuada de los antecedentes $\{\mu_{L^*}^k, \mu_{a^*}^k, \mu_{b^*}^k\}$

y consecuentes $\{L_f^* = \tilde{L}_f^{*k}, a_f^* = \tilde{a}_f^{*k}, b_f^* = \tilde{b}_f^{*k}\}$ para cada regla k . La optimización de estos parámetros del sistema difuso a partir de un conjunto de datos de entrada-salida (E/S), podría realizarse mediante métodos automáticos [5] [12] o mediante sistemas difusos genéticos [3]. La distribución de las reglas en concreto, podría seguir un particionamiento basado en grid, o un particionamiento sparse [12].

En el problema concreto propuesto en este trabajo, se definirán el número de reglas K y los conjuntos difusos en los antecedentes de dichas reglas según las medidas colorimétricas de los tonos definidos en la guía dental Vita-Classical, como información a priori sobre la categorización del color de los dientes. Esta guía es la más utilizada por los dentistas en las consultas debido a su relación con los sistemas de restauración dental y su utilización frecuente en estudios sobre blanqueamiento. Esto permitirá por tanto que el conjunto de reglas obtenido sea comprensible de forma más directa a nivel clínico.

Los valores en los consecuentes de las reglas sin embargo se optimizarán en un proceso de entrenamiento a partir de un conjunto de ejemplos de E/S. El conjunto de datos de entrenamiento serán mediciones de color antes (entrada) y después (salida) del cambio cromático. La obtención de los consecuentes óptimos en un modelo difuso se realiza mediante la minimización de la función del error cuadrático medio, que lleva a la creación de sistemas de ecuaciones lineales que pueden resolverse utilizando cualquier método matemático destinado a tal efecto [5]. En este trabajo se utilizó Descomposición de Valores Singulares para resolver los tres sistemas de ecuaciones construidos, uno para cada coordenada CIELAB, y obtener los coeficientes de los consecuentes óptimos para cada regla k : $\tilde{L}_f^{*k}$, $\tilde{a}_f^{*k}$ y $\tilde{b}_f^{*k}$.

El proceso de inferencia difuso diseñado modelará por tanto el comportamiento físico del cambio de color utilizando el conjunto de reglas construido. Ahora bien, la obtención de reglas significativas del proceso modelado no es sin embargo algo totalmente inherente al diseño de un modelo difuso [5] [6]: la interrelación de las reglas componentes del modelo es relevante en el proceso de entrenamiento de un sistema difuso, y los valores de los consecuentes obtenidos depende altamente del solapamiento existente entre las funciones de pertenencia en los antecedentes de las reglas.

Para evitar esta dependencia, se ha utilizado un proceso de inferencia modificada novedoso, adaptado de redes de funciones de base radial [4], y que corrige estas interacciones entre las reglas para evitar su solapamiento, inspiradas por estudios previos sobre sistemas difusos basados en grid [5]. Esta modificación

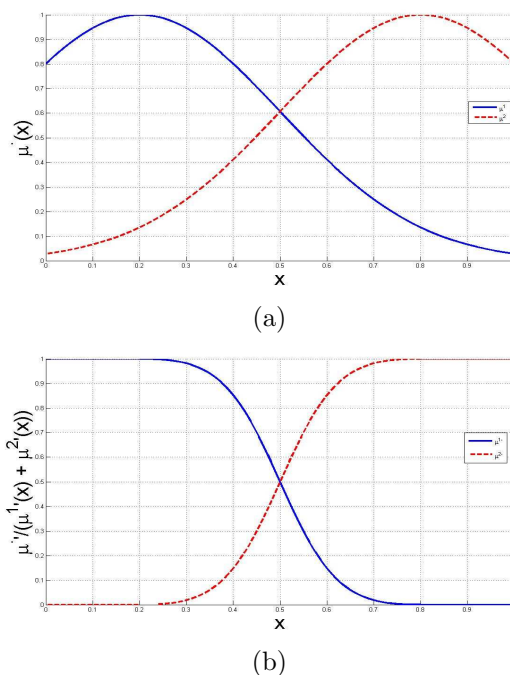


Figura 2: a) Funciones de pertenencia μ^1 y μ^2 originales para el ejemplo unidimensional. b) Funciones de pertenencia finales normalizadas utilizando el cálculo modificado $\frac{\mu^1(x)}{\mu^1(x) + \mu^2(x)}$ y $\frac{\mu^2(x)}{\mu^1(x) + \mu^2(x)}$.

consiste en forzar a que las activaciones de las reglas se anulen para todos los centros de reglas del sistema salvo para el propio centro, conservando sin embargo las propiedades de continuidad y derivabilidad e interpolativas del sistema de inferencia.

Tomando como ejemplo sencillo un problema unidimensional con dominio $[0,1]$ y con dos reglas con funciones de pertenencia Gaussianas, centradas, por ejemplo en $c^1 = 0.2$ y $c^2 = 0.8$ con $\sigma = 0.3$ (ver figura 2.(a)), existe un solapamiento moderado en la activación de ambas reglas. Para cumplir con el objetivo mencionado, se puede forzar a que la activación de la primera regla $\mu^1(x)$ quede limitada por la función $1 - \mu^2(x)$. En otras palabras, cuando la activación de la(s) regla(s) colindante(s) sea 1, la activación de la regla se forzará a tener valor cero. Así, el valor de activación de las reglas en este ejemplo sencillo (ver figura 2.(b)) sería:

$$\mu^{1'}(x) = \mu^1(x) (1 - \mu^2(x)) \quad (3)$$

$$\mu^{2'}(x) = \mu^2(x) (1 - \mu^1(x)) \quad (4)$$

Generalizando al caso n -dimensional, el valor de activación para la k -ésima regla se obtiene utilizando la

siguiente ecuación

$$\mu^{k'}(\vec{x}) = \mu^k(\vec{x}) \prod_{\substack{j=1; \\ j \neq k}}^K (1 - \mu^j(\vec{x})) \quad (5)$$

Quedando la expresión general para la salida del sistema difuso TSK con K reglas como

$$\begin{aligned} \tilde{y}(\vec{x}) &= \frac{\sum_{k=1}^K \mu^{k'}(\vec{x}) \tilde{y}^k}{\sum_{k=1}^K \mu^{k'}(\vec{x})} \\ &= \frac{\sum_{k=1}^K \left(\mu^k(\vec{x}) \prod_{\substack{j=1; \\ j \neq k}}^K (1 - \mu^j(\vec{x})) \right) \tilde{y}^k}{\sum_{k=1}^K \left(\mu^k(\vec{x}) \prod_{\substack{j=1; \\ j \neq k}}^K (1 - \mu^j(\vec{x})) \right)} \end{aligned} \quad (6)$$

Esta formulación para sistemas con particionamiento disperso cumple con las condiciones mencionadas. La función de salida del modelo propuesto no obstante es continua y derivable puesto que es una composición lineal de funciones continuas y derivables, con lo que se obtiene un modelado local y global simultáneo. Es fácil de deducir que, debido a la continuidad de la función Gaussiana, se cumplen las siguientes propiedades:

$$\begin{aligned} \mu^{2*}(c^1) &= 0; \mu^{1*}(c^1) = 1 \\ \Rightarrow \tilde{y}(c^1) &= \tilde{y}^1(c^1) = a^1; \\ \mu^{1*}(c^2) &= 0; \mu^{2*}(c^2) = 1 \\ \Rightarrow \tilde{y}(c^2) &= \tilde{y}^2(c^2) = a^2 \end{aligned} \quad (7)$$

Así, según la ecuación, los consecuentes $\tilde{y}^k$ de las reglas son exactamente los valores de salida del sistema difuso respecto del centro de la regla k respectiva. Estos resultados pueden extrapolarse al caso n -dimensional debido a la continuidad de las funciones compuestas.

Esta forma de inferencia garantiza por tanto que los consecuentes de las reglas son consistentes con el área del espacio de entrada que están cubriendo. De esta forma el conjunto de reglas obtenido proporciona información directa y relevante sobre el modelo construido y puede ser utilizada y modificada en su diseño directamente por los expertos como se observará en el ejemplo utilizado en este trabajo.

4 RESULTADOS

4.1 Establecimiento del problema y diseño de las reglas difusas

En el problema que nos atañe de modelado del cambio de color tras un proceso de blanqueamiento dental, el

Tabla 1: Coordenadas CIELAB para los tonos de la guía Vita Classical

Vita-Classical			
shades	L*	a*	b*
C4	34.92	7.23	12.87
A4	43.05	8.34	14.94
C3	46.29	6.78	12.88
B4	50.02	8.17	18.33
A3.5	48.94	8.49	15.70
B3	49.28	7.97	16.83
A3	56.16	7.96	14.58
D3	55.65	7.19	11.69
D4	55.57	6.18	14.40
C2	54.83	6.87	13.40
C1	55.87	5.15	8.81
A2	60.55	6.99	12.46
D2	59.41	5.59	8.59
B2	61.90	6.09	12.55
A1	63.46	5.05	9.11
B1	59.85	4.24	7.34

modelo tendrá como variables de entrada los valores CIELAB pre-blanqueamiento $\vec{x} = \{L_i^*, a_i^*, b_i^*\}$. Dado un vector de valores de entrada pre-blanqueamiento, el sistema de inferencia proporcionará una estimación de los valores CIELAB post-blanqueamiento $\tilde{\vec{y}}(\vec{x}) = \{\tilde{L}_f^*(\vec{x}), \tilde{a}_f^*(\vec{x}), \tilde{b}_f^*(\vec{x})\}$. Las reglas diseñadas para este ejemplo utilizarán como antecedentes los valores de la guía Vita-Classical, ampliamente utilizada por los odontólogos para el análisis y categorización del color de los dientes de los pacientes. Así, se definirán conjuntos difusos con función de pertenencia de tipo gaussiano para los antecedentes de las reglas, centrados en los valores CIELAB medidos para dichos tonos Vita. Los radios de dichas funciones de pertenencia se estimaron dependiendo de la cercanía del resto de reglas en el espacio tridimensional [9].

Según el enfoque de Takagi-Sugeno-Kang (TSK) [13], las reglas (una para cada tono Vita S), tendrían la siguiente forma:

$$\begin{aligned} \text{SI } L_i^* \text{ es } \mu_{L^*}^S \text{ Y } a_i^* \text{ es } \mu_{a^*}^S \text{ Y } b_i^* \text{ es } \mu_{b^*}^S \\ \text{ENTONCES} \\ L_f^* = \tilde{L}_f^{*S} \text{ Y } a_f^* = \tilde{a}_f^{*S} \text{ Y } b_f^* = \tilde{b}_f^{*S} \end{aligned} \quad (8)$$

donde $\mu_{L^*}^S$, $\mu_{a^*}^S$ y $\mu_{b^*}^S$ son funciones de pertenencia difusas centradas en los valores CIELAB para el tono Vita S para las coordenadas L^* , a^* , y b^* dadas en la tabla 1. Los valores en los consecuentes $\tilde{L}_f^{*S}$, $\tilde{a}_f^{*S}$, y $\tilde{b}_f^{*S}$ corresponden a los valores CIELAB para el tono Vita S . La activación de la regla S , $\mu^S(\vec{x})$, se obtendría utilizando el operador T-norma producto de las respectivas activaciones de los conjuntos difusos en las dimensiones L^* , a^* y b^* , $\mu^S(\vec{x}) = \mu_{L^*}^S(\vec{x}) \cdot \mu_{a^*}^S(\vec{x}) \cdot \mu_{b^*}^S(\vec{x})$

4.2 Optimización y evaluación del sistema difuso construido

Un grupo de 37 sujetos sin tinción por tetraciclina, con una media de edad de 42 años, y otro de 16 pacientes con tinción por tetraciclina con media de edad de 39 años, se involucraron en un estudio de blanqueamiento dental en el Departamento de Estomatología de la Universidad de Granada ¹. Se requirió que los pacientes tuviesen una dentadura maxilar anterior sana y sin restauraciones, caries o historia de blanqueamiento previa. En total se obtuvieron 222 mediciones de color de dientes sin tinción, y 96 con tinción por tetraciclina.

Todos los sujetos utilizaron un agente blanqueador basado en peróxido de carbamida al 20% durante dos semanas, dos horas al día. El color de los dientes de los pacientes se obtuvo utilizando un espectroradiómetro (SpectraScan PR-704, Photo Research inc., Chatsworth, USA). Estas medidas se repitieron tres veces para cada diente en su punto medio el día antes, y a los 14 días de la iniciación del tratamiento de blanqueamiento. Para asegurarse de que las mediciones se hicieran utilizando condiciones estándar de iluminación, se usó como fuente una luz Demetron Shade (KerrHawe SA, Bioggio Switzerland), que simula la irradiación relativa espectral del iluminante estándar CIE D65.

Así, se obtuvieron los valores de las coordenadas CIELAB pre- (L_i^*, a_i^*, b_i^*) y post-blanqueamiento (L_f^*, a_f^*, b_f^*) para los dientes, además de los valores L^* , a^* , y b^* de la guía Vita Classical bajo las mismas condiciones de iluminación (ver tabla 1).

La figura 3 muestra la distribución de los datos en el espacio CIELAB junto con la distribución de tonos Vita Classical en el mismo espacio ².

Para el grupo de pacientes sin tinción por tetraciclina, se seleccionó aleatoriamente los valores colorimétricos de 29 pacientes del conjunto de 37 para el diseño y optimización del sistema de reglas difusas, y las medidas de los 8 pacientes restantes se dejaron como conjunto de test para evaluar las capacidades aproximativas del modelo. Las medidas en L^* , a^* y b^* se normalizaron para dar el mismo peso a las tres coordenadas. Los tonos Vita utilizados como antecedentes de las reglas fueron aquellos que presentaban una cobertura de datos suficiente. No se encontraron suficientes datos colorimétricos de dientes asociables a los tonos B1, A1, D2, y C1 según las funciones de pertenencia

¹Tras la correspondiente aprobación del Comité de Ética de Investigación Humana de la Universidad de Granada

²Para el plano b^*L^* , una línea ficticia une los tonos Vita del C4 al B1 según el orden especificado en la tabla 1

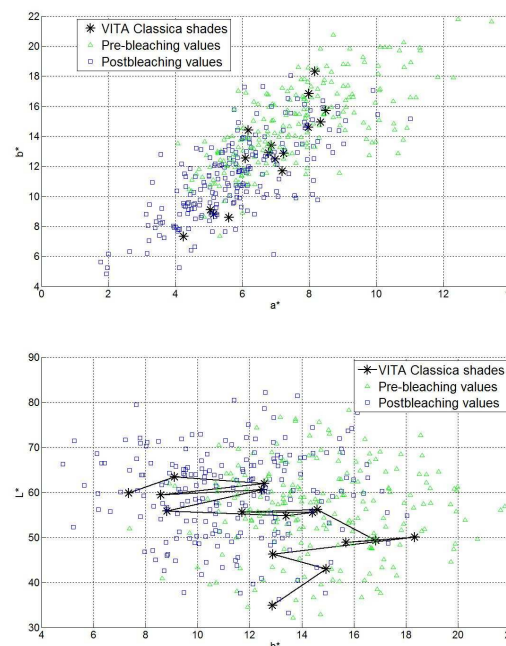


Figura 3: Distribución de los datos y de los tonos Vita Classical en el espacio CIELAB en los planos a^*b^* y b^*L^*

diseñadas. En el caso de los pacientes tetraciclínicos, se tomaron las medidas de 13 pacientes como datos de entrenamiento, dejando las medidas de 3 pacientes para la evaluación de los modelos obtenidos. Los tonos no cubiertos por datos en este caso fueron los tonos B1, A1, B2, C2, A3 y B3.

Una parte de las reglas difusas obtenidas a partir del conjunto de muestras de E/S para los 12 tonos seleccionados en el caso de los pacientes sin tinción por tetraciclina se observan en la ecuación 9; para los pacientes con tinción por tetraciclina se muestran en la ecuación 10 una parte de las 10 reglas obtenidas. En el primer caso, los valores de la Raíz del Error Cuadrático Medio (RMSE) obtenido para el sistema de reglas diseñado fue de 7.00, 1.39, y 2.22 para las coordenadas L_f^* , a_f^* , y b_f^* respectivamente, y el RMSE de test fue de 6.93, 1.41, y 2.43 respectivamente ³. Para el segundo caso los errores de entrenamiento fueron de 4.77, 1.11, y 1.56, y los de test de 5.43, 1.39, y 2.11.

En el caso de utilizar modelos lineales, esta misma subdivisión de entrenamiento y test produjo valores de RMSE de entrenamiento de 6.71, 2.17 y 1.31 y de test de 6.85, 1.45 y 2.16 para los pacientes del primer

³Sin la modificación obtenida en la ecuación 6 para el modelado local-global, los errores obtenidos fueron similares (RMSE de entrenamiento de 7.44, 1.41, y 2.28, y de test de 7.32, 1.42, y 2.40).

grupo, y de 4.95, 1.18 y 1.69 de entrenamiento y 6.48, 1.04 y 1.80 de test para los pacientes con tinción por tetraciclina. Se observa por tanto una ligera mejora del modelo neuro-difuso construido para esta aplicación, además de proporcionar una solución comprensible y diseñada para ser utilizada por expertos, al proporcionarse de forma directa el valor en CIELAB esperado para los dientes asociables a los tonos de la guía Vita Classical utilizada por los odontólogos en sus estudios.

$$\begin{aligned}
 & \text{SI } L_i^* \text{ es } \widehat{B2}_{L*} \text{ Y } a_i^* \text{ es } \widehat{B2}_{a*} \\
 & \text{Y } b_i^* \text{ es } \widehat{B2}_{b*} \text{ ENTONCES} \\
 & L_f^* = 68.89 \text{ Y } a_f^* = 4.68 \text{ Y } b_f^* = 9.49 \\
 & \text{SI } L_i^* \text{ es } \widehat{A2}_{L*} \text{ Y } a_i^* \text{ es } \widehat{A2}_{a*} \\
 & \text{Y } b_i^* \text{ es } \widehat{A2}_{b*} \text{ ENTONCES} \\
 & L_f^* = 60.39 \text{ Y } a_f^* = 4.49 \text{ Y } b_f^* = 8.15 \\
 & \dots \\
 & \text{SI } L_i^* \text{ es } \widehat{C4}_{L*} \text{ Y } a_i^* \text{ es } \widehat{C4}_{a*} \\
 & \text{Y } b_i^* \text{ es } \widehat{C4}_{b*} \text{ ENTONCES} \\
 & L_f^* = 47.32 \text{ Y } a_f^* = 5.66 \text{ Y } b_f^* = 11.78
 \end{aligned} \tag{9}$$

$$\begin{aligned}
 & \text{SI } L_i^* \text{ es } \widehat{D2}_{L*} \text{ Y } a_i^* \text{ es } \widehat{D2}_{a*} \\
 & \text{Y } b_i^* \text{ es } \widehat{D2}_{b*} \text{ ENTONCES} \\
 & L_f^* = 59.04 \text{ Y } a_f^* = 4.27 \text{ Y } b_f^* = 6.73 \\
 & \text{SI } L_i^* \text{ es } \widehat{A2}_{L*} \text{ Y } a_i^* \text{ es } \widehat{A2}_{a*} \\
 & \text{Y } b_i^* \text{ es } \widehat{A2}_{b*} \text{ ENTONCES} \\
 & L_f^* = 60.28 \text{ Y } a_f^* = 5.45 \text{ Y } b_f^* = 7.99 \\
 & \dots \\
 & \text{SI } L_i^* \text{ es } \widehat{C4}_{L*} \text{ Y } a_i^* \text{ es } \widehat{C4}_{a*} \\
 & \text{Y } b_i^* \text{ es } \widehat{C4}_{b*} \text{ ENTONCES} \\
 & L_f^* = 40.08 \text{ Y } a_f^* = 6.77 \text{ Y } b_f^* = 7.90
 \end{aligned} \tag{10}$$

5 CONCLUSIONES

Este trabajo ofrece una solución para el problema de modelado de cambio de color que proporciona modelos eficientes e interpretables. El ejemplo tratado en este trabajo opera sobre el espacio CIELAB de color, construyendo un conjunto de reglas asociadas a valores de guías de color utilizadas por los odontólogos en trabajo clínico. Gracias a una modificación del proceso de inferencia se obtienen reglas directamente interpretables y utilizables por los expertos, proporcionándose una solución efectiva y comprensible. La técnica neuro-difusa utilizada en este trabajo sería de gran interés en otras disciplinas que requieren el modelado de cambio cromático debido a cualquier fenómeno físico, químico o natural.

Agradecimientos

Este trabajo ha sido financiado por el Proyecto del CICYT TIN2007-60587 y el Proyecto P07-TIC-02768 de la Junta de Andalucía.

Referencias

- [1] R. Benavente, M. Vanrell, and R. Baldrich, A data set for fuzzy colour naming, *Color Res. Appl.* **31**, 48–56 (2006).
- [2] CIE, Colorimetry technical report 15 (3rd edition), Tech. rep., CIE Central Bureau, Vienna, Austria (2004).
- [3] O. Cordon and F. Herrera and F. Hoffmann and L. Magdalena, *Genetic Fuzzy Systems. Evolutionary tuning and learning of fuzzy knowledge bases* (Advances in Fuzzy Systems: Applications and Theory, World Scientific, 2001).
- [4] L. J. Herrera, H. Pomares, I. Rojas, A. Guillén, G. Rubio, and J. Urquiza, Global and local modelling in radial basis functions networks, *Lect. Notes Comp. Sci.* **5517**, 49–56 (2009).
- [5] L. Herrera, H. Pomares, I. Rojas, O. Valenzuela, and A. Prieto, Tase, a taylor series based fuzzy system model that combines interpretability and accuracy, *Fuzzy Set. Syst.* **153**, 403–427 (2005).
- [6] T. Johansen and R. Babuska, Multiobjective identification of takagi-sugeno fuzzy models, *IEEE T. Fuzzy Syst.* **11**, 847–860 (2003).
- [7] J. Kang, L. Min, Q. Luan, X. Li, and J. Liu, Novel modified fuzzy c-means algorithm with applications, *Digit. Signal Process.* **19**, 309–319 (2009).
- [8] G. Menegaz, A. L. Troter, J. Sequeira, and J. M. Boi, A discrete model for color naming, *EURASIP J. Adv. Sig. Pr.* **2007**, 1–10 (2007).
- [9] J. Moody and C. Darken, Fast learning in networks of locally-tuned processing units, *Neural Comput.* **1**, 281–294 (1989).
- [10] R. D. Paravina and G. Majkic and M.M. Pérez and S. Kiat-Amnuay, Color difference thresholds of maxillofacial skin replications, *Journal of prosthodontics*, **18(7)**, 618–625, (2009).
- [11] M.M. Pérez and A. Saleh and R. Pulgar and R. D. Paravina, Light polymerization-dependent changes in color and translucency of resin composites, *American journal of dentistry*, **22(2)**, 97–101, (2009).
- [12] H. Pomares, I. Rojas, J. González, and A. Prieto, Structure identification in complete rule-based fuzzy systems, *IEEE T. Fuzz. Syst.* **10**, 349–359 (2002).
- [13] T. Takagi and M. Sugeno, Fuzzy identification of systems and its applications to modelling and control, *IEEE T. Syst. Man. Cy.* **15**, 116–132 (1985).

Tecnología difusa aplicada a la comparación de motivos de ADN

Fernando García Alcalde¹ Carlos Cano Gutiérrez¹
Francisco J. López Domingo¹ Armando Blanco Morón¹

¹ Dept. Ciencias de la Computación e I. A., Univ. de Granada, 18071, Spain, {fernan,ccano,fjavier,armando}@decsai.ugr.es

Resumen

La identificación *de novo* de motivos de regulación (TFBSs) es un problema crucial en biología computacional. Los TFBSs se representan mediante matrices de frecuencia, y su identificación incluye la tarea de comparar entre sí secuencias candidatas a formar parte de un motivo en cuestión, así como la comparación de dichos motivos con otros motivos conocidos. Hasta la fecha no existe una propuesta difusa para dicho problema. En este trabajo proponemos el uso de medidas difusas en tareas de comparación de motivos. Investigamos el comportamiento de diferentes clases de medidas difusas clásicas basadas en teoría de conjuntos (Jaccard), proximidad (métrica de Minkowsky), y coeficiente angular (distancia de Bhattacharyya). Del mismo modo estudiamos una medida definida en el espacio polinucleótido difuso. Los resultados son excelentes para datos sintéticos y mejoran otras medidas existentes cuando se utilizan datos de motivos reales. Además, para solventar un problema común en el diseño de medidas de comparación de motivos, proponemos la ponderación de las posiciones de los motivos en función del nivel de conservación que presenten.

Palabras Clave: Motivos ADN, Medidas Difusas, TFBS.

sión de un gen se lleva a cabo en diversos pasos, siendo en la mayoría de los casos el más importante de ellos la transcripción. En células eucariotas las primeras etapas de la transcripción se controla mediante una serie de proteínas, llamadas *factores de transcripción* (TFs, del inglés *transcription factors*), que se unen a zonas específicas de la cadena de ADN llamadas *sitios de unión de los factores de transcripción* (TFBSs, del inglés *transcription factor binding sites*), produciendo mediante esta unión los patrones correctos de transcripción.

Parece ser que no existe un código de apareamiento sencillo aminoácido-nucleótido que determine la unión de la proteína a una secuencia de ADN determinada (como por ejemplo ocurre en los apareamientos entre los pares de bases en el ADN). Sin embargo, algunos tipos de interacción aminoácido-nucleótido aparecen con mucha más frecuencia que otros. De esta forma, y con el objetivo de representar las preferencias de unión de los TFs, surgieron los llamados *motivos de regulación*. Dado un TF, sus preferencias de unión se representan mediante una matriz que indica, para cada posición del sitio de unión (BS, del inglés *binding site*), la afinidad que presenta cada nucleótido. Dichas matrices se obtienen normalmente de alineamientos de los TFBSs conocidos para ese TF. Aunque se han propuesto diferentes representaciones para los motivos [7], la más extendida consiste en matrices que recogen para cada nucleótido su frecuencia de aparición en cada posición (PFMs, del inglés *position frequency matrices*), o bien matrices de pesos que proporcionan un valor ponderado de afinidad para cada posición de la unión entre el motivo y una secuencia de ADN (PWMs, del inglés *position weighted matrices*).

La identificación de los TFBSs es por tanto una tarea esencial para predecir la regulación de la transcripción. Así, durante los últimos años se han propuesto una gran cantidad de algoritmos para la búsqueda de TFBSs (ver el trabajo referenciado en [3]). Todos estos programas predicen un número muy elevado de

1. Introducción

Uno de los principales objetivos de la biología computacional consiste en descubrir cómo se regula la expresión de los genes. Es bien conocido que la expresi-

Cuadro 1: Ejemplo de PFM.

#	1	2	3
A	0,7	0,2	0
C	0,1	0	1
G	0	0	0
T	0,2	0,8	0

BSs para un TF dado (en media un BS cada 1000pb), incluyendo un alto porcentaje de falsos positivos [16]. Este problema se hace aún más crítico cuando se consideran simultáneamente varios TFs, circunstancia muy normal en este tipo de estudios. Los algoritmos existentes llevan a cabo diferentes estrategias para superar los problemas del resto de propuestas, lo que conlleva por lo general la aparición de otras limitaciones. Un enfoque bastante extendido consiste en aplicar varios algoritmos simultáneamente, proporcionando como salida un compendio obtenido de los resultados individuales de cada algoritmo. En este caso, los motivos encontrados por los distintos algoritmos pueden o bien corresponder al mismo TFBS, o a diferentes TFBSs, haciendo que el resultado de su combinación sea muy impreciso y ruidoso. Todo esto muestra la necesidad de desarrollar nuevos métodos de comparación entre motivos. La estrategia más comúnmente empleada para comparar dos motivos, consiste en considerar las columnas de las matrices como distribuciones de probabilidad. Así, diferentes técnicas estadísticas son empleadas para comprobar si las columnas pertenecen a la misma distribución o no. Las propuestas de este tipo más populares se basan en el coeficiente de correlación de Pearson (PCC) [9], en la implementación del *average log-likelihood ratio* (ALLR) [15], y en una adaptación del test chi-cuadrado de Pearson (PCST) [13].

La teoría de conjuntos difusos se ha utilizado en numerosas ocasiones en bioinformática. Durante los últimos años, han aparecido diferentes trabajos que integran soluciones difusas para resolver problemas biológicos, como por ejemplo el análisis de microarrays, localización de proteínas, estudio de genomas, etc., obteniendo todos ellos resultados prometedores [8, 4, 6]. Además, cuantificar la similitud empleando técnicas difusas es un tema crucial del razonamiento aproximado. Así, las medidas de similitud difusas se han aplicado de forma satisfactoria en diferentes áreas tales como sistemas expertos, recuperación de información o sistemas inteligentes de bases de datos [2]. En nuestro caso, los conceptos difusos son especialmente apropiados para la comparación y detección de motivos. Por ejemplo, un TF dado se une por lo general a más de un TFBS,

presentando además más afinidad por unos patrones de ADN que por otros. Así, en la representación matricial de los motivos, las preferencias de unión de cada posición (columna), pueden modelarse como grados de pertenencia a cuatro conjuntos, uno por cada nucleótido de ADN (A, C, G, T). En este trabajo presentamos un estudio de diferentes medidas de similitud difusas aplicadas a la comparación de motivos. Se analiza el comportamiento de distintos tipos de medidas clásicas aplicadas a los conjuntos difusos, medidas tales como las basadas en teoría de conjuntos (método de Jaccard [5]), basadas en proximidad (r-métrica de Minkowsky [17]), y basadas en el coeficiente angular (distancia de Bhattacharyya [1]). Del mismo modo, se considera también en este estudio una medida propuesta recientemente de disimilitud orientada a secuencias de ADN [14]. Además, se propone la ponderación de las posiciones de los motivos en función del grado de conservación como una mejora a tener en cuenta en futuros desarrollos de medidas para motivos de ADN.

2. Metodos

2.1. Otras medidas

En los últimos años se han propuesto diversas medidas para la comparación de motivos. Esta sección propone un breve repaso de las medidas más utilizadas. Las medidas se calculan una vez seleccionado el alineamiento óptimo entre dos motivos (considerando el alineamiento de las secuencias y de sus secuencias inversas y complementarias). Todas las medidas se calculan a partir de la comparación columna a columna. De aquí en adelante, supongamos que $C_1 = (A_{C_1}, C_{C_1}, G_{C_1}, T_{C_1})$ y $C_2 = (A_{C_2}, C_{C_2}, G_{C_2}, T_{C_2})$ son las dos columnas de las PFMs que van a ser comparadas, b_{C_1} y b_{C_2} ($b \in B, B = \{A, C, G, T\}$) son las probabilidades de la base b en C_1 y C_2 . $N_{b_{C_1}}$ y $N_{b_{C_2}}$ representan la frecuencia de la base b en C_1 y C_2 .

2.1.1. Coeficiente de correlación de Pearson.

En [9], Petrokovski propone el coeficiente de correlación de Pearson para la comparación de columnas de motivos:

$$PCC = \frac{\sum_{b \in B} (b_{C_1} - \overline{C_1})(b_{C_2} - \overline{C_2})}{\sqrt{\sum_{b \in B} (b_{C_1} - \overline{C_1})^2 \sum_{b \in B} (b_{C_2} - \overline{C_2})^2}}, \quad (1)$$

donde $\overline{C_1}$ y $\overline{C_2}$ son la media de las columnas C_1 y C_2 . Las correlaciones de todas las columnas se agregan mediante la media.

2.1.2. Promedio de máxima verosimilitud logarítmica

Wang and Stormo ([15]) definen el estadístico del promedio de máxima verosimilitud logarítmica (Average log-likelihood ratio -ALLR-) para efectuar la comparación de motivos por columnas. ALLR se define como:

$$ALLR = \frac{\sum_{b \in B} N_{b_{C_1}} \log \left(\frac{b_{C_2}}{p_b} \right) + \sum_{b \in B} N_{b_{C_2}} \log \left(\frac{b_{C_1}}{p_b} \right)}{\sum_{b \in B} (N_{b_{C_1}} + N_{b_{C_2}})}, \quad (2)$$

donde p_b es la probabilidad a priori para la base b . Para comparar una secuencia de varias columnas se promedia el valor individual obtenido para cada columna.

2.1.3. Test χ^2 .

Schones et al proponen en [13] el uso del test χ^2 para comparar motivos. Este test se calcula bajo la asunción de que las columnas son observaciones de la misma distribución. El p -valor se calcula del siguiente valor χ^2 con 3 grados de libertad:

$$\chi^2 = \sum_{j=C_1, C_2} \sum_{b \in B} \frac{(N_{jb}^o - N_{jb}^e)^2}{N_{jb}^e}, \quad (3)$$

donde N_{jb}^o es la frecuencia observada de la base b en la posición j , y N_{jb}^e es la frecuencia esperada de la base b en la posición j (ver [13] para más detalles). El p -valor es considerado una medida aditiva.

2.2. Medidas Difusas

Para situar este trabajo en contexto, hemos seleccionado cuatro medidas de similitud difusa de amplia difusión. Todas ellas excepto la última son medidas de comparación columna a columna, por lo que seguiremos empleando la notación anterior. La notación empleada para la última medida sigue las directrices de la Sección 2.2.4.

2.2.1. Basada en la teoría de conjuntos: coeficiente de Jaccard

El coeficiente de Jaccard es un coeficiente de similitud que no requiere parámetros [5]. El coeficiente de Jaccard para dos columnas de una PFMs se calcula como sigue:

$$S_J(C_1, C_2) = \sum_{b \in B} \frac{|b_{C_1} - b_{C_2}|}{\max(b_{C_1}, b_{C_2})}. \quad (4)$$

Para PFMs compuestas por múltiples columnas se calcula el promedio de los coeficientes de Jaccard obtenidos.

2.2.2. Basada en el coeficiente angular: distancia de Bhattacharyya

La distancia de Bhattacharyya mide el coseno del ángulo entre dos vectores cuando los valores de cada vector se normalizan como la desviación de la media de la función de pertenencia [1]. Este coseno se toma como medida de similitud. En el caso de dos columnas de PFMs:

$$S_B(C_1, C_2) = \frac{\sum_{b \in B} (b_{C_1} \cdot b_{C_2})}{\left(\sum_{b \in B} (b_{C_1})^2 \right)^{\frac{1}{2}} \cdot \left(\sum_{b \in B} (b_{C_2})^2 \right)^{\frac{1}{2}}}. \quad (5)$$

Para comparar múltiples columnas se promedian los coeficientes obtenidos para cada una de las columnas individuales.

2.2.3. Medida de proximidad: r-métrica de Minkowsky

La distancia entre la función de pertenencia parcial de los conjuntos difusos A y B en un universo finito $U = \{u_1, u_2, \dots, u_n\}$ puede ser calculada utilizando la r-métrica de Minkowsky [17]. Un conjunto difuso A se representa como un punto $(\mu_A(u_1), \dots, \mu_A(u_n))$ en el espacio n -dimensional. En nuestro caso $U = \{A, C, G, T\}$ y la distancia se calcula como:

$$d_r(C_1, C_2) = \left(\sum_{b \in B} |b_{C_1} - b_{C_2}|^r \right)^{\frac{1}{r}}, \quad r \geq 1, \quad (6)$$

con $r = 1$, d_r equivale a la distancia de Hamming; con $r = 2$, a la distancia Euclídea; y con $r = \infty$, a la conocida como *dominance metric*. En [2] puede obtenerse más información sobre esta materia. Aunque la elección del valor óptimo de r requiere de un amplio estudio, en este trabajo seleccionamos $r = 1,3$ ya que ha proporcionado los resultados más consistentes en la amplia experimentación llevada a cabo. De nuevo, en caso de comparar varias columnas se calcula el promedio de los valores obtenidos para cada columna.

2.2.4. Orientadas a las secuencias de ADN

En [14] las cadenas de polinucleótidos se transforman en conjuntos ordenados difusos para definir el denominado espacio difuso de polinucleótidos (fuzzy polynucleotide space -FPS-). En este trabajo también se propone una medida difusa para las secuencias de ADN basado en las ideas de [10]. Para calcular esta medida, los autores proponen la contravertida idea¹ de mapear las PFMs en un punto

¹Esta idea fue posteriormente discutida por Sadegh-Zadeh en [11] aunque los autores obtuvieron resultados prometedores en [14].

del espacio 12-dimensional $(0,1)^{12}$. Este mapeo convierte las matrices PFM's del Cuadro 1 en $M_T = (0,7; 0,1; 0; 0,2; 0,2; 0; 0; 0,8; 0; 0; 1; 0)$. La medida del espacio difuso de polinucleótidos (FPSM) para dos matrices PFM's se define como:

$$FPSM(M_1, M_2) = \frac{\sum_{i=1}^{12} |M_{1i} - M_{2i}|}{\sum_{i=1}^{12} \max(M_{1i}, M_{2i})}, \quad (7)$$

donde M_1 y M_2 son el mapeo de las dos matrices PFM's que se comparan. Ver [14] para más detalles.

3. Resultados y Discusion

3.1. Comparación entre medidas

Para medir la bondad de los distintos métodos propuestos en la sección anterior, comparamos los resultados obtenidos en la diferenciación de motivos. Para simular esta situación, generamos conjuntos de columnas de motivos obtenidas a partir de distribuciones diferentes. De esta forma, generamos 25 columnas *semilla* aleatorias y, correspondientemente, obtuvimos 25 conjuntos de datos derivados de ellas. Para evitar desviaciones en la generación de los datos, la distribución del contenido de la información (IC, del inglés *information content*) en las 25 semillas se mantuvo uniformemente distribuida en el intervalo $(0,01, 2)$. Para cada semilla, obtuvimos 50000 columnas derivadas de ella que consideramos como verdaderos positivos (VPs). Para cumplir con las características reales de los motivos, dichas columnas se obtuvieron muestreando su *semilla* correspondiente en una distribución de Dirichlet con una tamaño de muestra aleatorio entre 25 y 35 [13]. De manera similar, se generaron diferentes conjuntos de verdaderos negativos (VNs). Los pasos fueron los mismos a excepción de que suprimimos el muestreo de la columna aleatoria. Finalmente, medimos la efectividad de los distintos métodos en el siguiente experimento: para cada una de las 25 columnas *semilla*, calculamos la similitud de la *semilla* en cuestión y cada una de las columnas pertenecientes a su conjunto de VPs correspondiente, junto con la similitud a cada una de las columnas del conjunto de VNs. Consideramos que el método en cuestión obtuvo un acierto cuando la similitud más alta es asociada a una columna de los VPs. En la Figura 1 se muestra la curva característica operativa del receptor (ROC, del inglés *Receiver Operating Characteristic*) de cada método para un valor de la razón de falsos descubrimientos (FDR, del inglés *False Discovery Rate*) de 0,01. Todos los métodos presentan buenos resultados cuando el IC aumenta. Sin embargo, las medidas difusas obtienen mejores resultados para valores bajos

Cuadro 2: Distribución de JASPAR

Familia	Motivos	Familia	Motivos
ETS	7	TRP	5
FORKHEAD	8	HMG	6
BHLH	10	HOMEEO	8
bZIP EBP	4	NUCLEAR	8
MADS	5	bZIP CREB	4
REL	6		

de IC. Se puede observar como la medida PCC obtiene los peores resultados mientras que ALLR y χ^2 se comportan de manera inestable.

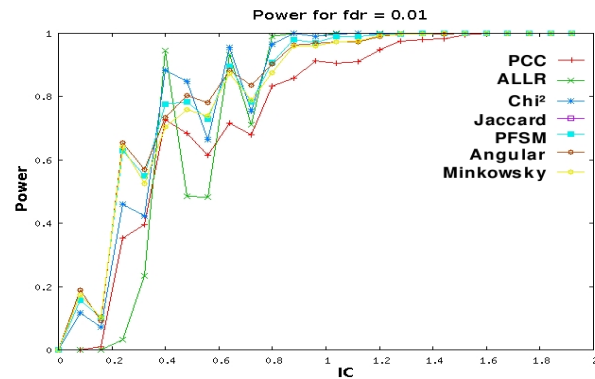


Figura 1: Curva de ROC para los distintos métodos correspondiente al reconocimiento de columnas de PFM's generadas por la misma distribución

3.2. Agrupación de motivos

Con el objetivo de comprobar la habilidad de las medidas para reconocer motivos reales relacionados, hicimos uso de la base de datos pública JASPAR [12]. JASPAR contiene 71 motivos divididos en 11 familias en función de las propiedades estructurales de los TFs (Cuadro 2). Los representantes de cada familia son conocidos como FBP's (del inglés *familial binding pro les*). Para cada familia, obtuvimos su correspondiente FBP mediante un alineamiento múltiple de los motivos de la familia en cuestión. Finalmente, para cada uno de las medidas propuestas, calculamos la similitud de cada motivo con su correspondiente FBP. En el caso de que la medida se trate de una distancia, convertimos el valor en una similitud normalizando la distancia D en $(0,1)$ para posteriormente obtener la similitud S como $S = 1 - D$. El Cuadro 3 muestra la similitud media obtenida para cada medida en cada familia de JASPAR. Se puede observar cómo las me-

Cuadro 3: Media de las distancias para los miembros de las familias JASPAR y sus FBPs

Familia	Jac.	FPSM	Mink.	Ang.	χ^2	ALLR
Difusa	Sí	Sí	Sí	Sí	No	No
ETS	0.63	0.62	0.71	0.75	0.19	0.71
Forkhead	0.46	0.45	0.55	0.02	0.01	0.05
bHLH	0.57	0.54	0.64	0.24	0.02	0.43
EBP	0.64	0.63	0.72	0.25	0.11	0.62
MADS	0.62	0.61	0.69	0.04	0.01	0.70
REL	0.70	0.68	0.76	0.27	0.19	0.77
TRP	0.50	0.49	0.58	0.26	0.03	0.17
HMG	0.55	0.54	0.65	0.90	0.05	0.50
HOME0	0.47	0.47	0.59	0.73	0.02	0.37
Nuclear	0.45	0.44	0.53	0.91	0.01	0.07
CREB	0.70	0.69	0.77	0.09	0.26	0.92
Mean	0.57	0.55	0.65	0.41	0.08	0.48

didadas Angular, PCC y ALLR proporcionan los peores resultados². Por otro lado las medidas Jaccard y FPSM obtienen resultados robustos para todas las familias. Sin embargo, la métrica de Minkowsky es la que muestra un mejor comportamiento. Usando un paired t-test, la hipótesis de que las similitudes obtenidas por la métrica de Minkowsky no son menores que aquellas obtenidas por las medidas Jaccard o FPSM se puede rechazar con un p-value de $p = 0,01$.

3.3. Ponderación de las posiciones

Uno de los principales problemas que presentan las medidas de comparación de motivos, es que no están diseñadas para tener en cuenta la diferente influencia de las posiciones según su grado de conservación. Aquellas posiciones con bajo grado de conservación no contribuyen a la unión de las proteínas a la secuencia específica de ADN. Estas posiciones deben ser por tanto menos relevantes al medir la similitud entre motivos. Por otra parte, aquellas posiciones con un alto grado de conservación son críticas para determinar la dicha similitud.

Siguiendo estas ideas, incorporamos a la medida de similitud entre motivos, no sólo las diferencias entre las posiciones de los mismos, sino también sus correspondientes grados de conservación. Así, proponemos la FWMM (Fuzzy Weighted Motif Measure), una versión

²Los resultados de la medida PCC no se muestran por motivos de espacio.

ponderada de la métrica de Minkowsky que definimos como:

$$FWMM_r(C_1, C_2) = \left(\sum_{b \in B} (|b_{C_1} - b_{C_2}| \cdot \text{máx}(b_{C_1}, b_{C_2}))^r \right)^{\frac{1}{r}} \quad (8)$$

donde $\text{máx}(b_{C_1}, b_{C_2})$ es la frecuencia de ocurrencia máxima de la base b en las columnas C_1 y C_2 . Multiplicando este factor por la diferencia entre las frecuencias observadas se obtiene el efecto deseado, es decir, ponderar cada posición de acuerdo con su especificidad: cuanto mayor sea la especificidad, mayor será la diferencia. El valor de la medida para motivos que contengan varias posiciones se obtiene calculando la media de las comparaciones columna-columna. Tal y como se describió en la sección anterior, variaciones de r dan lugar a medidas distintas formalmente equivalentes. Por claridad y consistencia, seguiremos utilizando $r = 1,3$ en el resto del trabajo, y llamaremos FWMM a $FWMM_{1,3}$ en lo que sigue.

Para demostrar la importancia de la ponderación de las posiciones de los motivos diseñamos el siguiente experimento. Definimos un motivo de referencia compuesto por tres posiciones bien conservadas y tres posiciones no conservadas. Las posiciones impares fueron generadas a partir de permutaciones del vector (10, 2, 2, 2), mientras que el vector (4, 4, 4, 4) se usó en las posiciones pares. Seguidamente examinamos el comportamiento de las medidas al comparar el motivo de referencia con otros dos motivos: i) Un motivo compuesto por seis posiciones no conservadas. Por lo tanto, dicho motivo será igual al de referencia en las posiciones pares, mientras que será muy diferente en las posiciones impares; ii) Un motivo compuesto en las posiciones impares por las mismas columnas que el motivo de referencia, mientras que en las posiciones pares nos encontramos con permutaciones del vector (7, 7, 1, 1), que difieren de aquellas en el motivo de referencia.

La Figura 2(a) muestra los logos para los tres motivos. Como se puede observar, las diferencias globales entre los dos motivos y el motivo de referencia son las mismas. Ambos son iguales en la mitad del motivo y difieren en la otra mitad. Además, estas diferencias están equilibradas en el sentido de que la diferencia entre los vectores (7, 7, 1, 1) y (4, 4, 4, 4) (diferencias con respecto al motivo conservado) es la misma que entre los vectores (4, 4, 4, 4) y (10, 2, 2, 2) (diferencias con respecto al motivo no conservado).

Finalmente, investigamos los resultados obtenidos por los diferentes métodos en dos casos: a) distancia del

motivo de referencia al motivo no conservado, y *b*) distancia del motivo de referencia al motivo conservado. De acuerdo con lo explicado arriba, se esperaba que el motivo conservado estuviera más cercano al motivo de referencia que el no conservado. Comprobamos los resultados de cada método calculando el cociente de la distancia para el caso *a*) y el caso *b*). En la Figura 2(b) se observa cómo la medida difusa ponderada se comporta de una manera mucho mejor que el resto de los métodos, mientras que tres de las medidas (χ^2 , ALLR, y PCC) obtuvieron el resultado erróneo de asignar una distancia menor para el caso *a*).

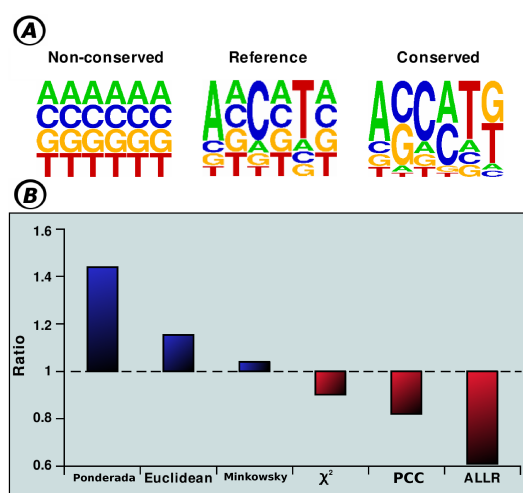


Figura 2: A) Representación de los motivos. B) Ratio de las distancias para cada medida.

Agradecimientos

Este trabajo se ha llevado a cabo como parte de los proyectos P08-TIC-4299 J.A., Sevilla y TIN2009-13489 DGICT, Madrid.

Referencias

- [1] Bhattacharya, A. On a measure of divergence of two multinomial populations, *Sankhya*, 7, pp. 401-406, 1946.
- [2] Cross V. and Sudkamp, T. Similarity and Compatibility in Fuzzy Set Theory: Assessment and Applications, *Springer*, 2002.
- [3] Das, M. K. and Dai, H. A survey of DNA motifs finding algorithms, *BMC Bioinformatics*, 8(Suppl 7), S21, 2007.
- [4] Huang, Y. and Li, Y. Prediction of protein sub-cellular locations using fuzzy k-NN method, *Bioinformatics*, 20, pp. 21-28, 2004.
- [5] Jaccard, P. Nouvelles recherches sur la distribution florale, *Bulletin de la Societe de Vaud des Sciences Naturelles*, 44, pp. 223, 1908.
- [6] Lopez, F.J. et al. Fuzzy association rules for biological data analysis: a case study on yeast, *BMC Bioinformatics*, 1:9, 2008.
- [7] Osada, R. et al. Comparative analysis of methods for representing and searching for transcription factor binding sites, *Bioinformatics*, 20, pp. 3516-3525, 2004.
- [8] Pan, Y. Advances in the Discovery of cis-Regulatory Elements, *Current Bioinformatics*, 1, pp. 321-336, 2006.
- [9] Petrokovski, S. Searching databases of conserved sequence regions by aligning protein multiple-alignments, *Nucleic Acids Res.* 24, pp. 3836-3845, 1996.
- [10] Sadegh-Zadeh, K. Fuzzy genomes, *Artif. Intell. Med.*, 18, pp. 1-28, 2000.
- [11] Sadegh-Zadeh, K. The fuzzy polynucleotide space revisited, *Artif. Intell. Med.*, 41, pp. 69-80, 2007.
- [12] Sandelin, A. et al. JASPAR: An open-access database for eukaryotic transcription factor binding profiles, *Nucleic Acids Res.*, 32: D91-D94, 2004.
- [13] Schones, D. E. et al. Similarity of position frequency matrices for transcription factor binding sites, *Bioinformatics*, vol. 21, pp. 3073-313, 2005.
- [14] Torres, A. and Nieto, J.J. The fuzzy polynucleotide space: basic properties, *Bioinformatics*, 5, pp. 587-592, 2003.
- [15] Wang, T. and Stormo, G. D. Combining phylogenetic data with co-regulated genes to identify regulatory motifs, *Bioinformatics*, 19, pp. 2369-2380, 2003.
- [16] Wasserman, W. and Sandelin, A. Applied bioinformatics for the identification of regulatory elements, *Nat Rev Genet*, 5:4, pp. 276-287, 2004.
- [17] Zwick, R. et al. Measures of similarity among fuzzy concepts: a comparative analysis, *International Journal of Approximate Reasoning*, 1, pp. 221-242, 1987.

Importancia de las Propiedades Físico-Químicas de los Aminoácidos en la Predicción de Estructuras de Proteínas usando Vecinos más Cercanos

Gualberto Asencio Cortés¹ Jesús S. Aguilar-Ruiz¹

¹ Universidad Pablo de Olavide, {guaasecor,aguilar}@upo.es

Resumen

La predicción de estructuras de proteínas es actualmente un importante campo de investigación dentro de la bioinformática. En esta área, existen numerosos estudios realizados en los que se ha usado la información de la separación entre los aminoácidos de una cadena para predecir la estructura de las proteínas, utilizándose en otros trabajos ciertas propiedades físico-químicas de aminoácidos. En este trabajo se han usado ambas informaciones y se ha estudiado cómo influyen en la predicción de estructuras de proteínas empleando el algoritmo de vecinos más cercanos. Hemos comprobado que la información proporcionada por las propiedades físico-químicas es de mayor interés que la separación, obteniéndose mejores tasas de acierto. Se han realizado cuatro experimentos en los que se ha usado como atributos, la separación entre aminoácidos y un conjunto determinado de propiedades físico-químicas de los mismos y, como ejemplos, todas las subsecuencias posibles encontradas en un conjunto de más de 5000 proteínas reales. Finalmente se demuestra empíricamente que la separación entre aminoácidos, ampliamente usada en la literatura, puede ser reemplazada por propiedades físico-químicas de aminoácidos, produciendo mejores predicciones. La tasa de acierto conseguida usando sólo la separación está en torno al 59%, ascendiendo este valor hasta el 79% al usarse un conjunto de propiedades físico-químicas de aminoácidos.

Palabras Clave: propiedades físico-químicas, separación, vecinos más cercanos, predicción de estructura secundaria de proteínas.

1 INTRODUCCIÓN

Este trabajo se encuentra emplazado dentro del área de la bioinformática y, dentro de la misma, en la predicción de estructuras secundarias de proteínas.

La predicción de la estructura secundaria de las proteínas consiste en averiguar un patrón de comportamiento en la formación tridimensional de las proteínas, únicamente a partir de la cadena de aminoácidos que la forman.

Hoy en día se desconoce el carácter determinista de dicho proceso de formación, estudiándose tan solo los estados inicial (cadena de aminoácidos) y final (proteína totalmente formada).

Para llevar a cabo la tarea de predicción se pueden utilizar métodos de minería de datos, los cuales trabajan, en este contexto, con bases de datos procedentes de proteínas conocidas, y mediante algoritmos de clasificación y regresión extraen el conocimiento suficiente para generar un modelo que permita averiguar de forma determinista la estructura que tendrá una proteína aún sin formarse, con un determinado grado de probabilidad de acierto. Entre estos métodos se encuentra la técnica de vecinos más cercanos, empleada en la predicción de estructuras de proteínas [4,13].

Para realizar las predicciones se pueden usar los llamados mapas de contacto, que ilustran dónde se producen las conexiones entre los aminoácidos dentro de la cadena proteica, y los llamados mapas de distancia, entre otros. En estos últimos, se representan los valores reales y predichos de distancia entre dos aminoácidos cualesquiera de la cadena. En este trabajo se predicen mapas de distancia, con lo que se hablará de regresión en lugar de clasificación, al ser la distancia un valor continuo. De esta forma, los resultados de las predicciones realizadas pueden ser expresados a posteriori en términos de contactos usando diferentes umbrales.

En numerosos estudios realizados se ha usado para la predicción la información de la separación entre

aminoácidos en las cadenas proteicas [2,3,11]. Esta separación mide el número de aminoácidos intermedios entre dos determinados en dichas cadenas. Parece claro que esta información puede ser útil para la predicción de la estructura de una proteína puesto que revela la posición relativa de los aminoácidos dentro de la cadena, y son éstos los que establecen contactos unos con otros. Además, se ha comprobado que dicha información es útil debido a que existen numerosos contactos entre aminoácidos bien distanciados en la cadena (long-range interactions) [7,12], siendo éstos ignorados en métodos de predicción locales.

Por otra parte, analicemos la naturaleza del principal fenómeno que acontece en la formación de la estructura de una proteína: la formación de un contacto entre aminoácidos. Éste es un proceso biológico donde intervienen una serie de elementos materiales, tales como los propios aminoácidos, con una serie de propiedades físico-químicas inherentes. Para determinar el lugar donde se producen estos contactos, creemos que es necesario observar los valores de dichas propiedades físico-químicas y usarlas en la predicción de la estructura de las proteínas. Existen trabajos en los que se han usado este tipo de propiedades físico-químicas [8,14] aunque son empleadas, entre otros fines, para determinar dominios de proteínas e interfaces proteína-proteína.

En consecuencia, nos planteamos predecir con ambas medidas: la separación y las propiedades físico-químicas; para luego omitir cada una y comprobar el comportamiento ofrecido tanto en conjunto como aisladamente. El objetivo es determinar cómo influyen las propiedades físico-químicas de los aminoácidos en la predicción de estructuras de proteínas, en las cuales confiamos que existe información útil.

2 MÉTODO

Se han realizado tres actividades encaminadas a probar un conjunto de datos de proteínas reales, una serie de propiedades físico-químicas conocidas y la separación entre aminoácidos, mediante un algoritmo de vecinos más cercanos, para finalmente analizar los resultados obtenidos y extraer conclusiones en relación al objetivo planteado.

Se ha escogido la herramienta Weka [5], ampliamente utilizada en procesos de minería de datos, para predecir la estructura de las proteínas. En concreto, para predecir la distancia entre cualquier par de aminoácidos de sus cadenas polipeptídicas.

Esta herramienta trabaja con archivos de datos donde deben incluirse una serie de ejemplos con un conjunto de atributos determinado y una clase o etiqueta, para la cual

se realizan las predicciones. Los archivos de datos suministrados a Weka en los experimentos realizados en este trabajo contienen los valores de separación entre aminoácidos y de determinadas propiedades físico-químicas para todas las posibles subsecuencias encontradas en un conjunto de proteínas procedentes del Protein Data Bank (PDB) [1].

En concreto, para los experimentos realizados se han tomado todas las proteínas existentes del PDB de una sola cadena polipeptídica y de longitud máxima 100; lo cual ha dado de resultado un total de 5380 proteínas (en la fecha en la que se tomaron los datos) que en adelante denominaremos set5380p. En la figura 1 se puede apreciar el procedimiento global empleado.

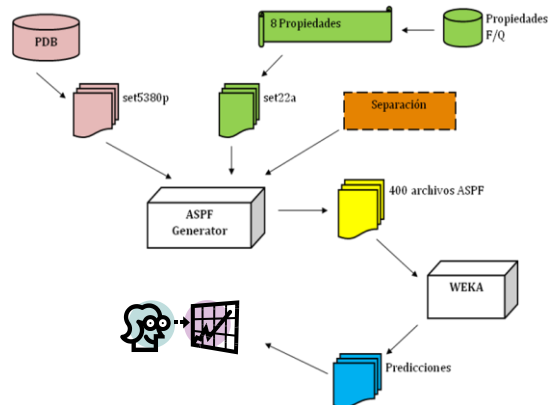


Figura 1: Proceso global empleado para obtener los resultados

2.1. GENERACIÓN DE FICHEROS DE PROPIEDADES DE SUBSECUENCIAS DE AMINOÁCIDOS (ASPF)

Para cada proteína analizada, se han considerado todos los pares posibles de aminoácidos naturales ($20 \times 20 = 400$ pares posibles: A-A, A-C, ...) y se han obtenido todas las subsecuencias incluidas en la proteína, clasificándolas según los aminoácidos de inicio y fin de cada una. Se han ignorado las subsecuencias con separación 0 entre dichos aminoácidos. Por ejemplo: si la cadena de aminoácidos es CACAC, el número de subsecuencias obtenidas para cada par de aminoácidos inicio-fin serían: A-A:1, A-C:1, C-A:1, C-C:2.

Todas las subsecuencias de todas las proteínas analizadas han sido organizadas en $20 \times 20 = 400$ ficheros de datos. Cada fichero contiene un resumen de los valores de las propiedades físico-químicas de los aminoácidos interiores, omitiendo los extremos, en todas las subsecuencias con unos aminoácidos inicio y fin determinados.

Estos ficheros reciben el nombre de Ficheros de Propiedades de Subsecuencias de Aminoácidos (ASPF). Así, por ejemplo, el fichero “A-C.aspf” contendrá un resumen de las propiedades físico-químicas de los aminoácidos que constituyen cada una de las subsecuencias con inicio en el aminoácido A y fin en el aminoácido C encontradas en todo el set de proteínas estudiado.

Se ha usado un conjunto de 8 propiedades físico-químicas descritas y evaluadas en [9], de las cuales, se han considerado siete propiedades nominales y una real. En la figura 2 se muestra un diagrama de Venn en el que se recogen varias de las propiedades usadas (hydrophobic, aliphatic, aromatic, polar, charge, small). Además, se han usado las propiedades cbetabranched y functional (numérica real), no descritas en la figura 2.

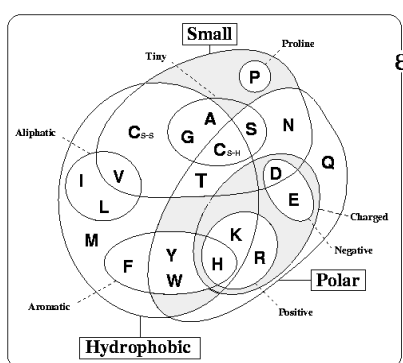


Figura 2: Diagrama de Venn de algunas propiedades físico-químicas de aminoácidos

Clasificamos los 20 aminoácidos naturales en función del grado que ofrecen para cada una de estas propiedades. Por ejemplo, para la propiedad charge, existen cuatro grados: usualmente positivo, usualmente negativo, a veces positivo y sin carga. Pueden consultarse los grados de todas las propiedades usadas en [9].

Dado que, en general, no existe ninguna relación de orden entre estos grados, se han creado 21 atributos, uno por cada grado de las 7 propiedades nominales. Cada uno de estos 21 atributos es un número real y representa el tanto por 1 de ocurrencia (en las subsecuencias) de aminoácidos que presentan cada grado de propiedad físico-química. En definitiva, se han usado 21+1 (la propiedad funcional, que es numérica real)=22 atributos físico-químicos, que denominaremos set22a, como se resume en la figura 1.

Existen numerosas propiedades físico-químicas de aminoácidos publicadas que no se han usado [6,10,15], y que nos proponemos utilizar en futuros trabajos.

Se ha implementado una aplicación denominada ASPFGenerator que permite, a partir de archivos .ENT de datos de proteínas descargados desde el PDB, generar

automáticamente los 400 ficheros ASPF descritos. Esta herramienta ha sido usada para procesar 5 GB de archivos de texto .ENT correspondientes al conjunto set5380p de estudio, generando 1,80 GB en 400 ficheros ASPF, empleando 180 hilos de ejecución y 5 minutos en el proceso.

2.2. PREDICCIÓN CON WEKA (IBK)

Una vez obtenidos los archivos de propiedades ASPF, éstos serán la entrada de la herramienta Weka para predecir las distancias entre los aminoácidos. En este trabajo se ha usado Weka 3.6.0 Experimenter y el clasificador IBk con K=1 (1-KNN).

Dado que los datos de entrada para la predicción con Weka son los archivos ASPF antes generados, se predecirá la distancia entre dos aminoácidos cualesquiera a partir del conocimiento obtenido en las propiedades físico-químicas y los valores de separación de todas las subsecuencias con origen y fin en dichos aminoácidos. De este modo, el conocimiento está basado en similitudes de naturaleza físico-química entre los aminoácidos intermedios en las cadenas polipeptídicas.

Para comprobar las predicciones resultantes y descubrir conocimiento acerca de la influencia de la separación entre aminoácidos frente a sus propiedades físico-químicas, se han realizado con Weka cuatro experimentos:

Experimento SOLO_SEP: En este experimento se ha usado únicamente el atributo separación en la obtención de vecinos más cercanos. Este experimento ha servido para sentar las bases a partir de las cuales intentar mejorar las predicciones.

Experimento SOLO_PFQ: Aquí se ha usado únicamente el set22a, antes descrito, de propiedades físico-químicas sin atributo separación en la obtención de vecinos más cercanos. Este experimento ha sido realizado para demostrar la validez de estas propiedades, aisladamente, en la predicción de estructuras de proteínas.

Experimento IGUAL_SEP: Aquí se ha utilizado como atributos el set22a, junto al atributo separación, teniendo en cuenta que se ha reducido el campo de exploración de vecinos más cercanos (subsecuencias más cercanas) a sólo aquellos con igual valor de separación que el ejemplo de test a clasificar. Se ha realizado este experimento para determinar cómo se comportan las predicciones cuando asumimos que las distancias entre aminoácidos son similares si éstos tienen la misma separación y similares propiedades.

Experimento CUALQ_SEP: En este experimento se han usado los mismos atributos que en el experimento IGUAL_SEP pero explorando entre todas las

subsecuencias disponibles. Se pretende probar la predicción de estructuras de proteínas usando la separación y las propiedades físico-químicas sin ninguna restricción.

Para la predicción en los cuatro experimentos se han usado todas las subsecuencias encontradas cuyo aminoácido de inicio es A, C, D, E, F, G, H, I, K o L.

Los atributos que se han usado en dicha distancia están todos normalizados entre 0 y 1. La distancia usada ha sido la distancia euclídea sin ponderación alguna de atributos:

$$dij = \sqrt{\underbrace{(S_i - S_j)^2}_{dSEP} + \sum_f \underbrace{(P_{fi} - P_{fj})^2}_{dPFQ}} \quad (1)$$

De ésta se deducen fácilmente los siguientes casos particulares: en el experimento IGUAL_SEP la componente dSEP = 0, en el experimento SOLO_PFQ la componente dSEP no existe, y en el experimento SOLO_SEP la componente dPFQ no existe.

Para obtener resultados adecuados al objetivo perseguido en el estudio, se han introducido algunas modificaciones al código fuente original de la herramienta Weka, de tal forma que permita obtener el error relativo cometido en las predicciones. No obstante, el comportamiento del clasificador IBk de Weka no ha sido alterado, usando para todos los experimentos K=1 (1-KNN).

2.3. ANÁLISIS DE RESULTADOS

Finalmente, una vez obtenidos los resultados de la predicción con Weka, se han analizado con Excel para descubrir, a partir de los resultados, el comportamiento de las predicciones; ilustrando mediante tablas y gráficas, las cuales se verán en el siguiente apartado, la información pertinente para justificar las conclusiones.

3 EXPERIMENTACIÓN

A continuación, se mostrarán las tablas y gráficas generadas a partir de los resultados obtenidos en la experimentación y se extraerán las conclusiones de las mismas en el marco de los objetivos planteados en este trabajo.

3.1 TASA DE ERRORES EN LOS EXPERIMENTOS

En primer lugar mostramos en la figura 3, mediante un gráfico de Boxplot, el error relativo medio cometido en cada experimento.

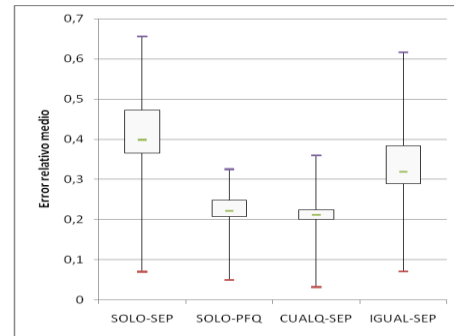


Figura 3: Gráfico Boxplot del error relativo cometido en los experimentos

Según se aprecia en la figura 3, se consigue menor tasa de errores (aprox. 0,216) cuando interviene, junto a la separación, la información de las propiedades físico-químicas de los aminoácidos en la búsqueda de vecinos más cercanos (CUALQ_SEP).

No obstante, es de interés destacar que la tasa de errores (0,227) y la desviación típica (0,272) obtenidas usando únicamente las propiedades físico-químicas (SOLO_PFQ) es similar a la conseguida en el experimento CUALQ_SEP; incluso se consigue una desviación estándar menor. Esto indica el buen comportamiento de dichas propiedades, por sí solas, a la hora de determinar las distancias entre aminoácidos.

Nótese, por otra parte, cómo el experimento IGUAL_SEP no ha arrojado tan buenos resultados como CUALQ_SEP, usando ambos la separación y las propiedades. Esto se debe a que las distancias entre aminoácidos no son siempre similares teniendo éstos la misma separación.

3.2 ANÁLISIS PORMENORIZADO DE ERRORES PARA CADA VALOR DE SEPARACIÓN

A continuación analizamos en detalle el error relativo cometido en las predicciones en función de la separación existente en las subsecuencias de proteínas estudiadas. Para cada experimento, estudiamos mediante una gráfica estos valores de error. En todas las gráficas que siguen se representa en el eje de abscisas la separación entre los aminoácidos para los cuales se ha predicho su distancia, y en el eje de ordenadas los valores de error relativo promedio cometido para cada par de aminoácidos estudiado.

En primer lugar, se presenta en la figura 4 la gráfica de errores para el experimento SOLO_SEP, donde se ha usado únicamente el atributo separación en la obtención de vecinos más cercanos.

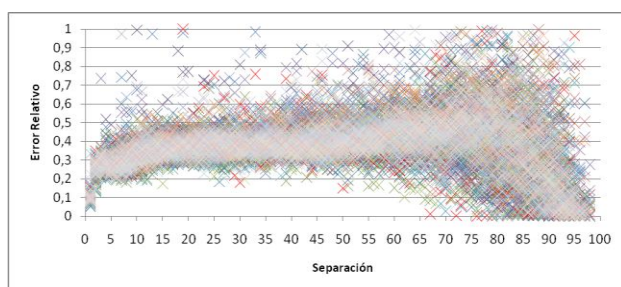


Figura 4: Gráfica de errores para el experimento SOLO_SEP

Como se puede apreciar en la figura 4, el error relativo medio aumenta conforme lo hace la separación entre aminoácidos, situándose en valores medios en torno a 0,35 y 0,4. Existe en este experimento una gran dispersión de valores de error para valores altos de separación, lo cual no es deseable. No obstante, tal como ocurrirá en el resto de experimentos, existe una tendencia clara al descenso del error relativo para valores altos de separación. En este caso, a partir del valor 80 de separación aproximadamente, los errores se hacen cada vez más cercanos a 0; realizándose perfectas predicciones en la distancia entre aminoácidos con alta separación.

Además, podemos apreciar tanto en la figura 4 como en las correspondientes al resto de experimentos, que para valores de separación pequeños (1,2) el error relativo cometido es siempre menor (entre los valores 0,05 y 0,3). Esto se debe a que la distancia entre aminoácidos próximos en la cadena proteica (separación pequeña) suele ser siempre la misma, debido a la cercanía de éstos en la cadena proteica y no debido a la formación de un contacto. De este modo, se encuentran vecinos con misma separación y distancia muy parecida y el error cometido desciende.

Mostramos a continuación en la figura 5 la gráfica de errores para el experimento SOLO_PFQ, donde se han usado únicamente las propiedades físico-químicas sin atributo separación en la obtención de vecinos más cercanos.

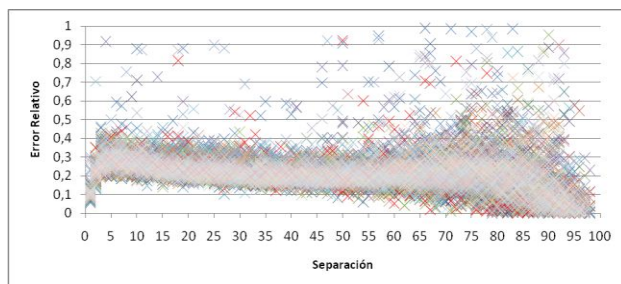


Figura 5: Gráfica de errores para el experimento SOLO_PFQ

La mejora conseguida es apreciable en varios sentidos. En primer lugar, el error relativo adopta mejores valores, situándose en valores medios en torno a 0,2 y 0,25. Por otra parte, el error relativo desciende conforme aumenta la separación, al contrario que en el experimento anterior, lo cual es altamente deseable. Además, se conserva la clara tendencia al descenso del error relativo para valores altos de separación, con la gran ventaja añadida de una considerable menor dispersión de valores de error.

A continuación se muestra en la figura 6 la gráfica de errores para el experimento CUALQ_SEP, donde se han utilizado tanto la separación como las propiedades de aminoácidos, usando en la exploración todos los ejemplos (subsecuencias de proteínas) disponibles en la base de datos. Habiendo observado los resultados conseguidos únicamente con las propiedades físico-químicas, comprobamos ahora cómo interviene la separación cuando ésta se incorpora en las predicciones.

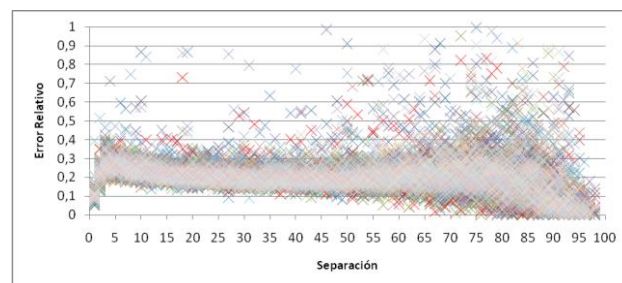


Figura 6: Gráfica de errores para el experimento CUALQ_SEP

Comprobamos que el comportamiento del error relativo es prácticamente el mismo que en el experimento SOLO_PFQ (véanse las figuras 5 y 6). Recordemos que el error relativo medio obtenido en la figura 3 era muy parecido en ambos experimentos (SOLO_PFQ y CUALQ_SEP). Además, se obtuvo un valor menor de desviación típica en el experimento SOLO_PFQ. Con ello, parece deseable usar de cualquier modo las propiedades físico-químicas, ya sea en presencia o en ausencia de la información de la separación entre aminoácidos.

En la figura 7 se presenta la gráfica de errores para el experimento IGUAL_SEP, donde se ha utilizado tanto la separación como las propiedades de aminoácidos, teniendo en cuenta que se ha reducido el campo de exploración de vecinos más cercanos a sólo aquellos con igual valor de separación que el ejemplo de test.

En este caso, como se comentó anteriormente, la componente dSEP de la distancia es siempre 0, puesto que el espacio de exploración de vecinos está formado únicamente por ejemplos con igual valor de separación que la instancia de test. Con ello, se reduce dicho espacio de exploración con respecto al total disponible y se

pueden dejar de considerar subsecuencias que pueden ser mejores, en el sentido de proporcionar mejor predicción.

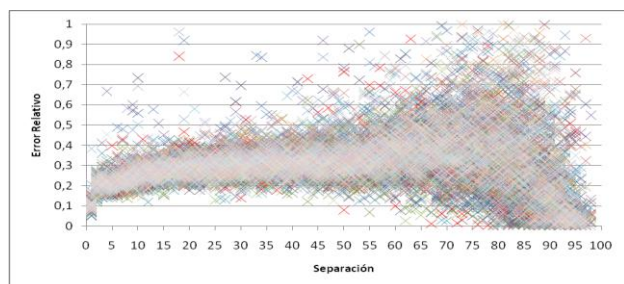


Figura 7: Gráfica de errores para el experimento IGUAL_SEP

Nótese que la gráfica de la figura 7 es similar a la de la figura 4 (experimento SOLO_SEP), salvo que, mientras en esta última la banda de errores se sitúa, en términos medios, por encima del valor 0,3, en este experimento IGUAL_SEP (donde se introducen las propiedades físico-químicas) se sitúa por encima del valor 0,2. Esto está en consonancia con el menor error relativo medio del experimento IGUAL_SEP, como revela la figura 3.

4 CONCLUSIONES

Las propiedades físico-químicas de los aminoácidos repercuten positivamente en la tasa de acierto de la predicción de estructuras proteínas mediante un clásico algoritmo de vecinos más cercanos, ya sea en presencia o en ausencia de la información de separación entre aminoácidos.

Según los resultados obtenidos en este trabajo, parece conveniente explorar más propiedades físico-químicas y añadirlas al conjunto de estudio, para determinar cuál es el conjunto o conjuntos de propiedades que mejor ayudan a la predicción de estructuras de proteínas.

Los resultados nos conducen a pensar que algún subconjunto de todas las posibles propiedades físico-químicas de aminoácidos conocidas, con una asignación oportuna de ponderaciones, permitiría realizar predicciones de mayor calidad y minimizar el error.

Referencias

- [1] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E. The Protein Data Bank. *Nucleic Acids Research*, 28 pp. 235-242, 2000.
- [2] Fariselli, P., Casadio, R. A neural network based predictor of residue contacts in proteins. *Protein Engineering* vol.12 no.1 pp.15-21, 1999.
- [3] Fariselli, P., Olmea, O., Valencia, A., Casadio R. Prediction of contact maps with neural networks and correlated mutations. *Protein engineering* vol.14 no.11 pp. 835-843, 2001.
- [4] Frishman, D., Argos, P. Incorporation of non-local interactions in protein secondary structure prediction from the amino acid sequence. *Protein Engineering* 9, 133-142, 1996.
- [5] Ian H. Witten and Eibe Frank. Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.
- [6] Kawashima, S. and Kanehisa, M. AAindex: Amino acid index database. *Nucleic Acids Res.* 28, 374, 2000. <http://www.genome.jp/aaindex/>
- [7] Kihara, D., The effect of long-range interactions on the secondary structure formation of proteins. *Protein Sci.*, 14:1955-1963, 2005.
- [8] Ladunga, I. PHYSEAN PHYsical SEquence ANALYSIS for the identification of protein domains on the basis of physical and chemical properties of amino acids. *Bioinformatics* vol.15 no.12 pp. 1028-1038, 1999.
- [9] M.J. Betts, R.B. Russell. Amino acid properties and consequences of substitutions. In *Bioinformatics for Geneticists*, M.R. Barnes, I.C. Gray eds, Wiley, 2003. <http://www.russell.embl-heidelberg.de/aas/>
- [10] Nakai, K., Kidera, A., Kanehisa, M. Cluster analysis of amino acid indices for prediction of protein structure and function. *Protein Eng.* 2, 93-100, 1988.
- [11] Olmea, O., Valencia, A. Improving contact predictions by the combination of correlated mutations and other sources of sequence information. *Folding & Design* 2, s25-s32, 1997.
- [12] Pan, X.M., Niu, W.D., Wang, Z.X. What is the minimum number of residues to determine the secondary structural state? *J. Protein Chem.* 18:579-584, 1999.
- [13] Salamov, A.A., Solovyev, V.V. Protein Secondary Structure Prediction Using Local Alignments. *J. Mol. Biol.*, 268,31-36, 1997.
- [14] Surendra, S. Negi, Werner Braun. Statistical analysis of physical-chemical properties and prediction of protein-protein interfaces. *J Mol Model.* November 2007; 13(11): 1157-1167. doi:10.1007/s00894-007-0237-0.
- [15] Tomii, K., Kanehisa, M. Analysis of amino acid indices and mutation matrices for sequence comparison and structure prediction of proteins. *Protein Eng.* 9, 27-36, 1996.

Definición de Umbral Mínimo para la Predicción de Estructura Secundaria de Proteínas

Alfonso E. Márquez Chamorro¹, Jesús S. Aguilar Ruiz¹ y Eloy Anguiano Rey²

¹ Departamento de lenguajes y sistemas, Universidad Pablo de Olavide, {amarcha,aguilar}@upo.es

² Departamento de ingeniería informática, Universidad Autónoma de Madrid, eloy.anguiano@uam.es

Resumen

Múltiples enfoques se han desarrollado para la predicción de la estructura secundaria de la proteína. Cada investigador toma diferentes ejemplos y bases de datos en sus experimentaciones, por tanto no existe una manera estándar para conocer la validez de unos nuevos resultados obtenidos. Se propone establecer un umbral mínimo de precisión en la predicción, calculado con un algoritmo simple de matrices de distancia y con el conjunto total de proteínas almacenadas en el Protein Data Bank. Para la verificación de los resultados se ha realizado un análisis exhaustivo de las medidas estadísticas de Sensibilidad, Especificidad y Precisión. Estos valores suponen una medida de referencia para la comunidad científica en este campo de investigación.

Palabras Clave: Mapas de contacto, Predicción de estructura secundaria de proteínas, Matrices de distancia.

1 Introducción

La bioinformática ha sido descrita como la ciencia del tratamiento, interpretación y minería de la información de secuencias y estructuras biológicas. Se consideran dos importantes campos en Bioinformática: Genómica y Proteómica. La Proteómica se define como la caracterización y la identificación de las proteínas codificadas en el genoma de un organismo. Hoy en día, la predicción de la estructura de las proteínas es uno de los desafíos más importantes en Proteómica. Desde que la comunidad científica desarrollara la secuenciación del genoma humano, el

siguiente paso es la localización y caracterización de genes y la identificación de las estructuras primaria y terciaria de las proteínas codificados por ellos. Se sabe que la estructura tridimensional de las proteínas está codificada por su cadena de aminoácidos, pero aún no hemos sido capaces de descifrar el código para obtener más información sobre la estructura, la función y los patrones de interacción de una proteína.

Las proteínas son uno de los componentes básicos en todos los organismos. Forman la base de la vida celular y molecular y afectan a las características estructurales y funcionales de las distintas células y genes significativamente. En el primer nivel, las proteínas se componen de secuencias lineales de aminoácidos naturales unidos por enlaces peptídicos. Esto se conoce como la estructura primaria de la proteína. Esta secuencia determina las estructuras formadas en los siguientes niveles. El cambio en un solo aminoácido en un área crítica de la proteína puede alterar la función biológica. La estructura secundaria de una proteína es el plegamiento de la cadena polipeptídica. En este nivel aparecen algunas estructuras denominadas α -hélices, láminas- β , bucles y giros. La estructura terciaria es la forma tridimensional de dicha cadena. La estructura cuaternaria es la estructura final tridimensional formada por todas las cadenas polipeptídicas que componen una proteína [10, 1].

El problema de predicción de la estructura secundaria de proteínas, consiste en predecir la localización de hélices alfa, láminas beta y giros en una secuencia de aminoácidos, en ausencia de cualquier conocimiento de la estructura terciaria de la proteína. La precisión en la predicción de la estructura secundaria es de gran interés, puesto que el conocimiento de la ubicación de los elementos de la estructura secundaria pueden ser utilizados por los algoritmos de aproximación de plegado para obtener la estructura terciaria de la proteína. Para comprender las funciones de las proteínas a nivel molecular, es necesario determinar

la estructura tridimensional de las proteínas.

Varios métodos se utilizan para predecir la estructura secundaria. Estos métodos pueden clasificarse en dos categorías: métodos estadísticos [3, 9, 15] y técnicas de Soft Computing. Soft Computing proporciona capacidad de procesamiento con el fin de resolver el problema de la predicción de estructura de proteínas. Las principales técnicas empleadas son: redes neuronales artificiales (RNA) [20, 7, 21, 18, 14, 4, 6, 16], métodos de vecinos más cercanos [8, 22, 24], support vector machines (SVMs) [13, 23, 2, 11] y computación evolutiva [25, 19, 5, 12].

Los métodos de investigación utilizados para resolver el problema de la predicción de estructura secundaria de proteínas se centran en la determinación de mapas de contacto (distancias) entre los aminoácidos de una secuencia de una proteína. Cuando un mapa de contacto se define, se puede obtener la estructura terciaria de la proteína.

Sin embargo no existe una medida estándar de referencia en la predicción para conocer si un nuevo resultado es válido. Cada investigador toma diferentes tamaños de bases de datos en sus experimentaciones. Por ello, con este artículo se pretende establecer unos valores de referencia mínimos para la predicción, que puedan ser útiles a la comunidad científica como comparativa a unos nuevos resultados obtenidos.

El artículo está dividido en varias secciones. La sección Metodología explica el proceso experimental llevado a cabo, donde cada subsección representa una de las etapas de dicho proceso. En la sección Resultados se expone la información obtenida de la experimentación y se analizan los resultados. En el último apartado se deducen las conclusiones.

2 Metodología

Se especifica a continuación la metodología llevada a cabo en este método para la predicción de estructura secundaria. En primer lugar, se obtienen los datos de las proteínas. Con ellos se entrena el algoritmo, para la posterior predicción de las distancias entre aminoácidos. Para comprobar la efectividad de nuestro algoritmo, se comparan las distancias estimadas con las distancias reales. Como último paso se procedería a la reconstrucción de la proteína. A partir de su estructura primaria, el algoritmo ha de ser capaz de predecir la estructura secundaria para una nueva proteína (Figura 1).



Figura 1: Procedimiento experimental.

2.1 Adquisición de datos

La estructura secundaria de las proteínas se obtiene a partir de su secuencia de aminoácidos, así como las distancias entre cada par de dichos aminoácidos. Toda esa información está incluida en PDB (Protein Data Bank). La Worldwide Protein Data Bank [26], es una colaboración internacional que organiza el procesamiento y distribución del archivo PDB. El archivo PDB online [27] es el repositorio que coordina y relaciona información de cerca de 60.000 estructuras, incluyendo proteínas, ácidos nucleicos, y complejas macromoléculas que han sido obtenidas a través de técnicas de cristalografía de rayos X, NMR (Nuclear Magnetic Resonance) y microscopio electrónico.

A partir de los ficheros del PDB, se extrae la información requerida de cada una de las proteínas. La secuencia de aminoácidos de la proteína se encuentra en la sección Primary Structure del fichero, precedida del comando SEQRES. Las coordenadas atómicas de los aminoácidos se extraen de la sección Coordinate Section y con el comando ATOM. Se eligen como referencia para el cálculo de las distancias D_{ij} entre cada par de aminoácidos i y j , los átomos de carbono alfa ($C\alpha$, es el más usado en bibliografía [17]). Se realiza el cálculo por medio de la fórmula [25] de la distancia euclídea:

$$D_{ij} = \|r_i, r_j\| = \sqrt{((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2)} \quad (1)$$

dónde $r_i = (x_i, y_i, z_i)$ y $r_j = (x_j, y_j, z_j)$ representan las coordenadas atómicas de los residuos i y j tomando como centro del aminoácido el átomo de $C\alpha$.

2.2 Representación

Un mapa de contacto es una forma de representación bidimensional de la estructura terciaria de una proteína. Dos aminoácidos de una proteína se dice que están en contacto, si la distancia entre ellos es menor o igual a un umbral dado (θ). Usando esta definición, se contruye una matriz C simétrica de tamaño $N \times N$,

dónde N es la longitud de la secuencia de aminoácidos que forma la proteína. Cada valor de esta matriz C_{ij} se corresponde con un 1 o un 0, dependiendo si hay o no contacto entre cada par de aminoácido (ij) que forma la proteína según el criterio especificado anteriormente. Los mapas de contacto de las proteínas son una manera útil e interesante de representación de la estructura de las proteínas porque capturan todas las rasgos importantes en el plegado de las proteínas.

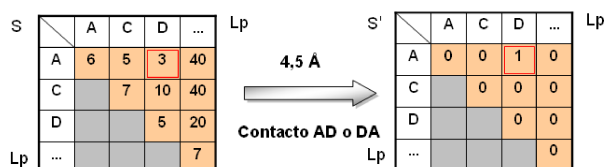


Figura 2: Una representación de una matriz de distancias (S) y un mapa de contactos (S'). L_p es la longitud de la proteína. La distancia ha de ser menor o igual que el umbral para considerarse el contacto.

Las distancias entre los aminoácidos se representan en las celdas de las llamadas matrices de distancia (Figura 2). Para una matriz S de tamaño $L_p \times L_p$, donde L_p es la longitud de la secuencia de aminoácidos, se establece un umbral de contacto (θ). Suponiendo que θ vale $4,5$ Angstroms ($\text{\AA}$), si la distancia es menor o igual a este umbral, existe contacto y se representa con un 1 en la nueva matriz S' . Por otra parte, si la distancia es mayor que este umbral, no hay contacto entre ese par de aminoácidos y se representa con un 0.

En nuestro caso, se representan las distancias entre aminoácidos en la matriz de distancias, para obtener mejores resultados en la predicción. Cada distancia se representa por un color. Este color determina si existe o no contacto en esa celda de la matriz como se puede ver en la Figura 3. Los valores reales de las distancias de la proteína se reflejan en la triangular superior de la matriz mientras que los valores predichos se representan en la triangular inferior de la misma (Figura 3).

2.3 Experimentación y predicción

Debido al enorme volumen de datos se ha ejecutado la aplicación en una estación de trabajo de 64 bits, con 32 GB DDR SDRAM y cuatro procesadores de doble núcleo mejorando así el rendimiento en los cálculos.

Para cada proteína del conjunto de entrenamiento se ha obtenido su matriz de distancias. A continuación se ejecuta el algoritmo de aprendizaje, con validación

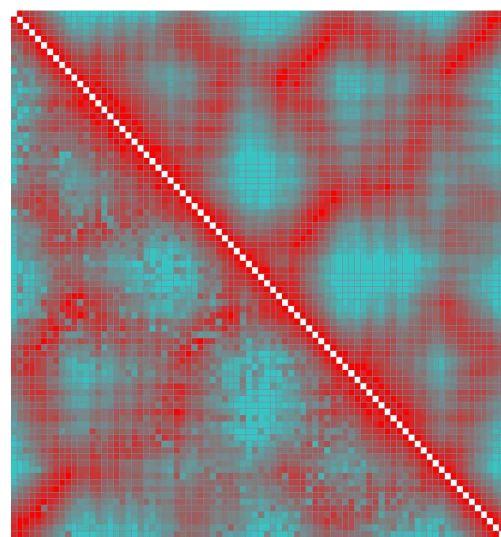


Figura 3: Un ejemplo de contact map para la proteína 1a7ge.

cruzada 10-fold, para obtener un modelo que poder aplicar sobre una nueva proteína a predecir (Figura 4).

Nuestro algoritmo calcula la distancia media para cada uno de los pares de aminoácidos con una cierta separación en la secuencia (Algoritmo A). La separación en la secuencia es un factor influyente en la predicción de proteínas [7]. Por ejemplo en la secuencia "ACDRY", los aminoácidos A y D tienen separación 1, ya que el aminoácido C se encuentra entre ellos. El modelo está formado por una matriz tridimensional de tamaño $20 \times 20 \times N$ (se representan todos los posibles pares de aminoácidos), donde N es el valor máximo de la separación en la secuencia. Cada celda de la matriz representa la distancia media sobre el conjunto total de proteínas de un par de residuos con una separación en la secuencia de entre 0 y N resultante del algoritmo A.

Se ha obtenido una matriz de confusión para cada proteína. Se trata de una manera de verificar la precisión en la predicción. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real. En una matriz de confusión se reflejan valores estadísticos que ayudan a entender la fiabilidad del algoritmo. Estos valores son el número de True Negatives, False Positives, False Negatives, y True Positives. True negative es el número de predicciones correctas para un caso negativo (No contacto), False Positive es el número de predicciones incorrectas para un caso positivo (Contacto), False negative es el número de predicciones incorrectas para

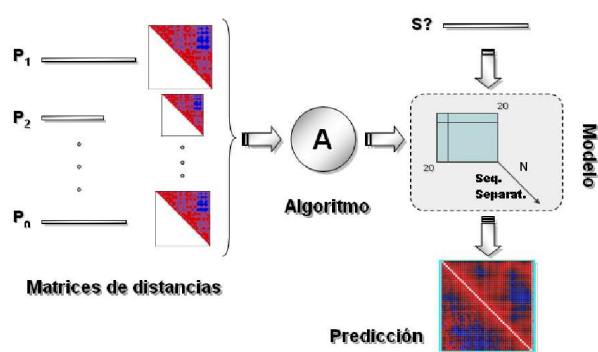


Figura 4: Procedimiento de experimentación y predicción.

un caso negativo (No contacto) y True positive es el número predicciones correctas para un caso positivo (Contacto).

A partir de estos valores se obtienen una serie de medidas estadísticas detalladas a continuación.

- La Sensibilidad representa la proporción de aciertos correctamente identificados. En nuestro caso indica qué proporción de contactos estimados han sido correctamente identificados con respecto al total de contactos.
- La Especificidad mide la proporción de negativos correctamente identificados. En este caso indica qué proporción de “no contactos” han sido correctamente identificados.
- La Precisión indica la proporción de positivos reales con respecto al total de positivos predichos. Se calcula con el valor de los falsos positivos, es decir aquellos contactos estimados que no lo sean en realidad.

3 Resultados

Las medidas especificadas se muestran en la Tabla 1. Estas medidas representan los resultados obtenidos por el algoritmo de predicción. El conjunto de training y test esta formado por 58.083 estructuras de proteínas del PDB. Se han considerado diferentes umbrales de contacto. Muchos de los artículos acerca de predicción de proteínas establecen este umbral entre los 4.5 y los 8.5 Å. Se ha considerado este rango en intervalos de 0.5 Å. Por ejemplo si el umbral es de 4.5 Å, se considera contacto si la distancia es menor o igual que 4.5 y que no lo hay en caso contrario.

Tabla 1: Tabla de resultados. θ representa el umbral de contacto. En cada columna se calcula la media (μ) y la desviación típica (σ)

θ	Sens. $\mu \pm \sigma$	Espec. $\mu \pm \sigma$	Prec. $\mu \pm \sigma$
8.5	0.420 $\pm$ 0.100	0.990 $\pm$ 0.083	0.978 $\pm$ 0.050
8.0	0.428 $\pm$ 0.094	0.992 $\pm$ 0.075	0.989 $\pm$ 0.048
7.5	0.357 $\pm$ 0.074	0.996 $\pm$ 0.055	0.995 $\pm$ 0.035
7.0	0.339 $\pm$ 0.078	0.993 $\pm$ 0.071	0.989 $\pm$ 0.047
6.5	0.299 $\pm$ 0.096	0.992 $\pm$ 0.073	0.986 $\pm$ 0.071
6.0	0.154 $\pm$ 0.080	0.993 $\pm$ 0.072	0.978 $\pm$ 0.115
5.5	0.003 $\pm$ 0.028	0.996 $\pm$ 0.059	0.015 $\pm$ 0.110
5.0	0.002 $\pm$ 0.004	0.998 $\pm$ 0.039	0.004 $\pm$ 0.061
4.5	0.000 $\pm$ 0.026	0.996 $\pm$ 0.053	0.006 $\pm$ 0.068
4.0	0.000 $\pm$ 0.004	0.998 $\pm$ 0.032	0.003 $\pm$ 0.053

Cada columna representa el valor de la media y desviación típica de la Sensibilidad, Especificidad y Precisión. La Sensibilidad representa la proporción de positivos reales (contactos) que hemos sido capaces de predecir para cada umbral. Ésta empieza a generar buenos resultados a partir de $\theta > 7.5\text{Å}$ ($Sens = 0.357$). La columna de la Especificidad representa la proporción de negativos reales (no contactos) que hemos sido capaces de predecir. La predicción de negativos es buena en todos los casos ($Spec. \simeq 0.99$). Observando la Precisión para $\theta > 6\text{Å}$ el algoritmo empieza a funcionar bien, ya que la proporción de contactos acertados con respecto a los predichos es muy alta, cercana a 1. El nivel de la sensibilidad está relacionado con el de la precisión, por tanto cuando la sensibilidad aumenta a partir de 6 Å, los valores de precisión también se incrementan. Se produce un aumento de contactos acertados con respecto al nivel de falsos contactos a partir de ese umbral, lo que explica el incremento de la precisión.

4 Conclusiones

En este artículo se ha establecido un marco de referencia para la calidad de precisión de cualquier algoritmo de predicción de estructura secundaria de proteínas. Los resultados obtenidos indican el valor mínimo de precisión en las predicciones. Se ha utilizado un algoritmo basado en distancias medias entre aminoácidos y se ha considerado como conjunto de entrenamiento el total de proteínas del PDB. La comunidad científica puede utilizar como referencia los valores aquí propuestos, ya que no existe una estandarización de la medida de la precisión en la predicción de estructura secundaria de proteínas.

Agradecimientos

Esta investigación ha sido financiada gracias al Proyecto de Excelencia P07-TIC-02611 “Sistemas Inteligentes para descubrir patrones de comportamiento. Aplicación a base de datos biológicas”.

Referencias

- [1] Berg J, Stryer L. Biochemistry. *Reverte*, 2008.
- [2] Cheng J, Baldi P. Improved residue contact prediction using support vector machines and a large feature set. *Bioinformatics*, 8, Pág.113, 2007.
- [3] Chou P, Fasman G. Prediction of protein conformation. *Biochemistry*, 13(2), Pág.222–245, 1974.
- [4] Cuff JA, Clamp ME, Siddiqui AS, Finlay M, Barton GJ. JPred. a consensus secondary structure prediction server. *Bioinformatics*, 14, Pág.892–893, 1999.
- [5] Cutello V, Narzisi G, Nicosia G. A multi-objective evolutionary approach to the protein structure prediction problem. *J. Roy. Soc. Int.*, 3, Pág. 139151, 2006.
- [6] Fariselli P, Casadio R. A neural network based predictor of residue contacts in proteins. *Protein Engineering*, 12, Pág.15–21, 1999.
- [7] Fariselli P, Olmea O, Valencia A, Casadio R. Prediction of contact map with neural networks and correlated mutations. *Protein Engineering*, 14, Pág.133–154, 2001.
- [8] Frishman D, Argos P. Incorporation of non-local interactions in protein secondary structure prediction from the amino acid sequence. *Protein Engineering*, 9, Pág.133–142, 1996.
- [9] Garnier J, Osguthorpe D, Robson B. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. Mol. Biol.*, 120, Pág.97–120, 1978.
- [10] Gu J, Bourne P. Structural Bioinformatics (Methods of Biochemical Analysis) 2003.
- [11] Guo J, Chen H, Sun Z, Lin Y. A novel method for protein secondary structure prediction using dual-layer SVM and profiles. *Proteins Structure Function and Bioinformatics*, 54, Pág.738–743, 2004.
- [12] Gupta N, Mangal N, Biswas S. Evolution and Similarity Evaluation of Protein Structures in Contact Map Space, *Proteins: Structure, Function, and Bioinformatics*, 59, Pág. 196–204, 2005.
- [13] Kim H, Park H. Prediction of protein relative solvent accessibility with support vector machines and long-range interaction 3D local descriptor. *Proteins*, 54, Pág.557–562, 2003.
- [14] Kneller DG, Cohen FE, Langridge R. Improvements in Protein Secondary Structure Prediction by An Enhanced Neural Network. *J. Mol. Biol.*, 214, Pág.171–182, 1990.
- [15] Lim VI. Algorithms for Prediction of a-Helical and b-Structural Regions in Globular Proteins, *J. Mol. Biol.*, 88, 857–872, 1974.
- [16] Lin K, Simossis VA, Taylor WR, Heringa J. A simple and fast secondary structure prediction method using hidden neural networks. *Bioinformatics*, 21, Pág.152–159, 2005.
- [17] Mirny L, Domany E. *Proteins*, 26, Pág.391–410, 1996.
- [18] McGuffin LJ, Bryson K, Jones DT. The PSIPRED protein structure prediction server. *Bioinformatics*, 16, Pág.404–405, 2000.
- [19] Pedersen JT, Moult J. Protein folding simulations with genetic algorithms and a detailed molecular description. *Mol. Biol.*, 269, Pág.240259, 1997.
- [20] Qian N, Sejnowski T. Predicting the secondary structure of globular proteins using neural network models. *J. Mol. Biol.*, 202, Pág.865–884, 1988.
- [21] Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.*, 232, Pág.584–599, 1993.
- [22] Salamov A, Solovyev V. Protein Secondary Structure Prediction Using Local Alignments. *J. Mol. Biol.*, 268, Pág.31–36, 1997.
- [23] Ward J, McGuffin L, Buxton B, Jone D. Secondary structure prediction with support vector machines. *Bioinformatics*, 13, Pág.1650–1655, 2003.
- [24] Yi TM, Lander ES. Protein secondary structure prediction using nearest-neighbor methods. *J. Mol. Biol.*, 232, Pág.1117–1129, 1993.
- [25] Zhang G, Han K. Hepatitis C virus contact map prediction based on binary strategy. *Comp. Biol. and Chem.*, 31, Pág.233–238, 2007.

- [26] <http://www.wwpdb.org> , Protein Data Bank web.
- [27] <ftp://ftp.wwpdb.org> , Protein Data Bank online repository.

Cis-cop: Multiobjective identification of cis-regulatory modules based on constraints

Rocío Romero-Zalíz¹ M. Martínez Ballesteros¹ Igor Zwir^{1,2} Coral del Val¹

¹ DECSAI, UGR, Granada, Spain, {rocio,delval,igor}@decsai.ugr.es

² Howard Hughes Medical Institute, St. Louis, Missouri, USA

Abstract

Gene expression regulation is an intricate, dynamic phenomenon essential for all biological functions. The necessary instructions for gene expression are encoded in cis-regulatory elements that work together and interact with the RNA polymerase to confer specific spatial and temporal patterns of transcription. Therefore, the identification of these elements is currently an active area of research in computational analysis of regulatory sequences. However, the problem is difficult since the combinatorial interactions between the regulating factors can be very complex. Here we present a web server, Cis-cop, that identifies cis-regulatory modules given a set of transcription factor binding sites and, additionally, also RNA pol sites for a group of genes.

1 INTRODUCTION

Gene expression regulation is an intricate, dynamic phenomenon essential for all biological functions. The necessary instructions for gene expression are encoded in cis-regulatory elements, or modules (CRMs). A CRM consists of a set of transcriptional factors (TFs) that work together [6] and interact with the RNA polymerase (RNA pol) to confer specific spatial and temporal patterns of transcription. Therefore, the identification of CRMs is currently an active area of research in computational analysis of regulatory sequences [2, 8]. However the problem is difficult since the combinatorial interactions between the regulating factors can be very complex.

Here we present a web server, Cis-cop, that identifies CRMs given a set of TFs binding sites and, additionally, also RNA pol sites for a group of genes.

The server takes into account spatial constraints on the arrangement of cis-elements such as relative position, distance and strand orientation. Cis-cop is also able to deal with more than one binding site for a certain TF in the same promoter region. The web server infers these modules using multi-objective and multi-modal techniques [4]. Our algorithm is able to identify not only the most obvious solutions (i.e., the most frequent), but also those that are not locally dominated and that represent less frequent but relevant ones.

There is a large number of DNA motif finding algorithms and a lack of standards to measure their correctness. Most of the algorithms perform better in lower organisms, including yeast, as compared to higher organism [9]. For this reason, we leave to the user the selection of one or multiple DNA motif finding algorithms or databases available. These algorithms may use combinatorial enumeration, probabilistic modeling (Stormo, Gibbs, AlineACE, ANN-Spec, MEME), mathematical programming, neural networks and/or genetic algorithms (EC, GAME), while databases can be TRANSFAC, REGULON DB, JASPAR, etc.

2 METHODOLOGY

The proposed methodology is depicted in Figure 1. The first step is the generation of Association Rules [1]. That is, discover the sets of features that appear frequently together in the set of cis-features in the input csv file. To do this, we use a modification of the Apriori algorithm [1] on Borgelt's eclat implementation [3] where candidate itemsets are counted in a pass and not generated on-the-fly. Borgelt's implementation is based on the idea to organize the counters for the itemsets in a special kind of prefix tree, which not only allows us to store them efficiently using little memory, but also supports processing the transactions as well as generating the rules. A further modification has been designed in order to deal with multi-objective prob-

lems. Apriori's method can be interpreted as a search problem with one objective function: the support of the itemset. In our case, we introduce a new objective to be maximized, complexity. Support is given by the number of genes that have the same cis-elements as the itemset, while complexity is calculated by the number of different cis-elements of the itemset. We extract itemsets from a database of transactions that maximize both support and complexity functions by post-processing the information generated by the Apriori algorithm (Figure 1).

As the number of features and genes (transactions) grows, the number of possible solutions increases exponentially. At a certain level of features and gene combination the eclat implementation is unable to produce results halting by a memory problem. For such cases we have developed a heuristic method. It uses an evolutionary approach to the problem of finding the best itemsets in a database of transactions. We propose a genetic algorithm (GA) based on the NSGA-II multi-objective approach.

The NSGA-II algorithm has been demonstrated as one of the most efficient algorithms for multi-objective optimization on a number of benchmark problems [5]. NSGA-II uses non-dominated sorting for fitness assignments. All individuals not dominated by any other individuals, are assigned front number 1. All individuals only dominated by individuals in front number 1 are assigned front number 2, and so on. Selection is made, using tournament between two individuals. The individual with the lowest front number is selected if the two individuals are from different fronts. The individual with the highest crowding distance is selected if they are from the same front. i.e., a higher fitness is assigned to individuals located on a sparsely populated part of the front. There are N parents and in every iteration N new individuals (offspring) are generated. Both parents and offspring compete with each other for inclusion in the next iteration.

The proposed GA instantiates the NSGA-II algorithm using a simple chromosome composed by a list of module structures (e.g., $\langle (AP2A,0.3,D,0), (SP1,0.6,D,1), (SP1,0.9,D,110), (SP1,0.5,D,25), (SP1,0.7,D,32) \rangle$). A module structure consists of five features: name, score, orientation and distance (e.g., name: CAAT, score: 0.7, orientation: D, distance: 30: (CAAT,0.76,D,30)). Some of the features of the chromosome may be not used in some cases, for instance, when strand is ignored, then the orientation feature will be ignored. One-point crossover is used, that is, the chromosomes of the parents are cut at some randomly chosen common point (between module structures) and the resulting sub-chromosomes are swapped. Different mutation operators are used: add, erase and modify. An add

mutation operator simply extends the chromosome by adding a new module structure at the beginning or the end of the chromosome. Erase mutation operator selects any module structure from the chromosome and deletes it. Finally, a modification mutation operator selects one of the features of the module structure and changes it. If the name is chosen, then a random name from the set of valid names is chosen and replaced in the module structure; if the orientation is chosen, then strand changes from D to R and vice versa. Score feature is not used in the evaluation process; therefore it is not used in any mutation operator. Finally, if distance is chosen, a small integer is added or subtracted from the one in the selected module structure from the chromosome. Again, strand and distance features are used only if the user selected them in the original parameters.

3 WEB SERVER

3.1 INPUT

The input form requires a file, provided by the user in gff format. Each entry of the csv input file contains the information about which TFBs are present in which promoters. The csv file contains the following fields: sequence name, source, feature, start, end, score, and strand and DNA pattern. As input parameters, the user can select which restrictions to apply to the search process: strand, same ordering of TFBs in the module, same relative distance among the TFBs, which can be crisp or fuzzy. There are two Cis-cop versions: exhaustive, for small input files, and heuristic for larger datasets. Once the parameters are selected, the query starts by clicking the 'submit' button.

3.2 OUTPUT

Results are provided as HTML for visual inspection and can also be received by e-mail. In case of error, a human-readable message is displayed. The output is a graphical panel containing a detailed description of all cis-features found in the CRM (Figure 2). The user can explore individually each module. All views can be saved as graph (e.g., png, pdf, jpg) or text (e.g., txt, csv, html).

4 DISCUSSION

The server is available at <http://gps-tools2.wustl.edu/modulos/modules.html> and it is open to all users and no login is required. There are two versions available, exhaustive and heuristic, both of them as required have an available tutorial along with example test files. The tu-

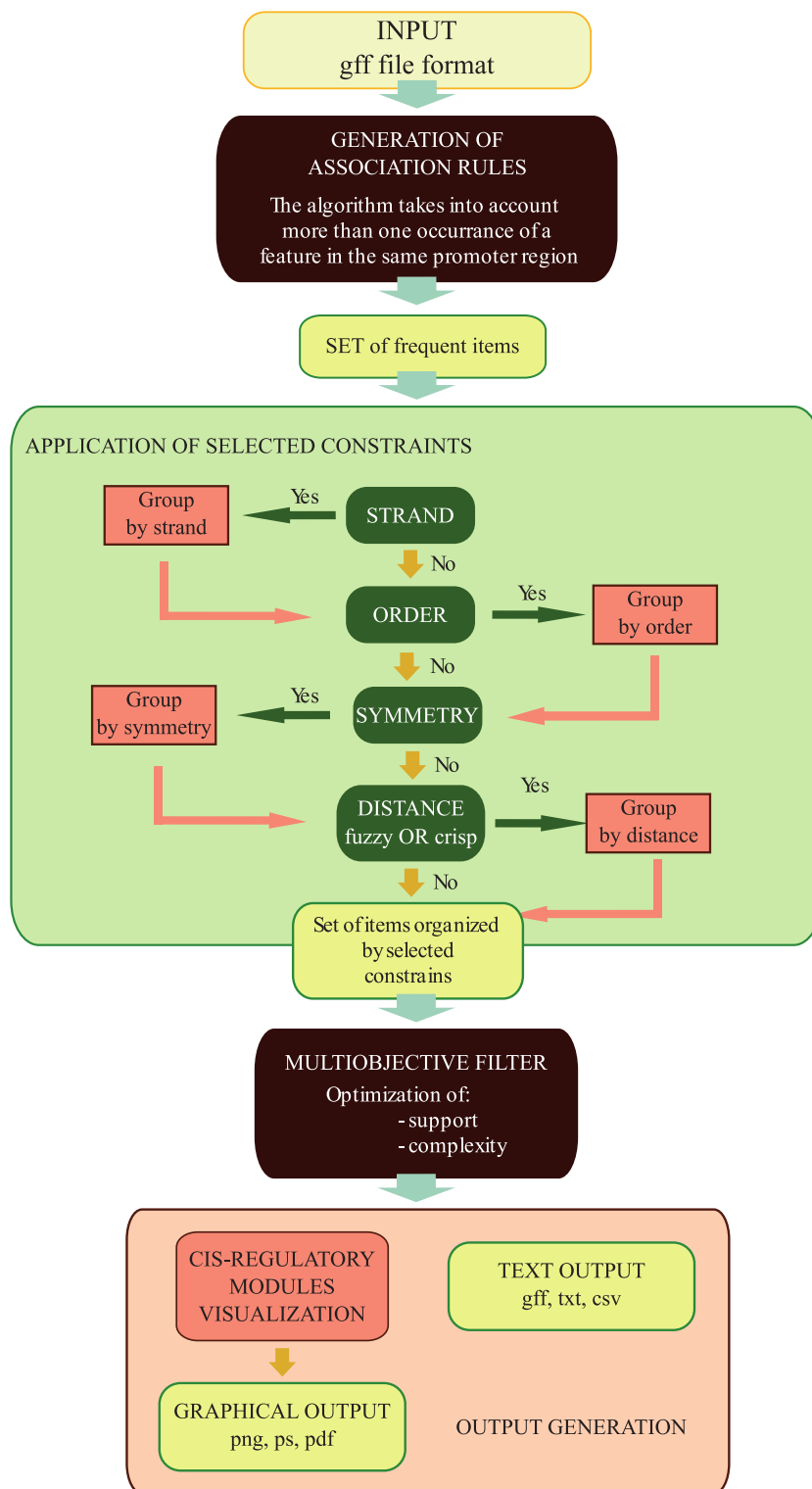


Figure 1: Pipeline

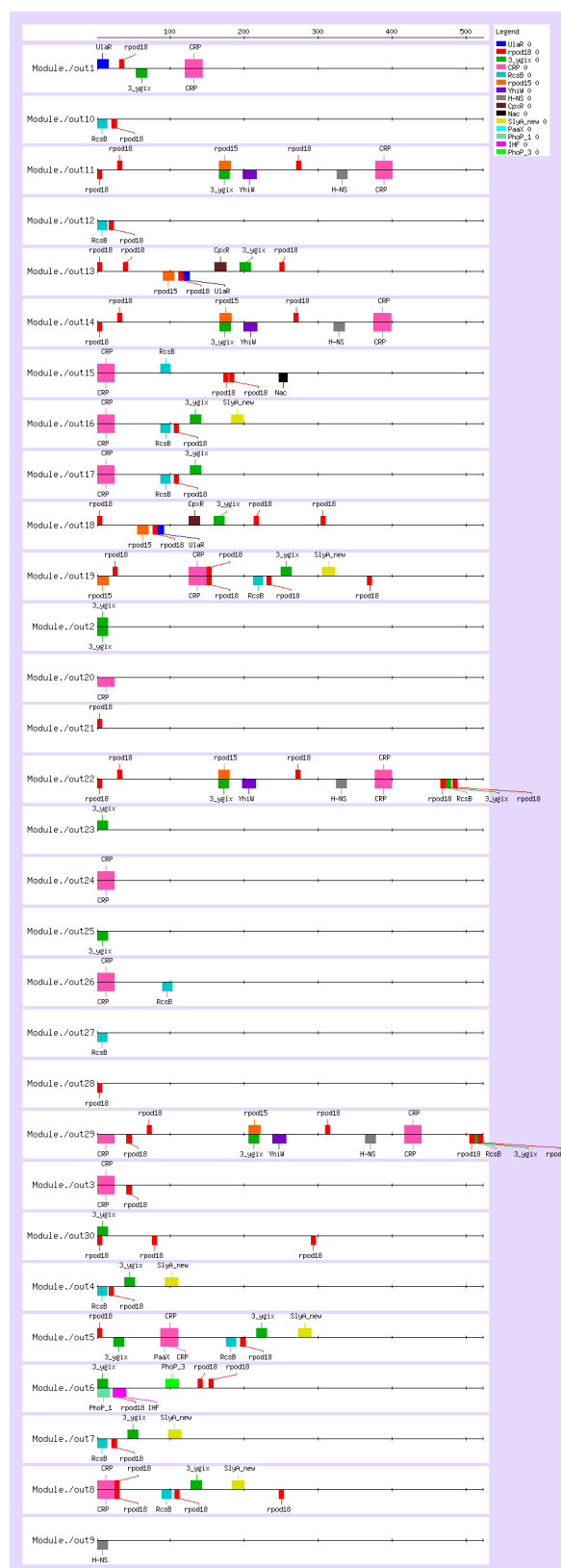


Figure 2: Modules image

torials explain which parameters can be modified and cover the following help topics: input, maximal number of features to be considered in a combinatorial module, order and distance, orientation, e-mail results, and output results. The development of the server presented here has been user-driven from the beginning. The here described approach has been already successfully used to identify PhoP regulated promoters harboring more than one binding site for the TF PhoP and sharing an atypical orientation and distance of the PhoP box/boxes to the RNA polymerase site [10] leading the results to the inference of PhoP transcription control over acid resistance genes in *Salmonella typhimurium* [10, 7]. The server has been active for the last year and the number of sequences used by de users varied very much from 10-500.

References

- [1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, D.C., 1993.
- [2] B.P. Berman, Nibu Y., Pfeiffer B.D., Tomancak P., Celniker S.E., Levine M., Rubin G.M., and Eisen M.B. Expliting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the *drosophila* genome. *PNAS*, 99:757–762, 2002.
- [3] Borgelt C. Efficient implementations of apriori and eclat. In *Workshop of Frequent Item Set Mining Implementations*, 2003.
- [4] K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc., 2001.
- [5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6:182–197, 2002.
- [6] Davidson E.H. Genomic regulatory systems. *Nat. Genet.*, 29:153159, 2001.
- [7] Zwir I., Harari O., and Groisman E.A. Gene promoter scan methodology for identifying and classifying coregulated promoters. *Methods Enzymol.*, 422:361–385, 2007.
- [8] Markstein M. and Levine M. Decoding cis-regulatory dnas in the *drosophila* genome. *Curr Opin Genet Dev.*, 12(5):601–606, 2002.
- [9] Das M.K. and Dai H.K. A survey of dna motif finding algorithms. *BMC Bioinformatics*, 8(Suppl. 7):S21, 2007.
- [10] I. Zwir, H. Huang, and E.A. Groisman. Analysis of differentially-regulated genes within a regulatory network by gps genome navigation. *Bioinformatics*, 21:4073–4083, 2005.

Optimization of multi-classifiers using a fuzzy logic approach: an application to the gene prediction problem

Rocío Romero-Zalaz¹ Coral del Val¹ Igor Zwir^{1,2}

¹ DECSAI, UGR, Granada, Spain, {rocio,delval,igor}@decsai.ugr.es

² Howard Hughes Medical Institute, St. Louis, Missouri, USA

Abstract

Genomes of many organisms have been sequenced over the last few years. However, transforming such raw sequence data into knowledge remains a hard task. A great number of prediction programs have been developed to address part of this problem: the location of genes along a genome. We propose a multi-objective methodology using fuzzy logic to combine algorithms into an aggregation scheme in order to obtain optimal methods' aggregations. Results show improvements in specificity and sensitivity when our methodology is compared to the performance of individual methods for gene finding problems. The here proposed methodology is an automatic method generator, and a step forward to exploit all already existing methods, by providing optimal methods' aggregations to answer concrete queries for a certain biological problem with a maximized accuracy of the prediction. As more approaches are integrated, *de novo* accuracy can be expected to improve further.

1 INTRODUCTION

Genomes of many organisms have been sequenced over the last few years. However, transforming such raw sequence data into knowledge remains a hard task. A great number of prediction programs have been developed to address one part of this problem: the location of genes along a genome [2, 3, 1, 9]. Unfortunately, finding genes in a genomic sequence is far from being a trivial problem. Gene prediction is one of the most important problems in computational biology due to the inherent value of the set of protein-coding genes for other analysis.

Despite the advances in the gene finding problem, existing approaches to predicting genes have intrinsic advantages and limitations [11]. Furthermore, there is no program that can provide perfect predictions for any given input. Our methodology combines these approaches into an aggregation scheme to provide better predictions by taking advantage of the different methodologies' starknesses and avoiding their weaknesses. Moreover, we use a multi-objective approach to extract the best aggregation of methods by maximizing the specificity and sensitivity of their predictions.

We applied our methodology to a reference dataset in gene prediction containing 570 multi-species DNA sequences of known genes [5].

2 MATERIALS AND METHODS

The aggregation of methods is accomplished by using fuzzy union $\cup$ and fuzzy intersection $\cap$ operators [8, 14]. All potential aggregations conform a space of potential hypotheses, which can be represented as a lattice structure (Figure 1). We search for the best aggregation of methods, moving from hypothesis to hypothesis towards the most general (i.e., the union of all methods) and the most specific (i.e., the intersection of all methods) which are located at the top and the bottom of the lattice, respectively [12] (Figure 1). In the gene finding problem we explore three methods, $n = 3$, termed M1 to M3, conforming a total set of seven potential aggregations.

The aggregation of the different methods in the gene finding problem is performed at a nucleotide level. This aggregation joins two overlapping or adjacent exons into a new exon (Figure 2 and 3) taking into account their exon probabilities.

Even though most *ab initio* gene finders develop a scoring scheme for exon prediction, many of them only report meaningless scores referring to the predicted

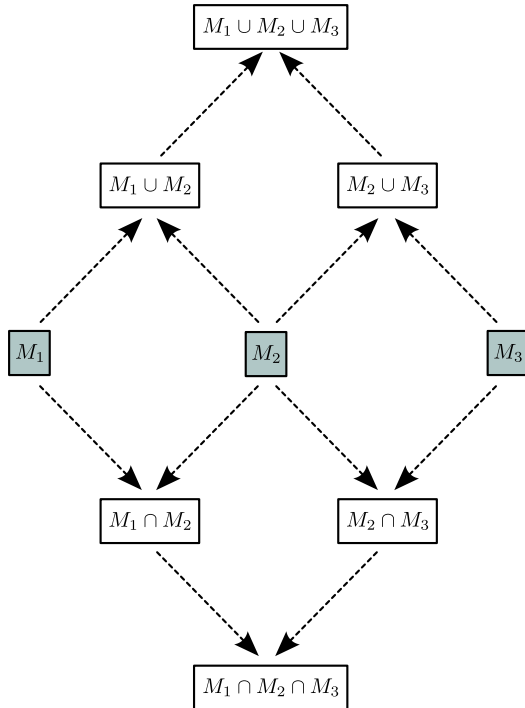


Figure 1: Lattice of potential hypothesis, methods' aggregations of $M_1, \dots, M_n$ using the $\cup$ - and $\cap$ - operators. The solid arrows show the direction of the search in the space of hypothesis.

exons. Although some gene finders, such as GENSCAN, give a probabilistic score to every predicted exon, the score does not respond to the likelihood correctly and is not reliable, especially when implementing in large DNA sequences [4]. Therefore, we applied the local polynomial regression method, a nonparametric regression model, to transform the raw scores to probabilistic ones as implemented in [10].

To perform the aggregation of exons using the fuzzy union and intersection operators, we first need to introduce some notation. We define the *exon* fuzzy set X as the a pair (A, m) where A is a set and $m : A \rightarrow [0, 1]$. For each $x \in A$, $m(x)$ is the grade of membership of x , where m corresponds to the probabilistic score calculated from the raw scores of each gene finder.

The fuzzy union operator joins two overlapped exons –exon x and exon y – when $m(x)$ and $m(y)$ are higher than a certain threshold. If $m(x) > \gamma$ while $m(y) < \lambda$, only exon x is kept (Figure 2 (c)). If $m(x) > \gamma$ and $m(y) > \gamma$, a new exon z is constructed by appending both exons (Figure 2 (b)) with $m(z) = \max(m(x), m(y))$. If there is no overlap, only the exons with membership above threshold λ are kept (Figure 2 (a)).

The fuzzy intersection operator intersects two over-

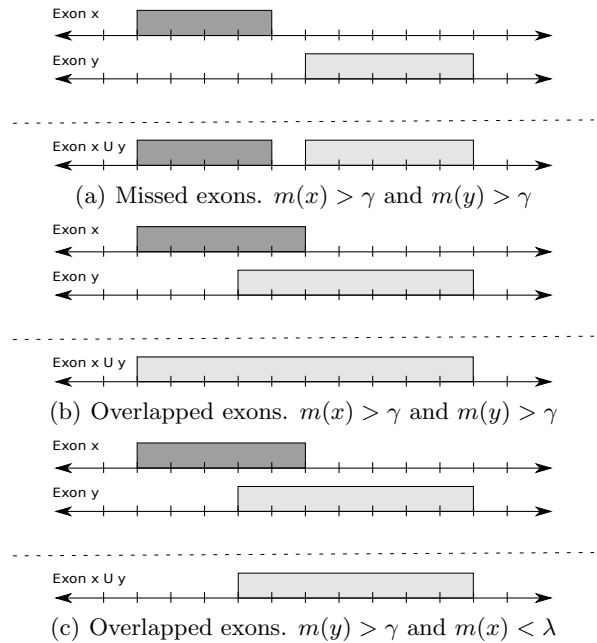


Figure 2: Example of exons aggregation by the fuzzy union operator.

lapped exons –exon x and exon y – when $m(x)$ and $m(y)$ are, again, higher than a certain threshold. If $m(x) > \lambda$ and $m(y) > \lambda$, a new exon z is constructed by taking only those nucleotides appearing in both exons (Figure 3 (b)) with $m(z) = \min(m(x), m(y))$. If there is an overlap but $m(x) < \lambda$ or $m(y) < \lambda$, then no intersection is performed (Figure 3 (a)). If there is no overlap, neither exon x nor exon y is kept (Figure 3 (c)).

For the experimental section we used a threshold $\gamma = 0.8$ and a threshold $\lambda = 0.2$.

2.1 DATASET

We selected the dataset from Guigó et al. [5] which is a reference for assessing the quality of gene prediction programs. This set contains 570 sequences from vertebrate genomes 570, having only those sequences representing only one complete spliceable functional product of a gene in the forward strand. The programs used in this study are Genscan [1], GeneID [7] and Augustus [16]. Genscan uses a general probabilistic model for the gene structure of human genomic sequences. It has the capacity to predict multiple genes in a sequence, to deal with partial as well as complete genes, and to predict consistent sets of genes occurring on either or both DNA strands [1]. GeneID combines different algorithms using Position Weight Arrays to detect features such as splice sites, start and stop codons and Markov Models to score exons

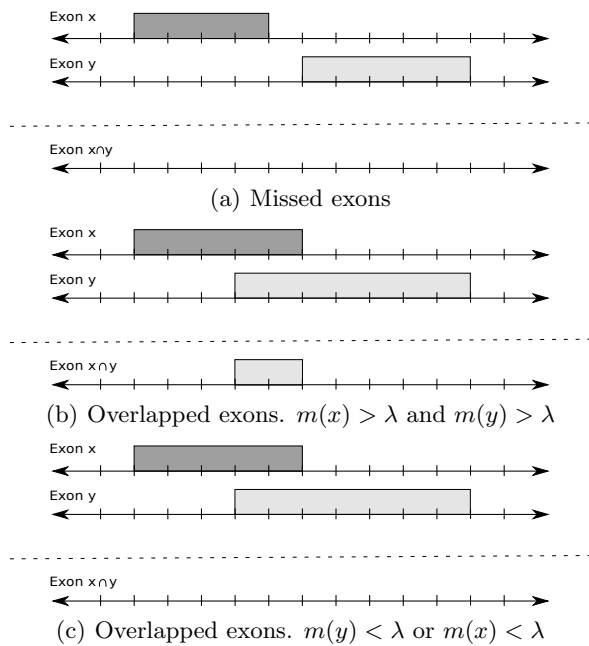


Figure 3: Example of exons aggregation by the fuzzy intersection operator.

and Dynamic Programming (DP) to assemble the gene structure [7]. Augustus is a gene predictor for eukaryotic genomic sequences that is based on a generalized hidden Markov model, a probabilistic model of a sequence and its gene structure [16].

2.2 MEASURE OF ACCURACY OF PREDICTIONS

We measured the accuracy of a prediction on a test sequence by comparing the predicted coding value (coding or non-coding) with the true coding value for each nucleotide along the test sequence. This has been one of the most widely used approaches in evaluating the accuracy of coding region identification and gene structure prediction methods. Nucleotide level accuracy is calculated as a comparison of the annotated nucleotides with the predicted nucleotides. Sensitivity (Sn) (Equation 1) is the proportion of annotated nucleotides (as being coding or part of an mRNA molecule) that is correctly predicted, and specificity (Sp) (Equation 2) the proportion of predicted nucleotides (as being coding or part of an mRNA molecule) that is so annotated. As a summary measure, we have computed the correlation coefficient (CC) (Equation 3) between the annotated and the predicted nucleotides [5].

$$Sn = \frac{TP}{TP + FN} \quad (1)$$

$$Sp = \frac{TP}{TP + FP} \quad (2)$$

$$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}} \quad (3)$$

3 RESULTS

Out of all gene prediction programs analyzed and all methods' aggregations, the union of all methods – Genscan $\cup$ GeneID $\cup$ Augustus – achieved the highest number of correctly predicted genes¹ (525 out of 570, over 92% of the dataset) and the highest average CC, 0.896 (Table 1). Whats more, these percentage of correctly predicted gene is increased by a 10% approximately when compared with the best individual predictor, GeneID. The Genscan $\cup$ GeneID methods' aggregation achieved the best specificity values while the union of all methods obtained the highest sensitivity value. All methods' aggregations using the fuzzy union operator obtained better sensitivity values compared to the individual methods. Moreover, some of these methods's aggregations' specificity values are also better than most of the individual gene predictors, while the others do not differentiate to much from them. If a crisp union operator is used, the sensitivity values are increase, but most of the time its specificity values decrease (data not shown) [15].

On the other hand, the fuzzy intersection operator proposed did not produce better results than individual methods (Table 1). This is mainly due to the fact that the fuzzy intersection greatly decreases the sensitivity of the results, and thus producing a very low CC.

A graphical representation of the methods' aggregations performance can also be seen in Figure 3. Specificity and sensitivity values are plotted for all methods' aggregations, both using the fuzzy union or intersection fuzzy operators. Methods' aggregations belonging to the Pareto set are highlighted in red, i.e., those methods that are both better in specificity and sensitivity than the rest. We can therefore infer that Genscan $\cup$ GeneID $\cup$ Augustus and Genscan $\cup$ GeneID methods' aggregations are better in both specificity and sensitivity than individual methods.

If we take a closer look into the results we can extract many specific genes where individual methods fail, while the aggregation of methods produced better results (e.g., MMU12565, HUMSEMIIB, MMIL5G).

¹We express the accuracy of the method aggregation by considering a gene correctly retrieved when its CC > 0.7.

Method	Sp	Sn	CC	Correctly predicted %
Genscan	0.885	0.753	0.753	78.42%
GeneID	0.899	0.808	0.830	82.28%
Augustus	0.829	0.715	0.796	73.33%
Genscan $\cup$ GeneID	<i>0.902</i>	0.903	0.881	90.35%
Genscan $\cup$ Augustus	0.882	0.841	0.847	84.56%
Augustus $\cup$ GeneID	0.900	0.894	0.886	90.00%
Genscan $\cup$ GeneID $\cup$ Augustus	0.893	<i>0.928</i>	<i>0.896</i>	<i>92.11%</i>
Genscan $\cap$ GeneID	0.836	0.622	0.680	64.91%
Genscan $\cap$ Augustus	0.783	0.586	0.657	61.93%
Augustus $\cap$ GeneID	0.809	0.613	0.696	63.16%
Genscan $\cap$ GeneID $\cap$ Augustus	0.757	0.517	0.601	52.98%

Table 1: Results obtained by all methods' aggregation using both the fuzzy union and the fuzzy intersection operators. The best result for each column is highlighted in italic and color-coded in blue.

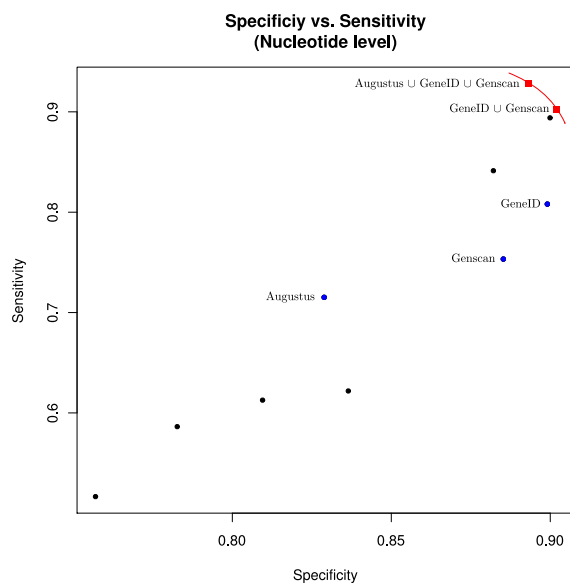


Figure 4: Graphical representation of the specificity and sensitivity values obtained by the methods' aggregations using the fuzzy union and fuzzy intersection operators to predict genes. Methods' aggregations belonging to the Pareto set are highlighted in red.

4 DISCUSSION

We propose a methodology to combine programs into a aggregation scheme using fuzzy logic operators. This idea provides better predictions by combining the advantages of the different methodologies used in each program. We introduced the use of a multi-objective approach to extract the best aggregation of methods by maximizing the specificity and sensitivity of their predictions. This way we avoid redundant and overlapping predictions that might be produced depending on the methodologies and the aggregation scheme

used. The application of the proposed methodology to the gene finding problem to obtain optimal methods' aggregations showed an improvement in both sensitivity and specificity when compared to the performance of individual methods. The specificity levels obtained by the aggregation of gene finding methods improved or decreased depending on the methods used in the aggregation. When determining which aggregation of methods was the best one for the gene prediction problem, sensitivity and specificity were in contradiction. Nevertheless, the calculation of the correlation coefficient helped in the selection of the best methods' aggregation. The best aggregations include methods employing different algorithmic strategies that predict correctly different subset of the genes in the dataset. Although the statistical properties of coding regions allow for a good discrimination between large coding and non-coding regions, the exact identification of the limits of exons or of gene boundaries remains difficult.

There are several previous works combining gene finding programs [13, 17], but they fail to obtain good results as they use simultaneously all programs instead of optimize their aggregation. *De novo* gene prediction for compact eukaryotic genomes is already quite accurate, although mammalian gene prediction lags way behind in accuracy. One future scope would be the application of this approach to identify ways to quickly combine many or all-existing programs trained for the same organism, and determine the upper limit of predictive power by aggregations of programs genome wide [6].

In the last ten years, the existing competitive spirit has increased the number of programs/algorithms created, updated and adapted for the two biological problems here presented [11, 2, 9]. On one side the development of a new algorithm always implies the sacrifice of an objective in favor of another, which makes very difficult for novel approaches to improve in absolute terms

the quality of the existing ones. On the other side, the impressive amount of alternative algorithms available for different biological problems is confusing the users, who wonder what makes the programs different, which one should be used in which situation and which level of prediction confidence to expect. Finally, users also wonder whether current programs can answer all their questions. The answer is most probably no, and will remain to be negative as it is unrealistic to imagine that such complex biological processes can be explained merely by looking at one objective. The here proposed methodology is an automatic method generator, and a step forward to exploit all already existing methods, by providing optimal methods' aggregations to answer concrete queries for a certain biological problem with a maximized accuracy of the prediction.

References

- [1] C. Burge and S. Karlin. Finding the genes in genomic dna. *Struct. Biol.*, 8:346–354, 1998.
- [2] J. M. Claverie. Computational methods for the identification of genes in vertebrate genomic sequences. *Hum. Mol. Genet.*, 6:1735–1744, 1997.
- [3] R. Guigó. Computational gene identification: an open problem. *Comput. Chem.*, 21:215–222., 1997.
- [4] R. Guigó, P. Agrawal, J.F. Abril, M. Burset, and J.W. Fickett. An assessment of gene prediction accuracy in large dna sequences. *Genome Res.*, 10:16311642, 2000.
- [5] R. Guigó and M. Burset. Evaluation of gene structure prediction programs. *Genomics*, 34(3):353–367, 1996.
- [6] R. Guigó et al. Identification and analysis of functional elements in 1% of the human genome by the encode pilot project. *Nature*, 447, 2007.
- [7] R. Guigó, S. Knudsen, et al. Prediction of gene structure. *J. Mol. Biol.*, 226:141–157, 1992.
- [8] P. Halmos. *Naive Set Theory*. D. Van Nostrand Company., Princeton, NJ, 1960.
- [9] D. Haussler. Computational genefinding. *Trends Biochem. Sci.*, pages 12–15., 1998.
- [10] Xiao Li, Qingan Ren, Yang Weng, Haoyang Cai, Yunmin Zhu, and Yizheng Zhang. Scgpred: A score-based method for gene structure prediction by combining multiple sources of evidence. *Genomics, Proteomics & Bioinformatics*, 6(Issues 3-4):175–185, 2008.
- [11] C. Mathé, M. F. Sagot, et al. Current methods of gene prediction, their strengths and weaknesses. *Nucleic Acids Res.*, 30(19):4103–4117, 2002.
- [12] T. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.
- [13] K. Murakami and T. Takagi. Gene recognition by combination of several gene-finding programs. *Bioinformatics*, 14:665–675, 1998.
- [14] W. Pedrycz, P. Bonissone, and E. Ruspini. *Handbook of fuzzy computation*. Institute of Physics, 1998.
- [15] R. Romero-Zaliz, C. Rubio-Escudero, I. Zwir, and C. del Val. Optimization of multi-classifiers for computational biology: application to gene finding and expression. *Theor Chem Acc*, 125, 2010. In Press.
- [16] Mario Stanke and Stephan Waack. Gene prediction with a hidden markov model and a new intron submodel. *Bioinformatics*, 19, Suppl. 2:ii215–ii225, 2003.
- [17] M. Tech and R. Merkl. Yacop: Enhanced gene prediction obtained by a combination of existing methods. *Silico Biology*, 3(4):441–451, 2003.

A coevolutionary genetic algorithm able to learn a FRBS and detect the less informative instances in low quality problems

Luciano Sánchez

¹ Universidad de Oviedo. Departamento de Informática. luciano@uniovi.es

Abstract

The evaluation of the fitness function in genetic fuzzy systems with low quality data is a costly process, and this conditions the applicability of this technique to large datasets. Nevertheless, we have observed that, in many practical cases, a few instances are responsible for the main part of the dispersion of the fitness function.

In this paper we propose a new technique for coevolving a FRBS and a list containing the less informative instances in the dataset. We guide the learning with an approximate fitness, that combines a crisp component with an estimation of the dispersion of the fitness value, obtained from the mentioned subset of instances.

1 Introduction

Obtaining Fuzzy Rule Based Systems (FRBS) from low quality data with Genetic Algorithms (GA) differs from the crisp case [2] on the use of a fuzzy fitness function [16], and on an order structure defined in the set of images of this function, that allows us to compare two FRBSs [9, 11, 18]. Nonstandard GAs are also used; a modified scalar genetic algorithm can find the primal element of this order, or an adapted multicriteria algorithm can retrieve a set of nondominated individuals [8, 17].

At the root of this search, we need to know the error of each candidate FRBS in the population, for any selection of the input data which is compatible with our set of imprecise measurements [16]. If our dataset is interval-valued, we need to obtain the extrema of the error of the model for each instance, when their inputs are box-constrained to the corresponding interval-valued data. The interval arithmetic-based

sum of these couples of bounds is regarded as an imprecise perception of the quality of the FRBS at this instance [3]. Else, if our data is fuzzy, these bounds must be produced for each α -cut of the data, and fuzzy arithmetic is used to compute the fuzzy fitness of the FRBS.

There are some difficulties that limit the applicability of this schema. Obtaining the bounds of the error of a model in an interval is a costly computational problem unless these extrema are known to be in the boundary of the region [15]. Since, generally speaking, the error of the FRBS will not be monotonic, it is expected that this stage will consume a large part of the learning time, precluding the application of this kind of algorithms to large datasets. That is the reason why, in this paper, we try to overcome this limitation by exploiting certain characteristics of the fitness function.

Let us introduce our idea with an example. In Figure 1 we have plotted the decision surface of a classifier, whose error we want to evaluate. The squares represent imprecise measurements of objects; that is to say, the actual measurements could be anywhere inside each square. Observe that, in the worst case, the error of this classifier can be as high as 3/15. In the best case it is 0. Observe also that the vagueness of the error of the classifier in the figure depends only on the three instances that have been marked: from a computational point of view, there is no need for obtaining the bounds of the value of the decision function at the 12 intervals that do not intersect the decision surface. We might as well compute the decision function at the centerpoint of each square except at the three “conflicting” instances, where we must launch that costly optimization algorithm we have just mentioned.

Let us introduce now the regression case with a second example. Observe the graph in Figure 2. In this case, all instances (A, B and C) would contribute to the vagueness of the fitness. But we can replace the output of the model in A and C by the output in the midpoints

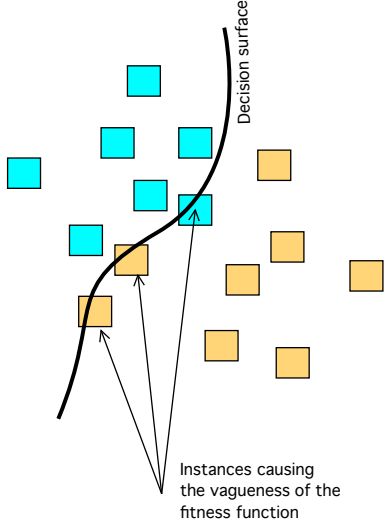


Figure 1: The vagueness in the estimation of the classification error is caused by the three marked elements.

of these intervals, without introducing large errors. In this case, we say that the less informative instance is B.

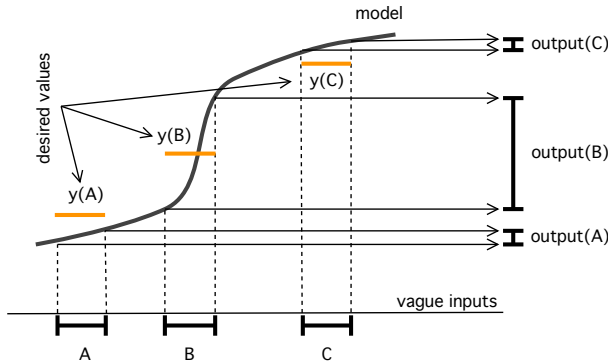


Figure 2: Most of the vagueness in the estimation of the regression error will depend on the instance 'B'.

The rationale of this work is, therefore: when learning FRBSs from low quality data, we can save most of the computing time if we replace each imprecise instance by a crisp value in all but the less informative instances. If we identify these points then we can divide the fitness computation into two parts: fuzzy (or interval) values at these instances, and crisp values at the remaining part of the dataset.

The remaining part of the paper is devoted to define a GA that coevolves a population of FRBSs and the mentioned set of instances (Section 2), thus solving at the same time two problems: the learning of a FRBS from vague data, and the detection of the instances of

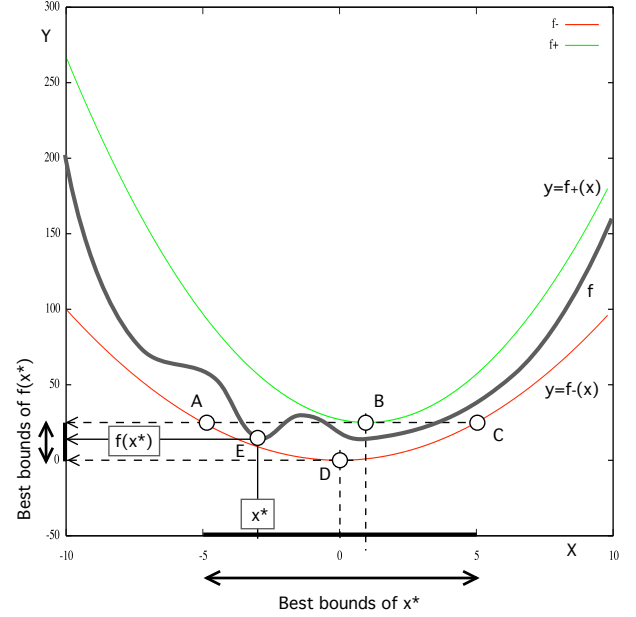


Figure 3: Interval optimization: $f^-(x) \leq f(x) \leq f^+(x)$. f^- and f^+ are known, but f is not. Hence, the minimum of f cannot be known; at the most we can bound the values of x and $f(x)$ at the minimum.

data that contribute the most to the variability of the fitness function. In Section 3 we provide compared numerical results. The paper finishes with the concluding remarks in Section 4.

2 A proposal of coevolutionary learning of FRBS from vague data

The objective of this work is to learn a FRBS from a dataset whose input values are imprecise. In the following, we will use an interval valued dataset $\{(A_1, y_1), (A_2, y_2), \dots, (A_n, y_n)\}$, $A_i \subset \mathbf{R}^n$. In turn, we will assume that a FRBS is completely defined by a vector of real parameters x . We will denote the output of the FRBS, given a crisp input t , by $F_x(t)$, with $F_x : \mathbf{R}^n \rightarrow \mathbf{R}$. The error of the FRBS in the mentioned dataset will be denoted by $f(x)$, and this function is unknown, but we are given a couple of functions f^- and f^+ (see Figure 3),

$$f^-(x) = \sum_{i=1 \dots N} \min_{t \in A_i} \|F_x(t) - y_i\|$$

$$f^+(x) = \sum_{i=1 \dots N} \max_{t \in A_i} \|F_x(t) - y_i\|$$

where $f^-(x) \leq f(x) \leq f^+(x)$ for all x .

The objective of the learning is obtaining the parameter vector x^* that minimizes the error. Observe in

Figure 3 that the most we can know about the value of the objective function in the minimum, $f(x^*)$, is that it is in the segment we have labelled “Best bounds of $f(x^*)$ ”. In turn, x^* is in the area marked “Best bounds of x^* ”. Observe also that the actual value of x^* is not related to the minimum of f^- neither f^+ : we just know that it cannot be out of the mentioned region.

Alternatively, this can be expressed by defining a partial order between fitness values as follows:

$$[a, b] \prec [c, d] \iff b \leq c. \quad (1)$$

If the fitness of the individual x is the interval $[f^-(x), f^+(x)]$ then the best bounds of x^* coincide with the set of all nondominated values of x under that order. This alternative expression is the basis of the use of multicriteria genetic algorithms for learning FRBS with low quality data; those algorithms produce sets of individuals contained in the “Best bounds of x^* ” [17].

Let us introduce now the approximation mentioned in the introduction: let $\mathcal{A}$ be the subset of $\{A_1, \dots, A_n\}$ containing the less informative input intervals. In this paper we want to approximate the fitness function $[f^-, f^+]$ by the pair of functions

$$g^-(x) = \sum_{A_i \in \mathcal{A}^-} \min_{t \in A_i} \|F_x(t) - y_i\| + \sum_{A_i \notin \mathcal{A}^-} \|F_x(\text{mid}(A_i)) - y_i\| \quad (2)$$

$$g^+(x) = \sum_{A_i \in \mathcal{A}^+} \max_{t \in A_i} \|F_x(t) - y_i\| + \sum_{A_i \notin \mathcal{A}^+} \|F_x(\text{mid}(A_i)) - y_i\| \quad (3)$$

where $\mathcal{A} = \mathcal{A}^- \cup \mathcal{A}^+$ and $\text{mid}(A)$ is the center of gravity of the interval A .

We propose to obtain the three unknowns ($\mathcal{A}^-$, $\mathcal{A}^+$ and also the best FRBS) at the same time, with a coevolutionary algorithm based on three populations. The first population contains different model candidates (each individual will represent a model, Pitts style [7]), and the other two codify the sets $\mathcal{A}^-$ and $\mathcal{A}^+$, respectively. In these two last populations, each individual represents one point in the sample, and the whole population is the solution (cooperative approach [6]). In the sections that follow we describe this algorithm in detail, detailing the representation of an individual, the fitness function, the coevolutionary scheme and the genetic operators.

2.1 Representation of an individual

For representing the individuals in the first population we will use the same representation that was defined in [18], that we will not repeat here because of space reasons: DNF rules and a single chromosome with binary codification for representing the linguistic terms

and real coding of the weights. Observe that will not evolve the membership functions, but assign weights to the rules instead [5].

The second population codify as a whole the set $\mathcal{A}^-$ and also the points of these intervals where the error of the FRBS is minimum. An individual is a pair comprising the index s of an interval, $A_s \in \mathcal{A}^-$, and a point $t_s \in A_s$. We have coded this pair with an integer and a real-coded number (s, δ) , where $\delta(t_s) = (t_s - t_s^-)/(t_s^+ - t_s^-)$.

For instance, if we are given a dataset formed by two imprecise tuples $\{(A_1 = [0, 3] \times [1, 2] \times [3, 4], y_1 = 1), (A_2 = [3, 4] \times [1, 1] \times [3, 3], y_2 = 2)\}$, the list $\{1, (0.5, 1, 0.25)\}$ is a valid individual, and it represents a point $(1.5, 2, 3.25) \in [0, 3] \times [1, 2] \times [3, 4]$. The presence of this individual in the second population means that $A_1 \in \mathcal{A}^-$. The codification of the third population is similar.

2.2 Fitness function

The fitness of an individual x in the first population is the interval $[g^-(x), g^+(x)]$, defined in eqs. (2) and (3). For regression problems,

$$\|F_x(t) - y\| = (F_x(t) - y)^2 \quad (4)$$

and for classification problems

$$\|F_x(t) - y\| = \begin{cases} 1 & \text{if } FRBS(t) = y_i \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The fitness of an individual in the first population depends on the other two populations, as follows. Let M be the size of populations 2 and 3, let s_j^- be the first part of j -th element of population 2, and let t_j^- be the value encoded in the second part of the same element. Similarly, let s_j^+ and t_j^+ be the first and second part of an element of the third population. Therefore, $\mathcal{A}^- = \{A_{s_j^-} \mid j = 1, \dots, M\}$ and $\mathcal{A}^+ = \{A_{s_j^+} \mid j = 1, \dots, M\}$. Then the fitness of an individual x is the interval $[g^-(x), g^+(x)]$ defined as follows:

$$g^-(x) = \sum_{j=1}^M \|F_x(t_j^-) - y_{s_j^-}\| + \sum_{\{i \mid A_i \notin \mathcal{A}^-\}} \|F_x(\text{mid}(A_i)) - y_i\| \quad (6)$$

$$g^+(x) = \sum_{j=1}^M \|F_x(t_j^+) - y_{s_j^+}\| + \sum_{\{i \mid A_i \notin \mathcal{A}^+\}} \|F_x(\text{mid}(A_i)) - y_i\|. \quad (7)$$

The fitness value of an individual in the second or the third last two populations is, respectively, the gain or loss in the lower and upper bounds of the error of the elite model, when the point contained in the individual

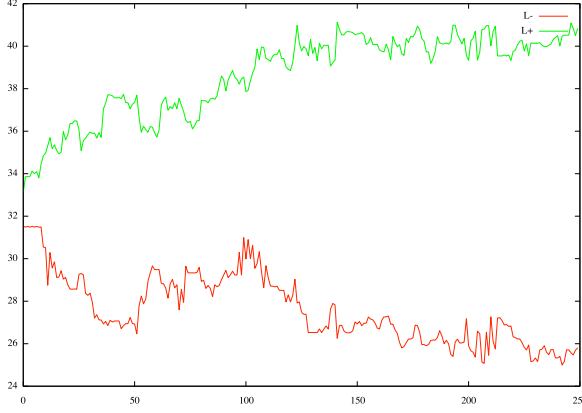


Figure 4: Example run of the GA: Bounds of the error of the best model in the first population when the second and third populations evolve

is replaced by the midpoint of A_s (where s is the index codified in the individual, as mentioned); that is to say, the fitness of the j -th element of the second and third populations are

$$\text{fit}_{j1} = -||\text{FRBS}_{\text{elite}}(t_j^-) - y_{s_j^-}|| + ||\text{FRBS}_{\text{elite}}(\text{mid}(A_{s_j^-})) - y_{s_j^-}|| \quad (8)$$

and

$$\text{fit}_{j2} = ||\text{FRBS}_{\text{elite}}(t_j^+) - y_{s_j^+}|| - ||\text{FRBS}_{\text{elite}}(\text{mid}(A_{s_j^+})) - y_{s_j^+}||. \quad (9)$$

With this definition, the sum of the fitness values of all the individuals in the population equals the difference between the error on the sample comprising the midpoints of the interval-valued training set and the error committed when $\mathcal{A}^-$ and $\mathcal{A}^+$ are codified by populations 2 and 3. The genetic evolution produces, therefore, a cooperative behavior and the populations evolve to encode sets of values with respectively lower and higher errors (see Figure 4 for an actual plot of the bounds of the error of the best model in the first population when the second and third populations evolve).

It is remarked that, in case that an index appears more than once in the same population, the fitness values of all the individuals but the best one must be set to zero, or else the sum of the fitness values is no longer the mentioned difference.

2.3 Coevolutionary scheme

The coevolutionary scheme is as follows:

1. The three populations are initialized with random values.
2. Repeat steps 3 to 9, G_1 times:

3. The fitness of individuals in the first population are computed

4. The first population is ranked by means of a precedence operator between intervals [9]. The elite is copied apart. Tournament selection, crossover and mutation are performed in this population, and the offspring is inserted in place of the worst individuals in the tournament.

5. Repeat steps 6 to 9, G_2 times:

6. Evaluate the fitness of population 2

7. Evaluate the fitness of population 3

8. Assign fitness 0 to duplicates in pop. 2 and 3

9. Crossover and mutation are performed in

these last two populations, and the offspring is inserted back in place (steady state).

2.4 Genetic operators

All algorithms are steady state and based in a tournament selection. The offspring of the winners of the tournament replace the last two elements of the tournament, whose length is used to control the selective pressure.

Standard two-point crossover and mutation are used in the first population. The application of the crossover operator is followed by a Lamarckian local search (Nelder and Mead's algorithm) with certain probability.

The other two populations need custom operators; two individuals (s_1, δ_1) and (s_2, δ_2) are crossed as follows:

- If $s_1 = s_2$, we do an arithmetic crossover between δ_1 and δ_2 [10].
- If $s_1 \neq s_2$, we insert a copy of the best individual and randomly generate the other.

3 Numerical results

This section contains the initial results of the algorithm, when applied to synthetic problems. There are two categories of datasets that should be suitable for a method like this:

1. Data for which the classification rules can be expressed with a compact rulebase: low to moderate number of features, not too complex decision surface.
2. Low quality data: censoring, interval valued and missing features in the data.

To comply with our first requirement, we have built a FRBS comprising 9 rules in a problem with two

	Linear	Quadratic	Neural	KNN	WM	ISH	PM	Crisp FRBS	Interval FRBS
censored - 100	0.492	0.478	0.460	0.448	0.448	0.488	0.478	0.478	0.424
censored - 1000	0.421	0.414	0.424	0.437	0.409	0.413	0.474	0.403	0.402
interval - 100	0.554	0.478	0.490	0.506	0.460	0.478	0.458	0.442	0.432
interval - 1000	0.394	0.397	0.402	0.416	0.450	0.393	0.424	0.351	0.346
missing - 100	0.408	0.372	0.426	0.376	0.364	0.328	0.518	0.330	0.372
missing - 1000	0.416	0.445	0.412	0.461	0.470	0.426	0.456	0.415	0.401

Table 1: Numerical results: Crisp algorithms (LDA and QDA discriminant analysis [4], multilayer perceptron, KNN classifier, Chi [1], Ishibuchi [5], Pal-Mandal [13] and FRBS [12] were compared to Interval-FRBS. The best test results are boldfaced.

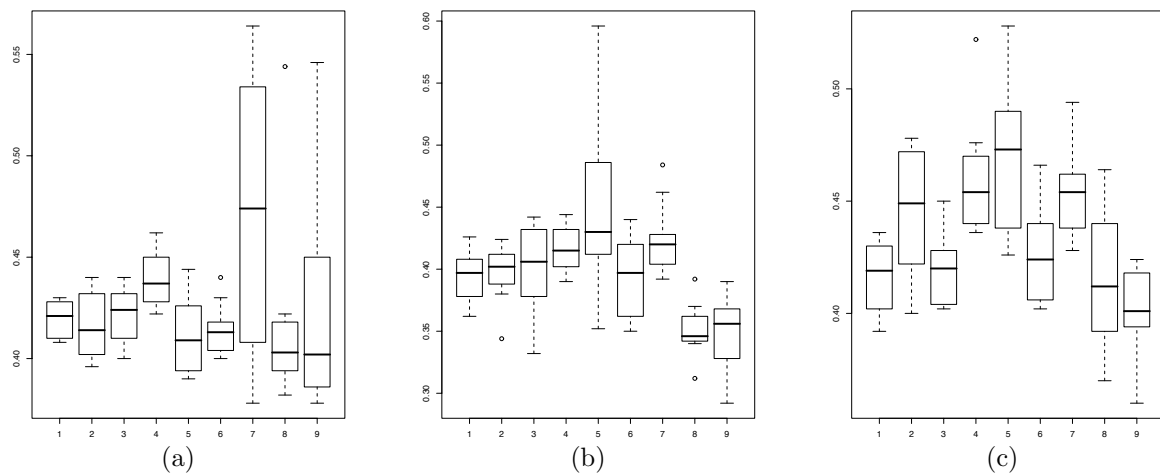


Figure 5: Boxplots showing the dispersion of the results in Figure 1. Censored data (a), interval valued data (c) and missing data (c). The algorithms being compared are in the same order as they appear in Figure 1.

inputs between 0 and 1, and two classes. Since we know the distribution of the population, we have generated datasets whose Bayes error is also known, and for which there exists a FRBS which is the optimal solution. Two datasets of sizes 100 and 1000 were generated.

The second requirement has been fulfilled by adding imprecision to these datasets. We have considered three different categories of imprecision:

1. Censoring: in the 50% of cases, the training data x_s is replaced by the interval $[0, x_s]$. The other cases were replaced by the interval $[x_s, 1]$.
2. Interval valued data: each training data is replaced by the interval $[x_s, x_s + 0.4]$. or $[x_s, 1]$ if $x_s + 0.4 > 1$
3. Missing values: 40% of the points in the training set had one of their features replaced by the interval $[0, 1]$.

These three additions were performed for both datasets, giving the six problems we will use in this section. Other details of the experimental setup are: each experiment has been repeated 10 times, with a 5x2cv experimental design. The size of the first genetic population is 50. Second and third populations have sizes 100 or 1000, depending on the dataset. The number of generations G_1 is 50 and G_2 is 5 (see Section 2.3). The probabilities of crossover and mutation in the first population are 0.7 and 0.1, and the probability of crossover in the second and third populations are equal to 0.9. The probability of the local search is 0.25. The tournament size is 5.

For crisp algorithms (LDA and QDA discriminant analysis [4], multilayer perceptron, KNN classifier, Chi [1], Ishibuchi [5], Pal-Mandal [13] and FRBS [12]), we replaced each interval by its midpoint. We expect that our approach performs the best in all the cases we selected, and also that the final populations 2 and 3 contain the less informative points for the classifier (i.e., those points that, if removed, reduce the most

the width of the interval of likelihoods of the model).

The mean value of the test results are shown in Table 1, and the boxplots depicting the relevance of the differences are displayed in Figure 5. We have obtained the expected results in all cases but one (40% of missing data, datasets of size 100), where the crisp version of the same algorithm improved the results. At the sight of these preliminary results, we think that this algorithm is a promising new technique for exploiting interval data in rule-based classification problems.

4 Concluding remarks

In this paper we have tried a different approach for obtaining linguistically understandable classifiers from interval-valued data. We have defined a coevolutionary scheme able to search in parallel for the best set of rules, and for the two selections of the training set where the lowest and highest errors are reached. These two bounds are used to find a model which is not dominated by other models, and that results in a robust estimation under vague input data. We have checked, with an ad-hoc dataset, that this approach is able to obtain better models than some statistical and fuzzy classifiers, if the conditions are appropriate.

Acknowledgements

This work was supported by the Spanish Ministry of Science and Innovation, under grant TIN2008-06681-C06-04.

References

- [1] Chi, Z., Yan, H., Pham, T. *Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition*. World Scientific. (1996).
- [2] Cordon O, Herrera F, Hoffmann F, Magdalena L, Genetic fuzzy systems. Evolutionary tuning and learning of fuzzy knowledge bases. World Scientific, Singapore (2001)
- [3] Ferson S, Kreainovich V, Hajagos J, Oberkampf W, Ginzburg L, Experimental uncertainty estimation and statistics for data having interval uncertainty. Technical Report SAND2007-0939, Sandia National Laboratories (2007)
- [4] Hand, D. J. *Discrimination and Classification*. Wiley. (1981)
- [5] Ishibuchi, H., Nakashima, T., Murata, T., A fuzzy classifier system that generates fuzzy if-then rules for pattern classification problems. In Proc. of 2nd IEEE CEC, 759–764. (1995)
- [6] Ishibuchi H, Nakashima T, Murata T, Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. IEEE Transactions on Systems, Man and Cybernetics. Part B-Cybernetics 29(5):601-618. (1999).
- [7] De Jong KA, Spears WM, Gordon D.F. Using genetic algorithms for concept learning. Machine Learning 13:161-188. (1993).
- [8] Koeppen, M., Franke, K., and Nickolay, B., Fuzzy-Pareto-Dominance driven multi-objective genetic algorithm. in Proc. 10th International Fuzzy Systems Association World Congress (IFSA), Istanbul, Turkey, 2003: 450–453. (2003).
- [9] Limbourg, P., Multi-objective optimization of problems with epistemic uncertainty. in EMO 2005: 413–427. (2005).
- [10] Michalewicz, Z. Genetic algorithms + Data Structures = Evolution Programs. Springer-verlag, (1992)
- [11] Montes, I., Martinetti, D., Diaz, S., Montes, S. Statistical Preference as a comparison method of two imprecise fitness values. ESTYLF 2010. To appear. (2010)
- [12] Otero J., Sanchez, L. (2006) Induction of descriptive fuzzy classifiers with the Logitboost algorithm Soft Computing 10 (9): 825–835
- [13] Pal, S. K., Mandal, D. P. “Linguistic recognition system based in approximate reasoning”. *Information Sciences* **61**, pp. 135-161. (1992).
- [14] Potter, M., De Jong, K. A cooperative coevolutionary approach to function optimization. Lecture Notes in Computer Science 866. 249-257. 2006.
- [15] Ratschek, H. Some recent aspects of interval algorithms for global optimization. In: Moore RE (ed) Reliability in Computing: The Role of Interval Methods in Scientific Computing. Acad. Press, New York, pp 325? 339. (1988).
- [16] Sánchez L., Couso I. Advocating the use of imprecisely observed data in genetic fuzzy systems IEEE Transactions on Fuzzy Systems 15 (4): 551–562. (2007)
- [17] Sánchez, L., Couso, I., Casillas, J. Modeling vague data with genetic fuzzy systems under a combination of crisp and imprecise criteria. First IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM 2007). Honolulu, Hawaii, USA, (2007).
- [18] Sánchez, L., Couso, I., Casillas, J. Genetic learning of fuzzy rules based on low quality data. Fuzzy Sets and Systems. 160 (17) 2524-2552. (2009)

Extracción de conocimiento a partir de reglas difusas ponderadas que soportan datos imprecisos

Ana M. Palacios

Departamento de Informática, Universidad de Oviedo, Gijón, Asturias, palaciosana@uniovi.es

Resumen

En trabajos anteriores, hemos propuesto una estrategia cooperativa-competitiva, basada en una función de fitness borroso-valorada, para obtener clasificadores basados en reglas borrosas a partir de datos imprecisos. Este esquema tiene un buen funcionamiento, en general, si bien la evaluación del fitness en los valores con salida imprecisa depende de un algoritmo en que cada instancia se ha de replicar un número de veces proporcional al producto del número de alternativas posibles por cada dato. En este trabajo desarrollamos una variante de este algoritmo donde la replicación de las instancias se reemplaza por una asignación de pesos a cada ejemplo y cada una de las reglas tendrá asociada, de forma heurística, una seguridad en su consecuente. La evaluación del clasificador propuesto se realiza sobre dos problemas reales, el diagnóstico de la dislexia y el rendimiento de atletismo.

Palabras Clave: Datos imprecisos, reglas borrosas, pesos, algoritmo genético, GCCL, dislexia

1. Introducción

La extensión de los Genetic Fuzzy Systems (GFS) a conjuntos de datos no numéricos es una línea de investigación activa [12]. En particular, el uso de datos de baja calidad (incompletos o imprecisos) permitirá aprovechar mejor la información recogida en ciertos procesos en que los datos tienen asociada una cierta incertidumbre, ya sea de forma inherente (como ocurre, por ejemplo, durante la digitalización de información analógica) o por aplicación de una interpretación

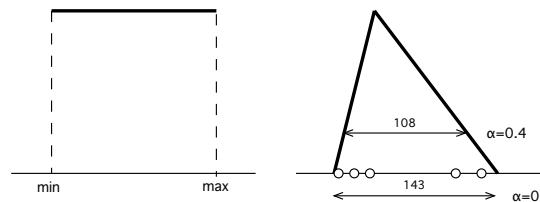


Figura 1: Representación borrosa de datos imprecisos. Izquierda: Valor desconocido de una variable. Derecha: Variable definida por cinco medidas distintas.

semántica de un conjunto borroso y la subsiguiente conversión de un agregado de datos en un conjunto borroso (como sucede por ejemplo en algunos tipos de cuestionario; ver [13] y figura 1).

El primer problema que aparece en la extensión de un GFS a datos de baja calidad es el de la obtención de la salida del GFS cuando la entrada del objeto es imprecisa. En este estudio nos basaremos en el método propuesto en [13] y adaptado en [10], donde la salida obtenida por el clasificador, a partir de una entrada imprecisa y un conjunto de reglas con un único consecuente, es un conjunto de clases. Supongamos que disponemos de un clasificador borroso compuesto por M reglas del estilo:

$$\text{Si } x \text{ es } \tilde{A}_i \text{ entonces la clase es } C_i \quad (1)$$

y la clase que se le asignaría a un objeto x es:

$$\text{class}(x) = C_{\arg \max_i \{\tilde{A}_i(x)\}}. \quad (2)$$

La extensión al caso en que 'x' sea imprecisa y toda la información de que disponemos es " $x \in X$ ", consiste en que la clase del objeto es el conjunto de valores [8][9][10]

$$\text{class}(X) = \{C_{\arg \max_i \{\tilde{A}_i(x)\}} \mid x \in X\} \quad (3)$$

o lo que es lo mismo:

$$\text{class}(X) = \{\text{class}(x) \mid x \in X\}. \quad (4)$$

En este trabajo extenderemos el cálculo anterior mediante la asignación de un grado de certeza a los consecuentes de las reglas, y una penalización el fitness de aquellas reglas que sean falsas compatibles, que redundará en un beneficio en la capacidad de descubrimiento de nuevas reglas. Se comprobará que el uso de reglas ponderadas y el tratamiento de las reglas no dominadas influye significativamente en la evolución del aprendizaje del clasificador propuesto.

Esta nueva extensión del GFS se organiza de la siguiente manera: En la sección 2 se muestra el método aplicado para obtener el peso en las reglas lingüísticas que soportan datos imprecisos. En la sección 3 describiremos la extensión realizada para obtener el clasificador propuesto destacando la penalización de las reglas. En la sección 4 aplicaremos el algoritmo propuesto a dos problemas reales y compararemos los resultados obtenidos con los de [10]. Por último, en la sección 6 resaltaremos las conclusiones finales.

2. Reglas Lingüísticas Ponderadas

Emplearemos reglas con un único consecuente y con un grado de certeza en el consecuente, como se ha mencionado:

$$\text{Si } x_1 \text{ es } \tilde{A}_{i1} \text{ y } \dots \text{ y } x_n \text{ es } \tilde{A}_{in} \text{ entonces la clase es } C_i \text{ con } CF_i, \quad (5)$$

donde $x = (x_1, \dots, x_n)$ son las entradas, A_{i1} son los antecedentes borrosos del atributo, C_i es el consecuente o salida de la regla y CF_i es el peso de la regla. El peso de la regla viene definido por un valor real comprendido entre $[0,1]$. Por lo tanto, si el peso de una regla tiene el valor 0 implica que la regla no tiene efecto en la clasificación. Para obtener el peso de una regla, extenderemos los criterios heurísticos de Ishibuchi [5], basados en las definiciones de confianza y soporte de una regla borrosa, extendidas a datos imprecisos de la forma que se muestra a continuación.

La confianza de una regla borrosa $c(A_i \Rightarrow C_i)$ viene definida por la siguiente expresión [5][3]:

$$c(A_i \Rightarrow C_i) = \sum_{x_p \in \text{Class } C_i} \mu_{A_i}(x_p) / \sum_{p=1}^m \mu_{A_i}(x_p) \quad (6)$$

Como las compatibilidades de las reglas son imprecisas, la expresión de confianza se ve modificada siguiendo la siguiente expresión:

$$c(A_i \Rightarrow C_i) = \sum_{x_p \in \text{Class } C_i} \mu_{A_i}(x_p) \odot n \quad (7)$$

donde n es el número de ejemplos x_p compatibles con el antecedente A_i . Con (7) todavía seguimos teniendo

una confianza imprecisa y por tanto, un peso impreciso. Por esta razón, calcularemos la distancia existente entre el valor máximo posible de la confianza max_{conf} (todos los x_p tienen 1 como grado de compatibilidad con el antecedente A_i) y la confianza de la regla. Por todo ello, la primera definición del peso de la regla se define como:

$$CF_i^I = 1 - d_H(max_{conf}, c(A_i \Rightarrow C_i)) = \min(c(A_i \Rightarrow C_i)) \quad (8)$$

La confianza así definida puede emplearse directamente como peso de una regla. Siguiendo el trabajo [5], también es razonable emplear como peso

$$CF_i^{II} = c(A_i \Rightarrow C_i) - c_{Average} \quad (9)$$

y en [6] se proponen otras dos definiciones:

$$CF_i^{III} = c(A_i \Rightarrow C_i) - c_{2nd}, \quad (10)$$

$$CF_i^{IV} = c(A_i \Rightarrow C_i) - c_{Sum}, \quad (11)$$

que generalizaremos a su vez de la forma que sigue:

$$CF_i^{II} = d_H(c(A_i \Rightarrow C_i), c_{Average}) \quad (12)$$

3. Extensión del GFS con penalización

Las extensiones del algoritmo introducido en [8][9][10] se describen a continuación. En resumen, éstas consisten en:

1. Un nuevo procedimiento para la asignación del consecuente a las reglas difusas.
2. Un nuevo procedimiento para la asignación del fitness a las reglas, destacando la selección de las reglas no dominadas y compatibles con el ejemplo actual, así como el tratamiento de las reglas no dominadas mediante la selección de la regla que va a ser puntuada y la penalización de la misma.
3. La selección y reemplazamiento de los mejores y peores individuos cuando el fitness de éstos es un conjunto de valores.

3.1. Asignación del consecuente

La función original de la asignación del consecuente consiste en calcular la confianza de las reglas con cada uno de los posibles consecuentes y después seleccionar el consecuente con más confianza. En esta extensión la confianza de la una regla vendrá definida por un conjunto de valores. Debido a esto, tendremos que seleccionar el consecuente a partir del conjunto de confianzas no dominadas [10]. Para ello, utilizamos la dominancia uniforme definida en [7].

3.2. Asignación y computación del fitness

La asignación de fitness en [4] se basa en puntuar únicamente a una regla, la regla ganadora. En este procedimiento la selección de la regla ganadora se basa en obtener una regla dentro de las reglas no dominadas [10] obtenidas según la dominancia uniforme definida en [7]. La selección de dicha regla y la puntuación de la misma influyen en la selección de futuras reglas ya que, dicha puntuación no solamente tiene en cuenta si la clase del clasificador coincide o no con la clase del objeto, sino que penaliza a las reglas que son falsas compatibles refinando la selección de las reglas.

La penalización de las reglas falsas compatibles viene definida por:

$$P_r = F_r \ominus (\eta^- \otimes F_r) \quad (13)$$

donde η^- es una constante positiva ($0 < \eta^- < 1$) y F_r representa el fitness actual de la regla r .

Como se ha mencionado anteriormente, la salida obtenida por el sistema borroso, a partir de información imprecisa de un ejemplo, es un conjunto de clases. Esto implica que la función de fitness es definida por un conjunto de valores. Es decir, si la salida del clasificador es una única clase y esta coincide con la clase del ejemplo entonces, puntuaremos a la regla ganadora, r , con 1 punto. Si la salida del clasificador es un conjunto de clases y la intersección de éstas con las clases del objeto es vacía entonces, penalizaremos a la regla P_r ya que se trata de una regla falsa compatible. En otro caso, la puntuación es el conjunto definido por $\{P_r, 1\}$.

Para la evaluación exhaustiva del FRBS obtenido por el clasificador, utilizamos la función mostrada en la figura 2. Dicha función es computacionalmente muy costosa, por lo que durante el aprendizaje del clasificador emplearemos la aproximación introducida en [8].

3.3. Selección y reemplazamiento genético

La función de fitness definida por un valor impreciso implica varios cambios respecto al algoritmo original. El primero de ellos, es la selección de los mejores individuos en el torneo, que vienen condicionados por el valor del fitness. Lo mismo ocurre con el reemplazamiento de los peores individuos. En ambos casos utilizamos de nuevo la dominancia uniforme definida en [7].

4. Experimentos

En esta sección mostramos los resultados numéricos obtenidos sobre el algoritmo propuesto. Para poder llevar a cabo dicha experimentación disponemos de data-

```

function FitnessExhaustivoTest(population,dataset)
1  for dataset in {1,...,1000}
2    fitness[dataset] = 0
3    for example in {1,...,N}
4      bestMatch = 0
5      WRule = -1
6      for r in {1,...,M}
7        m = membership(Antecedent[r],example)*CFr
8        if (m > bestMatch) then
9          WRule = r
10         bestMatch = m
11       end if
12     end for r
13     if (WRule == -1) then
14       WRule = rule_fre_class
15     end if
16     if (consequent(WRule) == class(example)) then
17       score = 1
18     else
19       if consequent(WRule) ⊂ class(example) then
20         score = {0,1}
21       end if
22     end if
23     fitness[dataset] = fitness[dataset] ⊕ score
24   end for example
25 end for dataset
26 fitness=0
27 for dataset in {1,...,1000}
28   fitness=fitness ⊕ fitness[dataset]
29 end for dataset
30 fitness=mean(fitness)
return fitness

```

Figura 2: Función para la evaluación exhaustiva del FRBS obtenido por el clasificador.

sets que contienen información imprecisa [10]. Hemos considerado dos tipos de problemas reales:

1. El diagnóstico de la dislexia [15] descrito en [10], “Dyslexic-12”. A este problema de la dislexia se ha insertado un nuevo dataset aportado por un psicólogo denominado “Expert-11-01”.
2. Rendimientos en las pruebas de atletismo descrito en [9].

Todos los experimentos se han realizado con una población de 100 individuos, una probabilidad de cruce y mutación de 0.9 y 0.1, respectivamente y 200 generaciones. El numero de particiones se indican en cada uno de los datasets utilizados. Nuestro diseño experimental está basado en una evaluación bootstrap que utiliza 1000 pruebas por cada partición de test, según se muestra en la figura 2.

Todos los dataset utilizados en este artículo están disponibles en la página web del proyecto KEEL: <http://www.keel.es>.

Cuadro 1: Representación de la media de 10 repeticiones en los problemas “Dyslexic-12” y “Dyslexic-11-01”.

Dataset	Crisp		Imprecisos	
	Train	Test	Exh.Test EVIN	Exh.Test Pon/Pen
Dyslexic-12 (4 labels) CF_i^0	0.541	0.657	[0.421, 0.558]	[0.437, 0.556]
Dyslexic-12 (4 labels) CF_i^I				[0.470,0.585]
Dyslexic-12 (4 labels) CF_i^{II}				[0.423,0.538]
Dyslexic-12 (4 labels) CF_i^{III}				[0.456,0.541]
Dyslexic-12 (4 labels) CF_i^{IV}				[0.428,0.524]
Dyslexic-12 (5 labels) CF_i^0	0.672	0.694	[0.490, 0.609]	[0.480, 0.593]
Dyslexic-12 (5 labels) CF_i^I				[0.488,0.606]
Dyslexic-12 (5 labels) CF_i^{II}				[0.488,0.603]
Dyslexic-12 (5 labels) CF_i^{III}				[0.492,0.589]
Dyslexic-12 (5 labels) CF_i^{IV}				[0.492,0.588]
Expert-11-01 (4 labels) CF_i^0	0.421	0.569	[0.508, 0.683]	[0.362, 0.492]
Expert-11-01 (4 labels) CF_i^I				[0.362,0.501]
Expert-11-01 (4 labels) CF_i^{II}				[0.354,0.500]
Expert-11-01 (4 labels) CF_i^{III}				[0.367,0.497]
Expert-11-01 (4 labels) CF_i^{IV}				[0.362,0.491]
Expert-11-01 (5 labels) CF_i^0	0.528	0.586	[0.500, 0.656]	[0.405, 0.543]
Expert-11-01 (5 labels) CF_i^I				[0.405,0.543]
Expert-11-01 (5 labels) CF_i^{II}				[0.405,0.543]
Expert-11-01 (5 labels) CF_i^{III}				[0.405,0.539]
Expert-11-01 (5 labels) CF_i^{IV}				[0.390,0.525]

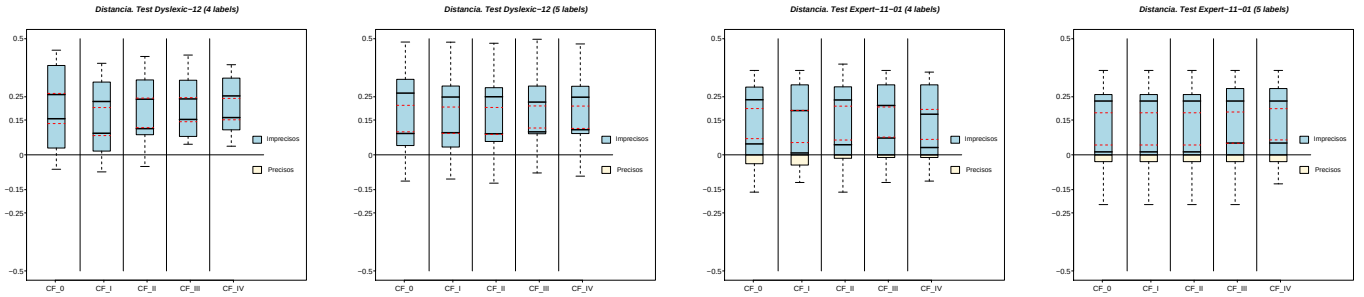


Figura 3: Representación de la distancia en los problemas “Dyslexic-12” y “Dyslexic-11-01” con 4/5 etiquetas.

4.1. Transformación de dataset imprecisos a precisos

La comparación entre dataset precisos e imprecisos es difícil de realizar. No conocemos trabajos anteriores en este campo. Por ese motivo, utilizamos las condiciones descritas en [10] para poder comparar dataset precisos con imprecisos.

4.2. Representación de los resultados

Para realizar la representación de los resultados obtenidos con el algoritmo original [4], y la mejora del algoritmo propuesto respecto al algoritmo realizado en [10], utilizamos dos maneras:

1. Tablas: Representación de la media de los resultados obtenidos en 10 repeticiones. En la columna

“Crisp” representamos los resultados del entrenamiento y la evaluación del GFS original [4]. En la columna “Imprecisos” mostramos los resultados obtenidos en la evaluación del algoritmo propuesto en [10] y del propuesto en este trabajo, columnas “Exh.Test EVIN” y “Exh.Test Pon/Pen”, respectivamente.

2. Boxplot: Los boxplot utilizados para comparar los resultados numéricos no son estándar (ver [10]) Observe que, en el boxplot, la zona inferior a cero representa que la distancia es favorable a los datos precisos, es decir, se obtienen mejores resultados con el algoritmo original, y la zona superior a cero representa que la distancia es favorable al algoritmo propuesto (ver figura 3).

Cuadro 2: Representación de la media de 10 repeticiones para los problemas “Long-4”, “100ml-4-I” y “100ml-4-P”.

Dataset	Crisp		Imprecisos	
	Train	Test	Exh.Test EVIN	Exh.Test Pond/Pen
Long-4 (4 labels) CF_i^0	0.308	0.473	[0.266, 0.533]	[0.308, 0.522]
Long-4 (4 labels) CF_i^I				[0.308 ,0.507]
Long-4 (4 labels) CF_i^{II}				[0.322,0.527]
Long-4 (4 labels) CF_i^{III}				[0.302 , 0.508]
Long-4 (4 labels) CF_i^{IV}				[0.330,0.546]
100ml-4-I (5 labels) CF_i^0	0.259	0.384	[0.189, 0.476]	[0.162, 0.361]
100ml-4-I (5 labels) CF_i^I				[0.168,0.368]
100ml-4-I (5 labels) CF_i^{II}				[0.173,0.376]
100ml-4-I (5 labels) CF_i^{III}				[0.134 , 0.335]
100ml-4-I (5 labels) CF_i^{IV}				[0.174,0.377]
100ml-4-P (5 labels) CF_i^0	0.288	0.419	[0.177, 0.406]	[0.177, 0.406]
100ml-4-P (5 labels) CF_i^I				[0.192,0.423]
100ml-4-P (5 labels) CF_i^{II}				[0.196,0.416]
100ml-4-P (5 labels) CF_i^{III}				[0.170 , 0.385]
100ml-4-P (5 labels) CF_i^{IV}				[0.177 , 0.392]

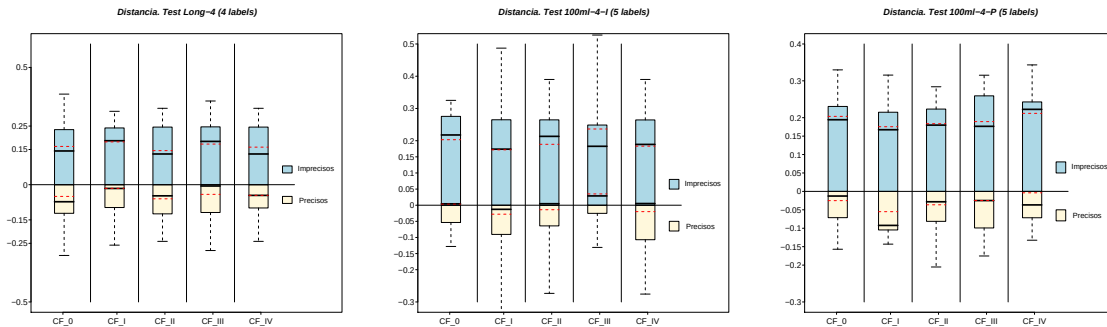


Figura 4: Representación de la distancia en los problemas “Long-4”, “100ml-4-I” y “100ml-4-P”.

4.3. Datasets relacionados con el diagnóstico de la dislexia

Los dataset utilizados en esta sección son:

- “Dyslexic-12” [8][10], está compuesto por 65 instancias, 4 clases y 12 características. Este dataset ha sido aportado por un experto en psicología cuando realiza el diagnóstico de un niño. Contiene datos desconocidos y las entradas y salidas son imprecisas.
- “Expert-11-01”, está compuesto por 65 instancias, 3 clases, 11 características, con datos desconocidos, entradas precisas y salidas imprecisas. En este nuevo dataset propuesto, “Expert-11-01”, el experto elimina la imprecisión de las entradas para evaluar al niño.

En la tabla 1 observamos como los resultados del algoritmo propuesto son mejores con respecto al resto

de algoritmos. Además, observamos que la inserción de reglas ponderadas hace más eficiente y robusto el clasificador. En la figura 3 la distancia representada es a favor del algoritmo propuesto lo que implica que se obtienen mejores resultados cuando trabajamos con datos imprecisos.

4.3.1. Datasets relacionados con el rendimiento de un equipo de atletismo

Los dataset utilizados en esta sección son:

- “Long-4” descrito en [9], determina si un atleta es relevante o no en la prueba de longitud.
- “100ml4-I” descrito en [9], determina si un atleta es relevante o no en la prueba de 100 metros lisos.
- “100ml4-P” descrito en [9], determina si un atleta es relevante o no en la prueba de 100 metros lisos. Este dataset difiere del anterior en que el entrenador no solamente se basa en los valores de los

indicadores de la prueba sino también aporta su conocimiento personal acerca del atleta.

Para obtener dichos datasets se han utilizado a 25 atletas lo que implica un conjunto de 25 instancias, con 4 características, 2 clases y sin valores desconocidos, donde las entradas y salidas son imprecisas.

Los resultados numéricos de dichos dataset se muestran en la tabla 2 y su representación gráfica sobre la distancia entre los datos precisos e imprecisos en la figura 4.

5. Conclusiones

La extensión del GFS para soportar datos imprecisos con reglas ponderadas y penalización de las mismas aporta una mejora de rendimiento en el clasificador. Esto se debe a la influencia que se ejerce sobre la selección de futuras reglas y a que, por tanto, pueda hacerse frente a las reglas que son ineficientes y falsas compatibles. En consonancia con esta extensión, los resultados numéricos obtenidos mejoran los resultados de trabajos previos.

Agradecimientos

Este trabajo está soportado por el Ministerio de Educación y Ciencia de España, TIN2008-06681-C06-04 y por el Principado de Asturias, PCTI 2006-2009.

Referencias

- [1] Agrawal R., Srikant R., Fast algorithms for mining association rules, Proc. of 20th International Conference on Very Large Data Bases, pp. 487-499, Santiago, September 1994. Expanded version is available as IBM Research Report RJ9839, June 1994
- [2] Couso, I., Sánchez, L. Higher order models for fuzzy random variables. Fuzzy Sets and Systems 159: pp 237-258 (2008).
- [3] Hong T., Kuo C., Chi S., Trade-off between computation time and number of rules for fuzzy mining from quantitative data. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 9, no. 5, pp. 587-604, October 2001.
- [4] Ishibuchi, H., Nakashima, T., Murata, T, A fuzzy classifier system that generates fuzzy ifthen rules for pattern classification problems. In Proc. of 2nd IEEE International Conference on Evolutionary Computation, 759-764 (1995)
- [5] Ishibuchi, H., Takashima, T., Effect of rule weights in fuzzy rule-based classification systems. IEEE Transactions on Fuzzy Systems, 3(3),260-270, 2001.
- [6] Ishibuchi, H., Yamamoto, T., Rule weight specification in fuzzy rule-based classification systems. IEEE Transactions on Fuzzy Systems, 13(4),260-270, 2005.
- [7] Limbourg, P., Multi-objective optimization of problems with epistemic uncertainty. in EMO 2005: 413-427. (2005)
- [8] Palacios, A., Sánchez, L., Couso, I. A baseline genetic fuzzy classifier based on low quality data. IFSA-EUSFLAT 2009.
- [9] Palacios, A., Sánchez, L., Couso, I. GFS-based analysis of vague databases in High Performance Athletics. IDEAL 2009.
- [10] Palacios, A., Sánchez, L., Couso, I. Extending a simple Genetic Cooperative-Competitive Learning Fuzzy Classifier to low Quality datasets. EVIN-D-09-00032R1.
- [11] Rivas, R.M., Fernández P., Dislexia, disortografía y disgrafía. Madrid: Pirámide (1994).
- [12] Sánchez L., Couso I. Advocating the use of imprecisely observed data in genetic fuzzy systems IEEE Transactions on Fuzzy Systems 15 (4): pp 551-562. (2007).
- [13] Sánchez, L., Couso, I., Casillas, J. Genetic learning of fuzzy rules based on low quality data. Fuzzy Sets and Systems. 160 (17) pp 2524-2552, (2009)
- [14] Sánchez, L., Otero, J., Couso, I. Obtaining linguistic fuzzy rulebased regression models from imprecise data with multiobjective genetic algorithms. Soft Computing 13 (5) pp 467-479, (2009)
- [15] Sánchez, L., Palacios, A., Couso, I., A Minimum Risk Wrapper Algorithm for Genetically Selecting Imprecisely Observed Features, applied to the Early Diagnosis of Dyslexia. Lecture Notes in Computer Science 5271, pp 608-615 (2008)

CÓMO HACER INFERENCIA ESTADÍSTICA A PARTIR DE DATOS DE BAJA CALIDAD

Inés Couso

Dep. Estadística e I.O. y D.M.

Universidad de Oviedo

e-mail: couso@uniovi.es

Resumen

En problemas de aprendizaje de máquina, si el conjunto de entrenamiento no es conocido de forma precisa, tampoco es posible determinar de forma exacta el error de un modelo. Por consiguiente, la comparación estadística entre diferentes algoritmos de aprendizaje no puede resolverse con los diseños experimentales habituales. Este trabajo estudia las extensiones de la inferencia estadística y las limitaciones de la toma de decisiones a los problemas con datos de baja calidad.

Palabras Clave: Datos de baja calidad, Diseño experimental, Inferencia estadística.

1. INTRODUCCIÓN

La última fase del diseño de un algoritmo de aprendizaje es la validación estadística de sus resultados. Los diseños experimentales más extendidos emplean tests para contrastar la precisión, interpretabilidad, velocidad de ejecución, u otro índice de calidad del algoritmo.

En general, dichos índices son atributos numéricos. Sin embargo, cuando los datos de entrenamiento son intervalares, o borrosos, los índices con los que se mide la calidad de un modelo son, a su vez, intervalos de valores o números borrosos. No tenemos constancia de que, hasta el momento, se haya detallado una metodología de comparación de algoritmos válida para este caso. Por esta razón, en este trabajo se estudiarán los procesos de inferencia estadística necesarios para comparar los resultados de los modelos que se obtienen cuando los datos de aprendizaje son de baja calidad.

Nuestra metodología parte de la base de que se debe utilizar toda la información disponible, pero sin añadir ningún dato de forma artificial. El hecho de añadir in-

formación artificial, para poder aprovechar técnicas estadísticas ya existentes, puede hacernos incurrir en conclusiones erróneas. No obstante, a lo largo de este trabajo haremos las siguientes consideraciones:

- Un atributo sigue cierta distribución de probabilidad en una población.
- Dicha distribución es conocida salvo por el valor de un parámetro (esto es, no consideraremos tests no paramétricos).
- Se dispone de n intervalos que representan la información (imprecisa) disponible acerca del valor del atributo en esas n mediciones (tal y como ocurriría, por ejemplo, si se hubiese realizado un diseño experimental basado en n -cv).

Para aclarar los conceptos introducidos en el trabajo utilizaremos ejemplos que involucren cálculos sencillos. Aunque no son ejemplos reales, los utilizamos deliberadamente, para que el lector no desvíe su atención hacia el problema de la complejidad del cálculo numérico. Por último, indicamos que en este estudio nos restringiremos a aquel caso en que nuestra información muestral viene dada por una colección de intervalos. La extensión al caso difuso es posible, pero por razones de espacio no se ha incluido aquí.

La estructura de este trabajo es como sigue: en la sección 2, explicaremos los dos tratamientos teóricos existentes en la literatura para los datos intervalares, y mostraremos cuál de estos tratamientos se corresponde con nuestra noción de datos de baja calidad. En la sección 3, explicamos cómo debe extenderse la noción clásica de test a nuestro contexto. En la sección 4, trataremos el problema de la comparación de la potencia de distintos tests. El trabajo termina con las conclusiones y el trabajo futuro en la sección 5.

2. DOS INTERPRETACIONES DE LOS DATOS INTERVALARES

A lo largo de esta sección, partiremos de la información que nos proporciona la observación (incompleta) en dos muestras apareadas. Así, nuestras entradas serán una colección de n parejas de intervalos, tal como se muestra en el cuadro 1:

Cuadro 1: Información muestral

muestra 1	$[l_1, u_1]$	...	$[l_n, u_n]$
muestra 2	$[l'_1, u'_1]$	...	$[l'_n, u'_n]$

Hay dos grandes grupos de técnicas estadísticas aplicables a datos intervalares: aquellas en las que se asume que el intervalo representa información imprecisa acerca de un valor real conocido parcialmente y aquellas en las que se asume que las salidas del experimento aleatorio son realmente intervalos. Ambas interpretaciones son válidas, pero cada una es aplicable en un contexto distinto. Con la ayuda de dos ejemplos sencillos, mostraremos la utilidad de cada una. En ambos ejemplos, partiremos de la misma información muestral, pero el tratamiento de los datos deberá ser diferente.

Ejemplo 2.1 *Se pretende comparar las temperaturas ambientales de las localidades de Huelva y Adeje (sur de Tenerife) durante los meses de mayo de los últimos 25 años. Con ese fin, se han tomado los datos del histórico para 36 días elegidos al azar. En el cuadro 2 se muestran los intervalos $[T_{min}, T_{max}]$ en ambas localidades.*

Cuadro 2: Temperaturas máximas y mínimas.

Huelva	[15,25]	[15,25]	[16,25]	[18,27]
Adeje	[19,26]	[20,25]	[21,26]	[21,26]
nº días	10	5	12	9

Denotemos por μ_{mH} la temperatura mínima media (esperanza de las mínimas) y por μ_{MH} la temperatura máxima media en la ciudad de Huelva. De forma análoga, denotemos por μ_{mA} y por μ_{MA} , respectivamente, a las cantidades análogas en la localidad de Adeje. Nuestro objetivo es contrastar si los intervalos $[\mu_{mH}, \mu_{MH}]$ y $[\mu_{mA}, \mu_{MA}]$ son similares. Para hacer un contraste de este tipo, un procedimiento válido¹ sería el siguiente: en primer lugar, elegimos una métrica adecuada pa-

¹En la literatura reciente podemos encontrar algunos trabajos dedicados a la determinación de tests para problemas en que los datos muestrales son intervalos interpretados de la forma indicada aquí. En todos ellos está implícita la idea

ra cuantificar distancias entre intervalos. Por ejemplo, podemos elegir la distancia:

$$d([a, b], [c, d]) = \sqrt{\frac{(c-a)^2 + (d-b)^2}{2}}.$$

que pertenece a la familia de distancias propuesta por Bertoluzza et al. en [1]. A partir de los datos anteriores, las distancias entre intervalos son:

Cuadro 3: Distancias entre intervalos

distancia	$\sqrt{5}$	$\sqrt{8,5}$	$\sqrt{12,5}$	$\sqrt{13}$
nº días	9	10	5	12

A continuación, se contrastaría si el valor esperado de las distancias es 0 o no. A partir de los datos recogidos, rechazaríamos la hipótesis nula (omitiremos los cálculos necesarios para llegar a esa conclusión). En efecto, aunque las temperaturas máximas en mayo son parecidas en ambas localidades, las temperaturas mínimas no lo son, ya que el clima en el sur de Tenerife es más suave.

Ejemplo 2.2 *Ahora pretendemos contrastar si las temperaturas ambientales de ambas localidades a las 16:00 h (horas locales) son similares o no. Para los mismos 36 días del ejemplo anterior, se han recogido las parejas de temperaturas a esa hora en ambas localidades. Los datos se presentan en el cuadro 4:*

Cuadro 4: Temperaturas a las 16:00h.

Huelva	23	24	25	25	26
Adeje	24	25	24	25	25
nº días	4	13	11	4	4

La muestra de las diferencias entre las temperaturas de ambas localidades en los 36 días señalados se muestra en el cuadro 5.

Cuadro 5: Diferencias a las 16:00h

diferencias	-1	0	1
nº días	17	4	15

No podemos descartar la hipótesis (omitimos los cálculos, por falta de espacio) de que esta muestra proviene de una variable aleatoria con media nula. Por

de contrastar simultáneamente la igualdad entre las esperanzas de los extremos de los respectivos intervalos ($H_0 : \mu_{mH} = \mu_{mA}$ y $\mu_{MH} = \mu_{MA}$), o, de forma equivalente, contrastar la igualdad entre las esperanzas de los puntos medios y de las amplitudes ($H_0 : \frac{\mu_{mH} + \mu_{MH}}{2} = \frac{\mu_{mA} + \mu_{MA}}{2}$ y $\frac{\mu_{MH} - \mu_{mH}}{2} = \frac{\mu_{MA} - \mu_{mA}}{2}$). Está fuera de los propósitos de nuestro trabajo hacer un análisis exhaustivo de estos procedimientos.

tanto, si denotamos respectivamente por μ_1 y μ_2 a las temperaturas medias en los días de mayo a las 16:00 h en ambas localidades, no rechazaremos la hipótesis nula $H_0 : \mu_1 = \mu_2$ frente a la alternativa $H_0 : \mu_1 \neq \mu_2$.

Supongamos ahora, que una persona debe contrastar ambas hipótesis, pero que no dispone de los datos del cuadro 5, sino sólo de las temperaturas máximas y mínimas diarias recogidas en el cuadro 2. Así que su única información acerca de la temperatura a las 16:00 h en un día concreto es que se encuentra entre los valores máximo y mínimo registrados. (Supongamos que no tiene ninguna información adicional acerca de la relación entre la temperatura a las 16:00 h y las temperaturas máxima y mínima del día). Con una información tan imprecisa, la persona no puede contrastar las hipótesis anteriores. En efecto, para la muestra real de las temperaturas (que él desconoce), no rechazaríamos la hipótesis nula. Sin embargo, la muestra siguiente también es compatible con los datos del cuadro 2:

Cuadro 6: Muestra compatible alternativa

Huelva	15	15	16	18
Adeje	26	25	26	26
nº días	10	5	12	9

Para esta muestra de datos sí rechazaríamos la hipótesis H_0 .

En nuestro contexto, cada intervalo refleja, al igual que en este último ejemplo, nuestra información (incompleta o imprecisa) acerca de un número. El hecho de que las amplitudes y/o los puntos medios de los intervalos de ambas muestras sean, en general, diferentes, no implica que las diferencias entre los valores reales sean significativas. De forma análoga, si la imprecisión es grande, el hecho de que los intervalos de ambas muestras sean parecidos no implica que los datos reales provengan de poblaciones similares. Para tomar decisiones a partir de datos de baja calidad, se considera un test determinado y se calcula su salida (1=rechazo / 0=no rechazo) para cada colección de n parejas compatible con la información disponible en el cuadro 2. Nos encontraremos en alguno de los tres casos siguientes:

- Para todas ellas, el test indica “rechazar”. (Rechazaremos H_0).
- Para todas ellas, el test indica “no rechazar”. (No rechazamos la hipótesis nula).
- Para algunas muestras, el test indica “rechazar” y para otras, “no rechazar”. En ese caso, la imprecisión de los datos nos impide tomar una decisión

clara, porque no se dispone de información suficiente.

En la sección 3, introduciremos algunas definiciones formales en relación con este problema.

2.1. Aplicación a la verificación en aprendizaje de máquina

Cabe preguntarse cuál de los dos enfoques es el más adecuado para la comparación de algoritmos de aprendizaje a partir de datos de baja calidad. Como hemos mencionado, en la validación de un algoritmo de aprendizaje se dispone de una muestra de n intervalos, cada uno de los cuales restringe los valores posibles del atributo que se desee contrastar. Con frecuencia, se busca un test que permita rechazar que los valores medios de los atributos obtenidos para dos algoritmos sean iguales, y por esta razón la segunda técnica es la más adecuada; si se emplease la primera, se estaría contrastando si las medias de los márgenes superiores e inferiores de los valores del atributo coinciden. En general, esta última información no es necesaria para la validación.

3. LIMITACIONES EN LA TOMA DE DECISIONES

En esta sección, vamos a mostrar una definición formal que generaliza el concepto de *test* al caso que nos ocupa. Antes debemos recordar algunas nociones de la estadística clásica. Consideremos una muestra aleatoria simple $(\mathbf{X}, \mathbf{Y}) = ((X_1, Y_1), \dots, (X_n, Y_n))$ tomada a partir de un vector aleatorio bidimensional (X, Y) (representa n mediciones tomadas al azar de la pareja de características representada por (X, Y)). (En el ejemplo 2.2, X representaría la temperatura en Huelva e Y , la temperatura en Tenerife y $((X_1, Y_1), \dots, (X_n, Y_n))$ denotaría la colección de $n = 36$ parejas de mediciones realizadas.) Denotemos por θ_X (respectivamente por θ_Y) a cierto parámetro (esperanza, mediana, etc.) asociado a la distribución de X (respectivamente, a la distribución de Y). Denotemos ahora por $\varphi : (\mathbb{R}^2)^n \rightarrow \{0, 1\}$ a una función de test para contrastar

$$H_0 : \theta_X = \theta_Y \quad \text{frente a} \quad H_1 : \theta_X \neq \theta_Y.$$

La función φ asocia, a cada posible realización muestral, $(\mathbf{x}, \mathbf{y}) = ((x_1, y_1), \dots, (x_n, y_n)) \in (\mathbb{R}^2)^n$, el valor 1

(rechazo) o 0 (no rechazo). Denotemos por R a la región crítica asociada a φ ,

$$R = \{(\mathbf{x}, \mathbf{y}) \in (\mathbb{R}^2)^n : \varphi(\mathbf{x}, \mathbf{y}) = 1\}.$$

Llamamos *tamaño* del test φ a la cantidad $\sup_{\theta \in \Theta_0} E_{\theta}(\varphi(\mathbf{X})) = \sup_{\theta \in \Theta_0} P_{\theta}(\text{Rechazar } H_0)$. Supongamos ahora que, para cada realización del experimento se obtienen una pareja de cotas de los valores de ambas variables. Consideremos, por tanto, una tupla de n vectores, $((L_1^X, U_1^X, L_1^Y, U_1^Y), \dots, (L_n^X, U_n^X, L_n^Y, U_n^Y))$ donde L_i^X (respectivamente L_i^Y) denota la cota inferior para el valor correspondiente a la i -ésima medición y U_i^X (respectivamente U_i^Y) denota la cota superior. Dicha tupla representa nuestra información (incompleta) acerca de la m.a.s. $((X_1, Y_1), \dots, (X_n, Y_n))$. De esta forma, para una realización muestral concreta, nuestra información vendrá dada por una colección ordenada de $4n$ valores de la forma $\gamma = ((l_1^X, u_1^X, l_1^Y, u_1^Y), \dots, (l_n^X, u_n^X, l_n^Y, u_n^Y))$. La salida del test φ será la siguiente:

$$\varphi(\gamma) = \{\varphi(x_1, \dots, x_n) : l_i \leq x_i \leq u_i, \forall i = 1, \dots, n\}.$$

Siempre que el parámetro θ sea un operador lineal, tal como ocurre con la esperanza, el contraste anterior se puede expresar alternativamente de la forma $H_0 : \theta_{X-Y} = 0$. Así, si denotamos por Z a la variable diferencia, $Z = X - Y$, el contraste anterior representa un caso particular de $H_0 : \theta_Z = \theta_0$ frente a $H_1 : \theta_Z \in \Theta_1$, donde $\Theta_1 \not\ni \theta_0$. A partir de ahora nos referiremos a este contraste más general y utilizaremos la notación θ en lugar de θ_Z , lo cual no dará lugar a confusión. Así, en lo que resta de trabajo, supondremos que nuestra entrada es un producto cartesiano $\gamma = [l_1, u_1] \times \dots \times [l_n, u_n]$, que representa nuestra información imprecisa acerca de una tupla de n valores, $\mathbf{z} = (z_1, \dots, z_n)$, tomados a partir de la variable $Z = X - Y$.

Dado un test, φ , para el contraste de la hipótesis $H_0 : \theta_Z = \theta_0$ frente a $H_1 : \theta_Z \in \Theta_1$, sus posibles salidas serán las siguientes:

$$\varphi(\gamma) = \begin{cases} \{1\} \text{ (rechazo)} & \text{si } \gamma \subseteq R \\ \{0\} \text{ (no rechazo)} & \text{si } \gamma \subseteq \mathbb{R}^n \setminus R \\ \{0, 1\} \text{ (indecisión)} & \text{en otro caso,} \end{cases}$$

donde R denota la región crítica asociada al test φ , $R = \{\mathbf{x} : \varphi(\mathbf{x}) = 1\}$.

Observemos que esta forma de proceder es conservativa, en el sentido de que si partimos de un test de tamaño α , la probabilidad de rechazo ($\varphi(\gamma) = \{1\}$) está acotada por dicho valor α , aun cuando los datos son imprecisos.

4. ELECCIÓN DE TESTS

Dado un test φ , la potencia asociada a un valor $\theta \in \Theta_1$, $\text{pot}_{\varphi}(\theta) = P_{\theta}(\varphi(\mathbf{X}) = 1)$, representa la probabilidad de rechazar la hipótesis nula, bajo la suposición de que el valor del parámetro es θ . Cuando las observaciones muestrales son totalmente precisas, un criterio extendido en la práctica consiste en elegir, entre todos los tests de un tamaño prefijado, α , aquel test φ^* (si es que tal existe) que sea uniformemente más potente, es decir, que satisfaga las desigualdades:

$$\text{pot}_{\varphi^*}(\theta) \geq \text{pot}_{\varphi}(\theta), \forall \theta \in \Theta_1, \forall \varphi.$$

Pero, cuando la información muestral es de tipo intervalar, el test uniformemente más potente no es, necesariamente, el que mejores propiedades tiene en la práctica. En una primera aproximación al problema de encontrar un criterio para comparar tests en este nuevo contexto, podemos centrar nuestra atención en las cantidades: $\underline{\text{pot}}(\theta) = P_{\theta}(\varphi(\Gamma) = \{1\})$ (probabilidad, bajo el supuesto de que el verdadero valor del parámetro es θ , de que el test indique “rechazar”) y $\overline{\text{pot}}(\theta) = P_{\theta}(\varphi(\Gamma) \ni 1)$ (probabilidad de rechazar o no concluir, bajo el mismo supuesto). Intuitivamente, cuanto más altas sean estas cantidades, más adecuado será el test φ . Pero surgen varias dificultades, que enumeramos a continuación:

1. $\underline{\text{pot}}(\theta)$ y $\overline{\text{pot}}(\theta)$ no son, realmente dos valores bien definidos en situaciones no triviales. En efecto, las probabilidades de las respectivas posibles salidas del test (rechazar, no rechazar, indecisión), dependen, en última instancia, de las distribuciones de probabilidad de lo que nosotros observamos (U =temperatura máxima diaria” y L =“temperatura mínima diaria”) y, por tanto, no quedan determinadas aunque conozcamos completamente la distribución de la variable Z (temperatura a las 16:00h). El problema anterior tiene una solución parcial. Aunque no disponemos de la información suficiente acerca de la relación entre nuestra información (observaciones de L y de U) y los valores reales de la variable X , como para determinar la distribución del vector (L, U) a partir de un determinado θ , sí tenemos, en ocasiones, cierta información (incompleta) acerca de esta relación. Así, en general, podemos calcular cotas inferiores y superiores no triviales (es decir, distintas de 1 y 0, respectivamente) para $\overline{\text{pot}}(\theta)$ y para $\underline{\text{pot}}(\theta)$.

A continuación mostramos un ejemplo sencillo para aclarar esta idea:

Ejemplo 4.1 Consideremos una v.a. Z con distribución $N(\mu, 1)^2$, donde μ es un valor desconocido. Supongamos que no podemos observar valores de Z de forma precisa, sino que únicamente podemos observar 25 valores tomados al azar a partir de una variable Y , de la cual sabemos que cumple la restricción $|Y - Z| \leq 0,1$. Queremos contrastar la hipótesis $H_0 : \mu = 0$ frente a $H_1 : \mu < 0$. Bajo la hipótesis nula, la variable aleatoria $\bar{Z}$ (media muestral) sigue distribución $N(0, 0.2)$. Por lo tanto, el test

$$\varphi(\mathbf{z}) = \begin{cases} 1 & (\text{rechazo}) & \text{si } \bar{z} < -0,33 \\ 0 & (\text{no rechazo}) & \text{si } \bar{z} \geq -0,33, \end{cases}$$

es un test de tamaño $\alpha = 0,05$ para este contraste. Para una realización muestral, sabemos que cada z_i ($i = 1, \dots, 25$) pertenece al intervalo $[y_i - 0,1, y_i + 0,1]$. Así que, para una secuencia de observaciones $\mathbf{y} = (y_1, \dots, y_{25})$ la salida del test será la siguiente:

$$\varphi(\gamma) = \begin{cases} \{1\} & (\text{rechazo}) & \text{si } \bar{y} < -0,43 \\ \{0\} & (\text{no rechazo}) & \text{si } \bar{y} \geq -0,23 \\ \{0, 1\} & (\text{indecisión}) & \text{en otro caso.} \end{cases}$$

Para el valor $\mu = -1$ la potencia de este test es:

$$\text{pot}(-1) = P_{\mu=-1}(\bar{Z} < -0,33) =$$

$$P_{\mu=-1}(5(\bar{Z} + 1) < 3,3) = \Phi_{N(0,1)}(3,35) = 0,9996.$$

Pero este valor nos informa de la probabilidad de rechazar H_0 (bajo el supuesto de que $Z \equiv N(-1, 1)$) a partir de una secuencia de 25 observaciones aleatorias totalmente precisas. No nos proporciona la probabilidad de rechazar a partir de una secuencia de 25 observaciones imprecisas.

Observemos que las probabilidades

$$P(\{\gamma : \varphi(\gamma) = \{1\}\}) = P(\bar{Y} < -0,43) \text{ y}$$

$$P(\{\gamma : \varphi(\gamma) = \{1\} \text{ ó } \varphi(\gamma) = \{0, 1\}\}) =$$

$$P(\bar{Y} \leq -0,23)$$

²En este ejemplo no pretendemos ilustrar una situación real. No son habituales los problemas en los que se desconoce la media de la población, pero se conoce la varianza. Por otra parte, tampoco mostramos las dificultades que presenta el hecho de determinar si los datos muestrales provienen o no de una población normal, cuando son observados de forma imprecisa.

no quedan completamente especificadas a partir del supuesto de que $Z \equiv N(-1, 1)$. Pero teniendo en cuenta las desigualdades:

$$\bar{Z} - 0,1 \leq \bar{Y} \leq \bar{Z} + 0,1,$$

podemos afirmar que:

$$P(\bar{Z} + 0,1 < y) \leq P(\bar{Y} < y) < P(\bar{Z} - 0,1 < y), \quad \forall y.$$

Así, el valor: $P_{\mu=-1}(\bar{Z} + 0,1 < -0,43) = \Phi_{N(0,1)}(2,35) = 0,9906$ es una cota inferior para $P(\{\gamma : \varphi(\gamma) = \{1\}\})$, mientras que el valor $P_{\mu=-1}(\bar{Z} + 0,1 < -0,23) = \Phi_{N(0,1)}(4,35) \approx 1$ es una cota superior para $P(\{\gamma : \varphi(\gamma) = \{1\} \text{ ó } \varphi(\gamma) = \{0, 1\}\})$. Por otra parte, la cantidad $\text{pot}(-1) = 0,9995$ es, simultáneamente, una cota superior para la primera y una cota inferior para la segunda.

2. Sin embargo, aunque esas cotas estén bien definidas, su cálculo es un problema de difícil resolución, tal como mostramos en el siguiente ejemplo.

Ejemplo 4.2 Consideremos ahora una v.a. Z con distribución $N(\mu, \sigma)$, donde μ y σ son valores desconocidos. Supongamos que no podemos observar valores de Z de forma precisa, sino que únicamente podemos observar 25 valores tomados al azar a partir de una variable Y , de la cual sabemos que cumple la restricción $|Y - Z| \leq 0,1$. Queremos contrastar la hipótesis $H_0 : \mu = 0$ frente a $H_1 : \mu < 0$. Bajo la hipótesis nula, la variable aleatoria $5\bar{Z}/S_Z$, donde S_Z denota la cuasi-varianza muestral, sigue distribución t_{24} . Por lo tanto, el test

$$\varphi(\mathbf{z}) = \begin{cases} 1 & (\text{rechazo}) & \text{si } \frac{5\bar{z}}{s_z} < -1,71 \\ 0 & (\text{no rechazo}) & \text{si } \frac{5\bar{z}}{s_z} \geq -1,71, \end{cases}$$

es un test de tamaño $\alpha = 0,05$ para este contraste. Para una realización muestral, sabemos que cada z_i ($i = 1, \dots, 25$) pertenece al intervalo $[y_i - 0,1, y_i + 0,1]$. Así que, para una secuencia de observaciones $\mathbf{y} = (y_1, \dots, y_{25})$ la salida del test será la siguiente:

$$\varphi(\gamma) = \begin{cases} \{1\} & \text{si } \sup_{\mathbf{x}: x_i \in [y_i - 0,1, y_i + 0,1] \forall i} \frac{5\bar{x}}{s_x} < -1,71 \\ \{0\} & \text{si } \inf_{\mathbf{x}: x_i \in [y_i - 0,1, y_i + 0,1] \forall i} \frac{5\bar{x}}{s_x} > -1,71 \\ \{0, 1\} & \text{en otro caso.} \end{cases}$$

Dado un vector $\mathbf{y} = (y_1, \dots, y_n)$ la determinación de las cantidades $\sup_{\mathbf{x}: x_i \in [y_i - 0,1, y_i + 0,1] \forall i} s_x$ e

$\inf_{\mathbf{x}: x_i \in [y_i - 0, 1, y_i + 0, 1] \forall i} s_x$ no es un problema trivial y ha sido objeto de diversos estudios recientes (ver, por ejemplo, [6]). Por tanto, en este ejemplo, la determinación de la salida del test para cada posible muestra imprecisa no es inmediata. Así, aunque, en principio, existen cotas no triviales para $\underline{\text{pot}}(\theta)$ y $\overline{\text{pot}}(\theta)$ (Denotemos por $\underline{\text{pot}}_{\varphi}(\theta)$ a una cota inferior no trivial de $\underline{\text{pot}}_{\varphi}(\theta)$ y por $\overline{\text{pot}}_{\varphi}(\theta)$ a una cota superior de $\overline{\text{pot}}_{\varphi}(\theta)$, respectivamente), su determinación requiere la utilización de ciertos algoritmos de optimización.

3. Para comparar dos tests diferentes, φ_1 y φ_2 , es necesario comparar los intervalos de cotas asociados, $[\underline{\text{pot}}_{\varphi_1}(\theta), \overline{\text{pot}}_{\varphi_1}(\theta)]$ y $[\underline{\text{pot}}_{\varphi_2}(\theta), \overline{\text{pot}}_{\varphi_2}(\theta)]$. Si $\underline{\text{pot}}_{\varphi_1}(\theta) \geq \overline{\text{pot}}_{\varphi_2}(\theta)$, preferiremos el test φ_1 al test φ_2 . Análogamente, $\underline{\text{pot}}_{\varphi_2}(\theta) \geq \overline{\text{pot}}_{\varphi_1}(\theta)$, preferiremos el segundo test. Pero en general, los intervalos $(\underline{\text{pot}}_{\varphi_1}(\theta), \overline{\text{pot}}_{\varphi_1}(\theta))$ y $(\underline{\text{pot}}_{\varphi_2}(\theta), \overline{\text{pot}}_{\varphi_2}(\theta))$ tendrán intersección no vacía, de forma que habrá que elegir un criterio adecuado para compararlos.

Las dificultades señaladas en los tres puntos anteriores serán objeto de nuestro estudio en un futuro próximo.

5. CONCLUSIONES

En este trabajo nos hemos ceñido al caso en que podemos modelar nuestra información por medio de intervalos. Este tipo de información puede encuadrarse en el marco más general de la teoría de la posibilidad. En efecto, la distribución de posibilidad 0-1-valuada:

$$\pi(c) = \begin{cases} 1 & \text{si } c \in [l, u] \\ 0 & \text{en otro caso,} \end{cases}$$

indica que el grado de seguridad (necesidad) de que el valor desconocido z pertenece al valor $[l, u]$ es 1, ya que la medida de necesidad asociada a ella satisface la igualdad: $N([l, u]) = 1 - \Pi([l, u]^c) = 1 - \sup_{c \in [l, u]^c} \pi(c) = 1$. Así, las nociones de test, p-valor e intervalo de confianza han sido extendidas recientemente al caso en que los datos muestrales (entradas) son subconjuntos difusos de $\mathbb{R}^n$ que indican distribuciones de posibilidad más generales. Tal como se demuestra en [2], dichos grados de posibilidad (o valores de la función de pertenencia)

quedan completamente especificados a partir de una familia anidada de intervalos de confianza que describe la información disponible acerca del valor z . En recientes trabajos, se han estudiado las extensiones de los conceptos de test, de p-valor y región de confianza a este contexto más general ([3, 4, 5]). En concreto, el p-valor extendido se define como un conjunto difuso, que se interpreta como una medida de posibilidad. En [3] hemos reducido ese conjunto difuso a un intervalo, dotándole de una interpretación coherente dentro de la teoría de las probabilidades imprecisas [7]. En un futuro próximo nos proponemos extender el concepto de potencia a este contexto más general. Presumiblemente, dicha extensión dará lugar a un número difuso, que podrá interpretarse como una medida de posibilidad. Al igual que en el caso del p-valor, intentaremos reducir este difuso a un intervalo, de forma que el problema de encontrar un test adecuado no se complique aun más para ese caso.

Agradecimientos

Este trabajo está financiado por el Ministerio de Educación y Ciencia, proyectos TIN2007-67418-C03-03 y MTM2007-61193, y por fondos FEDER FEDER-MEC.

Referencias

- [1] C. Bertoluzza, N. Corral, A. Salas, On a new class of distances between fuzzy numbers, *Mathware Soft Comput.* 2 (1995) 71- 84.
- [2] I. Couso, S. Montes y P. Gil, The necessity of the strong α -cuts of a fuzzy set, *IJUFKBS* 9 (2001) 249-262.
- [3] I. Couso, L. Sánchez, Defuzzification of fuzzy p-values En: *Soft Methods for Handling Variability and Imprecision*, D. Dubois et al. (Eds.) Springer, 2008.
- [4] I. Couso, L. Sánchez, Inner and outer regions induced by low quality data, *Eurofuse Workshop'09 -Preference modelling and decision analysis*, Navarra, 2009.
- [5] T. Denoeux, M.H. Masson, P.A. Hébert, Nonparametric rank-based statistics and significance tests for fuzzy data, *FSS* 153 (2005) 1-28.
- [6] S. Ferson, V. Kreinovich, J. Hajagos, W. Oberkampff and L. Ginzburg, *Experimental Uncertainty Estimation and Statistics for Data Having Interval Uncertainty*, T.R. SAND2007-0939, 2007.
- [7] P. Walley, *Statistical Reasoning with Imprecise Probabilities*. 1991. Chapman and Hall, London.

Uso de Marcadores Difusos para Solucionar el Problema de la Coplanaridad en la Calibración de la Cámara en 3D. Aplicación en Identificación Forense por Superposición Craneofacial

Óscar Ibáñez¹ Óscar Cordon¹ Sergio Damas¹ José Santamaría²

¹ European Centre for Soft Computing, {oscar.ibanez,oscar.cordon,sergio.damas}@softcomputing.es

² Departamento de Ciencias de la Computación, Universidad de Jaén, jslopez@ujaen.es

Resumen

La Superposición Craneofacial es un proceso forense cuyo objetivo es identificar a personas desaparecidas a partir de fotografías y de un cráneo encontrado. Las técnicas más avanzadas usan un modelo 3D del cráneo.

La segunda etapa de esta compleja técnica forense, el emparejamiento cráneo-cara, trata de encontrar la pose más adecuada del modelo 3D del cráneo que será proyectado sobre la fotografía. Esta segunda etapa se puede modelar como un problema de calibración de la cámara, problema ampliamente referenciado en la bibliografía de visión por ordenador.

Esta contribución se centra en el estudio de la coplanaridad del conjunto de marcadores usados para guiar el emparejamiento cráneo-cara. Además, se propone el uso de marcadores difusos para evitar situaciones de coplanaridad

Palabras Clave: Superposición craneofacial, marcadores difusos, calibración de la cámara.

1. INTRODUCCIÓN

En antropología forense, la superposición craneofacial (SC) es un proceso en el que fotografías de una persona desaparecida son comparadas con un cráneo encontrado. Proyectando este sobre una fotografía (o, incluso mejor, haciendo corresponder un modelo 3D del cráneo con la fotografía de la cara), el forense trata de determinar si se trata de la misma persona.

En [4] los autores presentan un enfoque metodológico de este proceso, identificando las siguientes tres etapas:

La primera etapa conlleva la construcción de un modelo digital del cráneo de manera precisa. Este problema ya fue abordado con éxito en nuestros trabajos mediante el uso de metaheurísticas para el registrado de las diferentes vistas del cráneo [12].

En la siguiente etapa de la tarea de identificación, conocida como *emparejamiento cráneo-cara*, se trata de superponer el modelo 3D del cráneo sobre la fotografía de la cara mediante el emparejamiento de dos conjuntos de puntos radiométricos (puntos antropométricos faciales en la cara del sujeto y puntos antropométricos craneales en el modelo 3D del cráneo obtenido) [10].

A continuación, comienza la tercera etapa, ayuda a la toma de decisiones, en donde se analizan los diferentes tipos de emparejamientos entre marcadores logrados en la etapa anterior.

La SC es un proceso que implica mucho tiempo de trabajo y para el que no hay una metodología sistemática ya que cada experto suele aplicar su propio proceso. Por lo tanto, hay un fuerte interés en el diseño de métodos automáticos que den soporte al antropólogo forense en la realización de esta tarea.

La propuesta aquí presentada se centra en la segunda etapa del proceso de superposición craneofacial, formulada por Ibáñez y colaboradores [9] como un problema de registrado de imágenes (RI) [14] y abordado por medio de algoritmos evolutivos (AEs) [2].

En todos esos trabajos previos los autores no llegaron a estudiar la influencia que tiene el conjunto de marcadores proporcionados por el experto forense para guiar la superposición, sobre el emparejamiento cráneo-cara resultante, motivo de este trabajo.

Además, proponemos el uso de conjuntos difusos para modelar un nuevo tipo de marcador, marcador difuso o incierto, que permitan a los expertos forenses localizar un mayor número de puntos cefalométricos, y así evitar la coplanaridad de éstos.

Este artículo se ha estructurado de la siguiente forma. La Sección 2 resume la modelización utilizada en la etapa de emparejamiento cráneo-cara, así como los aspectos más importantes del problema de coplanaridad en visión por ordenador. A continuación, en la Sección 3 se presenta un estudio experimental sobre el comportamiento de diferentes AEs cuando son guiados por un conjunto de marcadores coplanares. En la Sección 4 se presenta nuestra propuesta basada en el uso de marcadores difusos. La experimentación sobre casos de estudio reales se presenta en la Sección 5 para posteriormente, en la Sección 6, analizar los emparejamientos resultantes. Finalmente, en la Sección 7 se agrupan una serie de conclusiones y trabajos futuros.

2. PRELIMINARES

2.1. EMPAREJAMIENTO CRÁNEO-CARA

Como ya se ha mencionado, el objetivo de esta etapa es lograr una adecuada superposición del modelo 3D del cráneo sobre la fotografía de la cara de la persona desaparecida mediante el emparejamiento de pares de marcadores. En [9], los autores formularon esta tarea como un problema de optimización numérica relacionado con el registrado de imágenes (RI). De este modo, dos conjuntos de marcadores guían la búsqueda automática de los parámetros de la transformación que logre el preciso emparejamiento de dichos puntos. Esta transformación se corresponde con una proyección 3D/2D, modelada en [9] como un conjunto de operaciones geométricas, definidas por doce parámetros/incógnitas.

Formalmente, la formulación del emparejamiento cráneo-cara es la siguiente: dados dos conjuntos de marcadores faciales 2D y craneales 3D, F y C , respectivamente, ambos compuestos de N puntos, la tarea de emparejamiento trata de resolver el sistema de ecuaciones con las siguientes doce incógnitas: una rotación representada por un eje (d_x, d_y, d_z) y un ángulo (θ) , un centro de masas (r_x, r_y, r_z) , un vector de traslación (t_x, t_y, t_z) , un escalado uniforme (s) , y una función de proyección 3D/2D dada por un campo de visión (ϕ) . Estos doce parámetros determinan la transformación geométrica f que proyecta cada punto craneométrico C_i del modelo 3D sobre su correspondiente punto cefalométrico F_i en la fotografía:

$$F = f(C) = C \cdot (A \cdot D_1 \cdot D_2 \cdot R_\theta \cdot D_2^{-1} \cdot D_1^{-1} \cdot A^{-1}) \cdot S \cdot T \cdot P \quad (1)$$

donde $R = (A \cdot D_1 \cdot D_2 \cdot R_\theta \cdot D_2^{-1} \cdot D_1^{-1} \cdot A^{-1})$ representa una matriz de rotación para orientar el cráneo a la misma pose que tiene la cara en la fotografía. S , T , y P son las matrices de escalado uniforme, traslación, proyección perspectiva, respectivamente¹.

¹Se recomienda [7], para aquellos lectores interesados en

Por otro lado, en [9], los autores adaptaron tres AEs para resolver el problema de emparejamiento: dos variantes de un algoritmo genético (AG) de codificación real y la estrategia evolutiva CMA-ES.

La función objetivo a minimizar por estos algoritmos de búsqueda/optimización es el error medio²:

$$EM = \frac{\sum_{i=1}^N \|f(cl^i) - fl^i\|}{N} \quad (2)$$

donde $\|\cdot\|$ es la distancia Euclídea 2D, N es el número de marcadores considerados (proporcionados por los expertos forenses), cl^i se corresponde con cada punto craneométrico 3D, mientras que fl^i hace referencia a los puntos cefalométricos 2D, f es la función que define la transformación proyectiva 3D-2D, y $f(cl^i)$ representa cada punto craneométrico cl^i una vez proyectado sobre el plano imagen/foto.

2.2. COPLANARIDAD EN VISIÓN POR ORDENADOR

El problema de emparejamiento cráneo-cara puede ser claramente relacionado con otro problema ampliamente estudiado en visión por ordenador (VO), el problema de la calibración de la cámara (CC) [6]. El propósito de la CC es estudiar la relación entre coordenadas 3D del mundo real y sus correspondientes coordenadas 2D de la imagen adquirida mediante una cámara. La mayoría de las aproximaciones de CC distinguen dos etapas:

- Determinar los *parámetros intrínsecos*: centro de proyección y distancia focal de la cámara.
- Determinar los *parámetros extrínsecos*: transformación rígida (rotación y traslación) que relaciona el sistema de coordenadas de la cámara al sistema de coordenadas del mundo real.

Dentro de los métodos más usados para determinar los conjuntos de parámetros intrínsecos y extrínsecos está la transformación lineal directa (DLT, del inglés *direct linear transformation*), propuesto originalmente por Abdel-Aziz y Karara [1]. En este método, los parámetros de la cámara se obtienen a partir de correspondencias entre pares de puntos de referencia 3D/2D.

En el problema de emparejamiento cráneo-cara no se suelen conocer los parámetros internos de la cámara con la que se tomó la fotografía de la persona desaparecida, por lo que deben ser también estimados durante

una descripción más detallada de las matrices de la ecuación 1 y de su relación con las doce incógnitas del problema.

²Nótese que el error cuadrático medio se descartó debido al efecto negativo que éste tiene cuando los rangos de la imagen se normalizan entre $[0,1]$, como es nuestro caso.

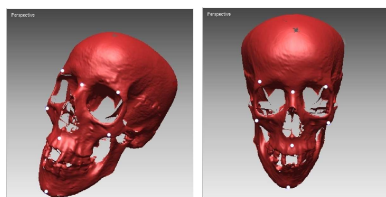


Figura 1: Poses frontal y lateral del caso sintético de cráneo humano. Los puntos craneométricos y cefalométricos se han resaltado mediante círculos blancos.

el proceso de optimización. Si bien la mayoría de los trabajos en VO que abordan el problema de CC usan puntos de referencia no coplanares, el escenario es mucho más complejo con la coplanaridad de éstos [11], situación en la que estamos más interesados. Además, no es un problema que se haya estudiado profundamente en la comunidad científica, siendo el trabajo de Tsai [13] uno de los primeros que estudió el problema en profundidad, así como uno de los más citados. En un primer paso, el método lleva a cabo una estimación lineal de los parámetros de CC. Dicha estimación se utilizará como la inicial para una segunda etapa en donde se usa una estimación no-lineal basada en el método de Levenberg-Marquardt, que hará un refinado de los parámetros³. Por contra, es bien sabido que este tipo de métodos de optimización no-lineales son muy dependientes de la inicialización siendo muy frecuente su convergencia a óptimos locales. Esta situación nos llevó a pensar nuevamente en el uso de AEs para abordar el problema de manera más robusta.

3. COPLANARIDAD EN EL EMPAREJAMIENTO

En esta sección se presenta un estudio experimental del comportamiento de un AE abordando el problema de emparejamiento cráneo-cara cuando el conjunto de marcadores proporcionados es coplanar. En particular, de entre los tres diseños evolutivos diferentes propuestos en [9] para resolver el problema, hemos optado por el AG generacional de codificación real (RCGA) con los siguientes operadores: selección por torneo, cruce binario simulado (SBX, del inglés *simulated binary crossover*) y mutación aleatoria. La función objetivo a minimizar viene dada por la Ecuación 2.

Este estudio preliminar se centra en un caso sintético. Como primera aproximación al problema de la coplanaridad, hemos preferido simplificarlo considerando el mismo objeto, es decir, el modelo 3D de cráneo y una

³Aquellos lectores interesados en una descripción más detallada del método de Tsai pueden dirigirse a

http://people.csail.mit.edu/people/bkph/articles/Tsai_Revisited.pdf

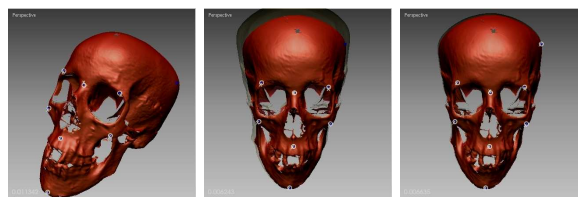


Figura 2: Mejor superposición resultante en la pose lateral (izquierda), y para la pose frontal usando siete (centro) y ocho (derecha) marcadores. Cruces y círculos indican la localización de los puntos craneométricos y cefalométricos, respectivamente.

imagen 2D del mismo. Se ha obtenido un modelo 3D, y se han considerado dos proyecciones del mismo, una rotada (lateral) y otra frontal, que se utilizarán como la imagen 2D sobre la que superponer el modelo 3D. Con respecto al conjunto de marcadores, se localizaron siete puntos craneométricos simulando el proceso seguido por los expertos forenses. Aunque dadas las características de un cráneo no es común tener un conjunto de puntos completamente coplanares, podemos tratarlo como tal por tener todos sus puntos un valor de profundidad muy similar. Los puntos cefalométricos se corresponden directamente con los puntos craneométricos proyectados sobre el plano imagen. La Figura 1 muestra las dos imágenes consideradas y el conjunto de puntos de referencia usados.

En cuanto a la pose lateral, la imagen más a la izquierda de la Figura 2 muestra el mejor resultado del RCGA con siete marcadores. Nótese que, no hay diferencias apreciables entre el modelo 3D proyectado y la imagen 2D.

El comportamiento del AE para el caso de la pose frontal y los mismos siete marcadores es bastante diferente (Figura 2, centro). La distribución casi coplanar de estos marcadores hace que el sistema de ecuaciones que rige la transformación 3D-2D sea indeterminado, y por lo tanto, no es posible encontrar una transformación que empareje adecuadamente cráneo y cara. Sin embargo, añadiendo un nuevo marcador en un plano diferente al del resto, el sistema de ecuaciones pasa a ser determinado y el AE es capaz de encontrar un mínimo global que se corresponde con un adecuado emparejamiento (Figura 2, derecha).

Teniendo en cuenta que tanto en el caso de la pose lateral como frontal ambos comparten el mismo conjunto de puntos craneométricos casi coplanares, este simple caso de estudio nos lleva a las siguientes conclusiones. El inadecuado comportamiento del procedimiento de optimización del emparejamiento cráneo-cara se debe principalmente a: i) el uso de un conjunto de puntos craneométricos coplanar o casi coplanar; y ii) el pa-

ralelismo entre el plano que contiene los puntos craneométricos y el plano imagen de la cámara.

Por tanto, para resolver estas es necesario el estudio de alternativas más robustas.

4. MARCADORES DIFUSOS PARA LA COPLANARIDAD

En el problema de SC, los expertos forenses han usado durante siglos marcadores precisos (un punto) para identificar los diferentes puntos antropométricos (craneales y faciales). Sin embargo, por culpa de oclusiones de zonas de la cara debido a la perspectiva o a la presencia de otros objetos, y también por la baja calidad/resolución/estado conservación de las fotografías, muchos de los marcadores faciales no pueden ser localizados en la fotografía. En el modelo 3D del cráneo se pueden localizar de manera precisa todos los marcadores dada la resolución de los escáneres usados.

En [8] se propone el uso de marcadores difusos para abordar el problema de la incertidumbre en la localización de marcadores faciales. Esta aproximación permite a los expertos forenses usar una figura geométrica (una elipse en el artículo) para delimitar una pequeña zona de la imagen en donde con toda seguridad se encontrará el punto fisonómico que se corresponde con el marcador. Por lo tanto, usando marcadores difusos en lugar de, o además de, marcadores precisos, los expertos pueden localizar un mayor número de marcadores con el adecuado nivel de confianza (usando marcadores difusos de diferente tamaño). Dado que los forenses son incapaces de situar un número suficiente de marcadores precisos solo en la fotografía, el uso de más marcadores faciales (difusos) eliminará la coplanaridad en el nuevo conjunto de puntos craneométricos correspondiente.

Siguiendo la idea de métricas espaciales descrita en [5], hemos usado el marco basado en conjuntos difusos propuesto en [8] para modelar los marcadores difusos y la distancia de estos a marcadores precisos.

Si bien la distancia entre un par de marcadores precisos se calcula mediante la distancia Euclídea, en el caso en el que uno de los marcadores sea difuso (conjunto de puntos), se trata de calcular la distancia de un punto x a un conjunto difuso $\tilde{F}$ de la siguiente manera,

$$d^*(x, \tilde{F}) = \sum_{i=1}^m d_i \cdot \alpha_i / \sum_{i=1}^m \alpha_i$$

donde, $d_i = d(x, \tilde{F}_{\alpha_i})$ es la distancia del punto x al conjunto de puntos del α -corte $\tilde{F}_{\alpha_i}$.

Por lo tanto, la función de objetivo a minimizar es la que se muestra en la Ecuación 2):

$$fuzzy ME = \frac{\sum_{i=1}^N d^*(f(cl^i), \tilde{F}^i)}{N} \quad (3)$$

donde N es el número de marcadores considerados, cl^i se corresponde con cada punto 3D craneométrico, f es la función de transformación 3D-2D, $f(cl^i)$ representa la posición de los puntos craneométricos cl^i una vez han sido proyectados al plano imagen, es decir, un único punto. $\tilde{F}^i$ representa el conjunto difuso correspondiente a cada marcador difuso.

5. EXPERIMENTACIÓN

En la Sección 5.1 se presenta el diseño experimental seguido. En las Secciones 5.2 y 5.3 se analizan los resultados en dos casos de estudio reales, donde la pose que tiene la persona desaparecida en la fotografía es frontal.

5.1. DISEÑO EXPERIMENTAL

El primer caso de estudio real tuvo lugar en Cádiz. La familia de la desaparecida proporcionó una fotografía⁴. La pose de la cara es casi frontal y el correspondiente conjunto de marcadores es casi coplanar. El segundo caso de estudio real tuvo lugar en Granada. En este caso sólo se dispone de una imagen de la cara de la desaparecida, también en una pose casi frontal. Para la adquisición de los modelos 3D del cráneo se utilizó el laser-escáner VI-910 de Konica-Minolta.

Los expertos forenses nos proporcionaron dos conjuntos de marcadores distintos por cada caso de estudio. El primero compuesto por puntos precisos (sólo aquellos marcadores cuya localización es precisa) y el segundo compuesto por puntos difusos (elipses de tamaño y orientación variable). Es importante destacar que el uso de marcadores difusos posibilitó que los expertos fueran capaces de localizar más marcadores.

En todos los experimentos se utilizó la mejor configuración de parámetros del RCGA, según [9]: número de generaciones = 600, tamaño de población = 1,000, parámetros η del SBX = 1, probabilidad de mutación = 0.2, probabilidad de cruce = 0.9, y tamaño de torneo = 2. Para todas las pruebas se consideraron 30 ejecuciones del algoritmo. Para conjuntos de marcadores precisos, la función de evaluación viene dada por la Ecuación 2. En el caso de conjuntos de marcadores difusos, la función de evaluación es la definida en la Ecuación 3.

⁴Por motivos legales ésta ha sido procesada.



Figura 3: Caso de estudio 1. modelo 3D del cráneo (izquierda) y fotografía de la persona desaparecida con marcadores precisos (izquierda) y difusos (derecha).



Figura 4: Caso de estudio 1. Mejores resultados de emparejamiento cráneo-cara usando nueve marcadores precisos (izquierda) y catorce difusos (derecha).

5.2. CASO DE ESTUDIO 1

En este primer caso (Figura 3), los antropólogos identificaron nueve y catorce marcadores siguiendo un enfoque preciso y difuso, respectivamente. Estos marcadores adicionales juegan un rol esencial a la hora de resolver el problema de la coplanaridad ya que, sus correspondientes puntos craneométricos (como por ejemplo el marcador más elevado), se sitúan en un plano que no es paralelo al plano imagen de la cámara.

Para poder realizar una comparativa visual, en la Figura 4 se muestra el mejor resultado usando marcadores precisos y difusos.

5.3. CASO DE ESTUDIO 2

En este segundo caso de estudio real, la única fotografía disponible es la que aparece en el pasaporte (Figura 5). Como es normal en los pasaportes, incluye un sello de agua que hace más difícil aún una localización precisa de puntos cefalométricos. Por lo tanto, el uso de marcadores difusos puede ayudar a localizar un mayor número de marcadores faciales. De hecho, si bien los expertos forenses sólo localizaron seis marcadores de manera precisa, fueron capaces de identificar dieciséis marcadores difusos (Figura 5).

La Figura 6 muestra la mejor solución encontrada para ambas aproximaciones.

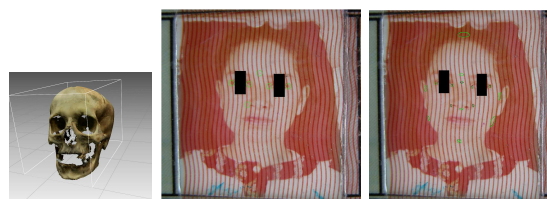


Figura 5: Caso de estudio 2. Modelo 3D del cráneo (izquierda) y fotografía de la desaparecida con puntos cefalométricos precisos (centro) y difusos (derecha).



Figura 6: Caso de estudio 2. Fila superior. Mejor emparejamiento con marcadores precisos (izquierda) y difusos (derecha). Fila inferior. Imagen binaria resultante de aplicar el operador XOR sobre las imágenes del mejor emparejamiento, usando marcadores precisos (izquierda) y difusos (derecha).

6. DISCUSIÓN

Teniendo en cuenta resultados cualitativos, es decir, analizando visualmente los emparejamientos cráneo-cara obtenidos, se identifica claramente la mejora en el proceso de emparejamiento cuando se incluye el uso de marcadores difusos (Figuras 4 y 6). El equipo de antropólogos forenses confirmaron que los resultados obtenidos eran suficientemente buenos para poder usarlos en la última etapa del proceso de SC.

Dado que no es posible una comparación cuantitativa de la calidad de los emparejamientos cráneo-cara en base a los valores de la función objetivo (diferente en cada caso), se diseñó un nuevo método que posibilita un análisis cuantitativo más en correspondencia con lo que los expertos forenses consideran un buen emparejamiento. Para ello, los antropólogos forenses segmentaron manualmente el contorno de la cara en la fotografía. A continuación, se obtiene una imagen binaria tanto de los bordes de la cara como del cráneo, una vez proyectado. Después, se aplica el operador lógico XOR considerando las dos imágenes binarias resultantes en el paso anterior. Finalmente, el error asociado a un emparejamiento cráneo-cara se calcula como el

porcentaje del contorno de la cara que no es cubierto por el área que ocupa el cráneo proyectado. Por ejemplo, la Figura 6 muestra el resultado de aplicar el operador XOR sobre la mejor superposición del segundo caso de estudio. El error obtenido para los dos casos de estudio puede verse en la Tabla 1. Hay que tener en cuenta que, si bien este método posibilita una comparativa cuantitativa, esta está ligada al proceso de valoración subjetiva del forense.

Tabla 1: Error de desviación del área de los mejores emparejamientos cráneo-cara mediante las dos aproximaciones estudiadas.

Tipo de Marcadores	Caso 1	Caso 2
Precisos	49,18 %	48,03 %
Difusos	24,17 %	41,35 %

7. CONCLUSIONES

Hemos propuesto el uso de marcadores difusos para abordar el problema de la coplanaridad en el emparejamiento cráneo-cara. Se han probado los beneficios del uso de este tipo de marcadores en comparación con los tradicionales precisos. Los resultados muestran como mediante el uso de marcadores difusos se elimina la coplanaridad del conjunto de marcadores, posibilitando la obtención de emparejamientos adecuados.

Como única desventaja, el uso de marcadores difusos, y en concreto el cálculo de distancias entre puntos difusos y precisos, conlleva un mayor tiempo de cómputo del AG: de los 20-40" por ejecución en el caso de usar marcadores precisos a los 2-4' para el caso difuso. Sin embargo, este tiempo es todavía significativamente mucho menor que el empleado por los expertos forenses a la hora de realizar la misma tarea de manera manual, en la cual pueden llegar a invertir 24 horas por cada caso de estudio.

Por otro lado, estos resultados tienen que ser confirmados en un estudio más extensivo, con un mayor número de casos y usando otras definiciones para distancias entre un punto y un conjunto difuso [3].

Finalmente, planeamos abordar la última etapa del proceso de superposición, es decir la etapa de ayuda a la toma de decisiones, mediante el uso de lógica difusa.

Agradecimientos

Este trabajo está parcialmente soportado por el Ministerio de Ciencia y Tecnología (ref. TIN2009-07727) y la Consejería de Innovación, Ciencia y Empresa de la Junta de Andalucía (ref. TIC1619), incluyendo fondos FEDER.

Referencias

- [1] Y.I. Abdel-Aziz and H.M. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Symposium on Close-Range Photogrammetry*, pages 1–18, 1971.
- [2] T. Bäck, D. B. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation*. IOP Publishing Ltd and Oxford University Press, 1997.
- [3] I. Bloch. On fuzzy distances and their use in image processing under imprecision. *Pattern Recognition*, 32:1873–1895, 1999.
- [4] S. Damas, O. Cordon, O. Ibáñez, J. Santamaría, I. Alemán, F. Navarro, and M. Botella. Forensic identification by computer-aided craniofacial superimposition: a survey. Technical report, European Centre for Soft Computing, Mieres, Spain.
- [5] P. Diamond and P. Kloeden. Metric topology of fuzzy numbers and fuzzy analysis. In D. Dubois and H. Prade, editors, *Fundamentals of Fuzzy Sets*, The Handbooks of Fuzzy Sets, chapter 11, pages 583–637. Kluwer Academic, 2000.
- [6] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [7] D. Hearn and M. Pauline Baker. *Computer graphics (2nd ed.): C version*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1997.
- [8] L. Ibáñez, O. Cordon, S. Damas, S. Guadarrama, and J. Santamaría. A new approach to fuzzy location of cephalometric landmarks in craniofacial superimposition. In *Proceeding of the IFSA/EUFLAT Conference*, pages 195–201, 2009.
- [9] O. Ibáñez, L. Ballerini, O. Cordon, S. Damas, and J. Santamaría. An experimental study on the applicability of evolutionary algorithms to craniofacial superimposition in forensic identification. *Information Sciences*, 179:3998–4028, 2009.
- [10] M. Y. Iscan. Introduction to techniques for photographic comparison. In M. Y. Iscan and R. Helmer, editors, *Forensic Analysis of the Skull*, pages 57–90. Wiley, 1993.
- [11] J. Salvi, X. Armangué, and J. Batlle. Algorithms for coplanar camera calibration. *Machine Vision and Applications*, 12:84–97, 2000.
- [12] J. Santamaría, O. Cordon, S. Damas, J. M. García-Torres, and A. Quirin. Performance evaluation of memetic approaches in 3D reconstruction of forensic objects. *Soft Computing*, in press (2008), 13(8-9):883–904, 2009.
- [13] R.Y. Tsai. An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 364–374, 1986.
- [14] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.

NUEVOS DISEÑOS EXPERIMENTALES PARA ALGORITMOS DE EXTRACCIÓN DE CONOCIMIENTO CON DATOS DE BAJA CALIDAD

José Otero¹

¹ Edificio Departamental 1, 33204 (Gijón) Asturias, jotero@uniovi.es

Resumen

El estudio de los diseños experimentales para algoritmos que usan datos de baja calidad y particularmente, datos intervalares ha recibido la atención por parte de la comunidad de Machine Learning en los últimos años. Hay consenso en que no se pueden extraer conclusiones de las técnicas habituales como las basadas en validación cruzada cuando la varianza de los puntos centrales de los intervalos es del mismo orden que la imprecisión de los datos. En este trabajo, se muestra un estudio empírico preliminar en donde se usa la extensión intervalar habitual de un test de hipótesis no paramétrico para incorporarlo a un diseño experimental basado en validación cruzada con el fin de solventar este problema. Usando esta configuración se analizan los resultados de una experimentación con datos intervalares, realizando una estimación del error de tipo I y de la potencia de los tests utilizados.

Palabras Clave: Datos de baja calidad, tests estadísticos, diseño experimental.

1 INTRODUCCIÓN

Existe una gran cantidad de material bibliográfico en donde se recopilan o analizan la mayor parte de los diseños experimentales y tests estadísticos de aplicación en experimentaciones de Machine Learning [10], ya que la inclusión de estos en el análisis de los resultados se ha hecho obligatoria [3]. En la casi totalidad de los casos, estos estudios se ciñen a las situaciones en las que los datos son nítidos, sin incertidumbre. Sin embargo es habitual en aplicaciones reales encontrarse

con esta en los datos por distintos motivos, tolerancias en las medidas, datos perdidos o censurados, etc. La temática del análisis estadístico de datos con imprecisión está también muy desarrollada [1, 8], pero no así su aplicación a experimentaciones de Machine Learning. En este trabajo pretendemos mostrar algunos resultados preliminares de la aplicación directa de las técnicas existentes en Estadística a las experimentaciones de Machine Learning con datos de baja calidad. Los resultados provienen de una experimentación con datos sintéticos que nos permite controlar la tolerancia de los datos y la dificultad del problema (para el test estadístico). El objetivo es observar como se comportan las extensiones habituales de los tests estadísticos cuando se aplican a experimentaciones de Machine Learning en las que los resultados poseen imprecisión. En la siguiente sección se describe el problema de las fuentes de imprecisión en los algoritmos y se citan las contribuciones más destacadas en el campo del análisis estadístico de datos imprecisos. En la sección 3 se detallan los experimentos realizados y se muestran los resultados obtenidos. Finalmente en 4 recopilamos sobre lo discutido en este trabajo en la forma de conclusiones y posible líneas futuras de trabajo.

2 DESCRIPCIÓN DEL PROBLEMA

Existen diversas situaciones del mundo real en donde la utilización de datos con una imprecisión intrínseca es implícita. En [12] se mencionan varios ejemplos algunos de los cuales reproducimos a continuación:

- Dispositivos digitales de medición con un número de cifras significativas determinado. Por ejemplo, una báscula digital con un sólo decimal para indicar el peso en kilogramos. Evidentemente, al pesar objetos desde $x.00...$ hasta $x.09...$ la báscula indicará la misma medida.
- Dispositivos que proporcionan una indicación de la tolerancia de la medición asociada a cada nivel

de confianza. Por ejemplo, un GPS, dispositivo que proporciona la posición del receptor sobre la corteza terrestre con la tolerancia (CEP, Círculo de Error Probable) a distintos niveles de probabilidad. Típicamente se da un CEP de 3 metros al 50% y 5 metros al 90%. Esto quiere decir que con la probabilidad mencionada el receptor estará *realmente* dentro de un círculo del radio correspondiente, centrado en la posición *medida*.

- Observación por parte de un ser humano de un instrumento analógico. Por ejemplo los instrumentos con indicación óptica y escala, todos ellos son susceptibles de ofrecer distintas medidas en función del ángulo de observación, debido al efecto del paralaje.

Cuando se tratan datos de esta clase mediante algoritmos que no los convierten de algún modo en datos nítidos, es posible obtener resultados que a su vez sean imprecisos. Si se deben comparar los resultados obtenidos por distintos algoritmos de este tipo, surge la necesidad de utilizar las herramientas estadísticas desarrolladas para este tipo de datos.

El estudio de los tests estadísticos y diseños experimentales para datos crisp ha sido objeto de estudios pormenorizados por parte de la comunidad de Machine Learning recientemente [10]. En el caso de datos de baja calidad o con imprecisión, el material bibliográfico es más escaso. Uno de los artículos más exhaustivos y extensos sobre el tema en general es [8]. En [7] se muestra como extender un test no paramétrico para poder ser usado con datos intervalares o fuzzy, de modo que los resultados del mismo sean un intervalo o un conjunto borroso, respectivamente. Este trabajo guarda cierta relación con [5], en el que se define el concepto de región de confianza para datos borrosos. Por otra parte en [4] se propone un método para defuzzificar p-valores borrosos y en [6] se da una definición de potencia para test de hipótesis para datos con imprecisión. Un texto completamente dedicado al análisis de datos fuzzy es [1], en donde se recopilan una serie de trabajos en los que se muestra como realizar tests de hipótesis con datos fuzzy, regresión, etc. En [14] se muestra una estimación intervalar de la media de una población con datos imprecisos.

La obtención de modelos y clasificadores a partir de datos de este tipo tiene sus problemas asociados, sin embargo, en este trabajo, sólo nos ocuparemos de la etapa de test. Es decir, como obtener una muestra del error del clasificador o modelo que responda a la cualidad imprecisa de los datos y como utilizar las herramientas estadísticas existentes para comparar distintos clasificadores o modelos. Es decir, en primer lugar obtendremos una muestra del error de test de

cada clasificador, de modo que cada elemento de las muestras sea un intervalo. En segundo lugar, compararemos esas muestras intervalares mediante un test de hipótesis adaptado a datos de este tipo. Este test de hipótesis proporcionará un intervalo de p-valores y analizaremos su potencia y error de Tipo I, los cuales serán también intervalares [6].

3 ESTUDIO EXPERIMENTAL

3.1 METODOLOGÍA

Con el fin de estudiar el comportamiento de las técnicas estadísticas existentes más representativas para evaluar modelos y clasificadores sobre datos con imprecisión se han realizado una serie de experimentos con un dataset sintético parametrizable. Este dataset se ha construido partiendo de la idea original de Haykin [9], consistente en dos gaussianas solapadas. De esta forma se han construido dos muestras de tamaño n , la primera se compone de pares (x^1, y^1) en donde x^1 sigue una $N(x_c^1, v^1)$ e y^1 sigue una $N(y_c^1, v^1)$ (siendo $(x^1$ e $y^1)$ independientes) y la segunda de pares (x^2, y^2) en donde x^2 sigue una $N(x_c^2, v^2)$ e y^2 sigue una $N(y_c^2, v^2)$ ($(x^1$ e $y^1)$ también independientes). Las dos muestras se concatenan y los índices de la muestra resultante se permutan. Sobre la muestra con estos índices permutados se aplica 10-cv, con el fin de obtener una muestra de tamaño 10 del error de los clasificadores. Más adelante explicaremos como obtener muestras del error intervalares, considerando una tolerancia en los datos de test.

Esto nos ha permitido realizar experimentos sobre condiciones controladas:

- La dificultad del problema (para el test) se gradúa mediante la distancia entre los centros de dichas gaussianas.
- La tolerancia de los datos de test incide en las diferencias entre el error máximo y mínimo, a mayor tolerancia mayores son las diferencias entre estos, lo que a su vez incide sobre la diferencia entre los p-valores máximos y mínimos.
- Es posible establecer a priori una serie de clasificadores que se puedan ordenar según su error para este problema concreto. Para este trabajo hemos escogido los clasificadores lineal, cuadrático y el óptimo bayesiano. No existen diferencias significativas entre el cuadrático y el óptimo y si entre el lineal y el óptimo. De esta forma se puede estimar el error de Tipo I comparando los test cuadrático y óptimo Bayesiano (H_0 cierta) y el de Tipo II (y por lo tanto la potencia) comparando el lineal y el óptimo (H_0 falsa).

En el caso del dataset sintético utilizado, el error de test máximo y mínimo se determina mediante un análisis de Montecarlo, como se sugiere en [8]. Sean $\{(x_1, y_1, c_1), (x_2, y_2, c_2), \dots, (x_i, y_i, c_i), \dots, (x_k, y_k, c_k)\}$ los datos de la partición de test, en donde cada (x_i, y_i) es un punto que pertenece a la clase c_i , en el caso de este trabajo $c_i \in \{1, 2\}$. A cada (x_i, y_i) se le añade una cantidad aleatoria dentro de la tolerancia especificada, de modo que se obtiene $(x_i + d_{x_i}, y_i + d_{y_i})$ con $d_{x_i} \in (-t, t)$ y $d_{y_i} \in (-t, t)$. Se evalúa el clasificador mediante la partición de test modificada de esta forma y se obtiene el error de test, se actualiza el máximo y el mínimo para esa partición. Este proceso se realiza un cierto número (evidentemente suficientemente elevado) de veces.

En lo que se refiere al cálculo de los p-valores máximo y mínimo, hemos utilizado el test de Wilcoxon adaptado a datos intervalares usando el método que se sugiere en la referencia citada.

Dadas dos muestras intervalares $X^1 = \{(e_{min\ 1}^1, e_{max\ 1}^1), (e_{min\ 2}^1, e_{max\ 2}^1), \dots, (e_{min\ n}^1, e_{max\ n}^1)\}$, $X^2 = \{(e_{min\ 1}^2, e_{max\ 1}^2), (e_{min\ 2}^2, e_{max\ 2}^2), \dots, (e_{min\ n}^2, e_{max\ n}^2)\}$ se enfrenta el cálculo de los p-valores máximo y mínimo como un proceso de optimización de la función $W(e_1^1, e_2^1, \dots, e_n^1, e_1^2, e_2^2, \dots, e_n^2)$ que devuelve el p-valor correspondiente al test de Wilcoxon para las muestras $\{e_1^1, e_2^1, \dots, e_n^1\}, \{e_1^2, e_2^2, \dots, e_n^2\}$.

El objetivo es encontrar los valores nítidos $X_{min}^1 = \{e_{min\ 1}^{*1}, e_{min\ 2}^{*1}, \dots, e_{min\ n}^{*1}\} / e_{min\ i}^{*1} \in (e_{min\ i}^1, e_{max\ i}^1)$, $X_{min}^2 = \{e_{min\ 1}^{*2}, e_{min\ 2}^{*2}, \dots, e_{min\ n}^{*2}\} / e_{min\ i}^{*2} \in (e_{min\ i}^2, e_{max\ i}^2)$, que proporcionan el valor mínimo de W y $X_{max}^1 = \{e_{max\ 1}^{*1}, e_{max\ 2}^{*1}, \dots, e_{max\ n}^{*1}\} / e_{max\ i}^{*1} \in (e_{min\ i}^1, e_{max\ i}^1)$, $X_{max}^2 = \{e_{max\ 1}^{*2}, e_{max\ 2}^{*2}, \dots, e_{max\ n}^{*2}\} / e_{max\ i}^{*2} \in (e_{min\ i}^2, e_{max\ i}^2)$ que proporcionan el valor máximo de la función W .

En [7] se sugiere de nuevo emplear el Método de Montecarlo para este proceso de optimización y en [6] se comenta la dificultad del problema de optimización. Por esta razón, en este trabajo hemos escogido utilizar el método L-BGFS-B [2] del cual existe una implementación para el paquete R en lugar del método de Montecarlo. El motivo es que para muestras de tamaño 10, como las que se obtienen al aplicar 10-cv, el número de variables que intervienen es 20, con lo cual el problema es abordable por este método en menos tiempo del que se necesitaría usando Montecarlo.

Los valores de los p-valores máximos y mínimos obtenidos se utilizan para estimar la potencia y el error máximos y mínimos. La estimación de la potencia se realiza como se propone en [6] para este caso concreto de imprecisión.

Cada vez que se comparan los clasificadores lineal y

exacto (H0 falsa) se estiman los p-valores máximos y mínimos. La fracción de veces en que el p-valor máximo es mayor que el nivel escogido es la potencia máxima. La fracción de veces en que el p-valor mínimo es mayor que el nivel escogido es la potencia mínima. Análogamente, cuando se comparan los clasificadores cuadrático y exacto (H0 cierta), la fracción de veces en que el p-valor mínimo es menor que el nivel es el error de Tipo I máximo y la fracción de veces en que el p-valor máximo es menor que el nivel es el error de Tipo I mínimo.

A modo de resumen gráfico de lo que se pretende realizar, véase la figura 1. En dicha figura se representan parte de los puntos del conjunto de datos utilizado, así como las superficies de decisión, para los clasificadores lineal (izquierda) y cuadrático (derecha). En cada figura se muestra, alrededor de cada punto un cuadrado del tamaño de la tolerancia. En la parte superior la tolerancia es de 0.05 y en la inferior de 0.1. Como puede verse, debido a la tolerancia de los datos, durante el proceso de determinación del error de test máximo y mínimo, es posible que los puntos utilizados para estimar el error estén a uno u otro lado de la superficie de decisión. Cuanto mayor es la tolerancia, mayor será el número de puntos susceptibles de estar a uno u otro lado de la superficie de decisión y por lo tanto la diferencia entre el error máximo y mínimo será mayor. Consiguientemente la diferencia entre los p-valores menor y mayor también será mayor.

De esta forma, se pueden poner de relieve diferencias entre clasificadores con un error nítido similar cuando la superficie de decisión de uno de ellos pasa más cerca de zonas con mayor densidad de ejemplos [13].

3.2 RESULTADOS

En la tabla 1 se detallan los parámetros de los experimentos realizados. Como puede observarse, se ha generado el dataset con puntos procedentes de dos Gaussianas bidimensionales con matrices de covarianza 1.0,0,0,1.0) y (2.0, 0,0,2.0), siendo el dataset balanceado. La tolerancia utilizada varía desde 0.05 a 0.10, en pasos de 0.01. El centro de una de las Gaussianas está fijo en el punto (0,0) mientras que el otro se mueve de (2.0,0) a (2.1,0) y finalmente a (2.2,0).

En la tabla 2 se muestran las estimaciones de potencia y error de Tipo I, máximos y mínimos, que se han obtenido para las combinaciones de parámetros de la tabla 1. Realmente, para analizar los resultados de forma objetiva, habría que utilizar alguna herramienta estadística clásica a los resultados de potencia y error de tipo I y para ello se necesitaría un conjunto de resultados mayor. Como no es el caso se tratará de extraer alguna conclusión cualitativa con los resultados de que

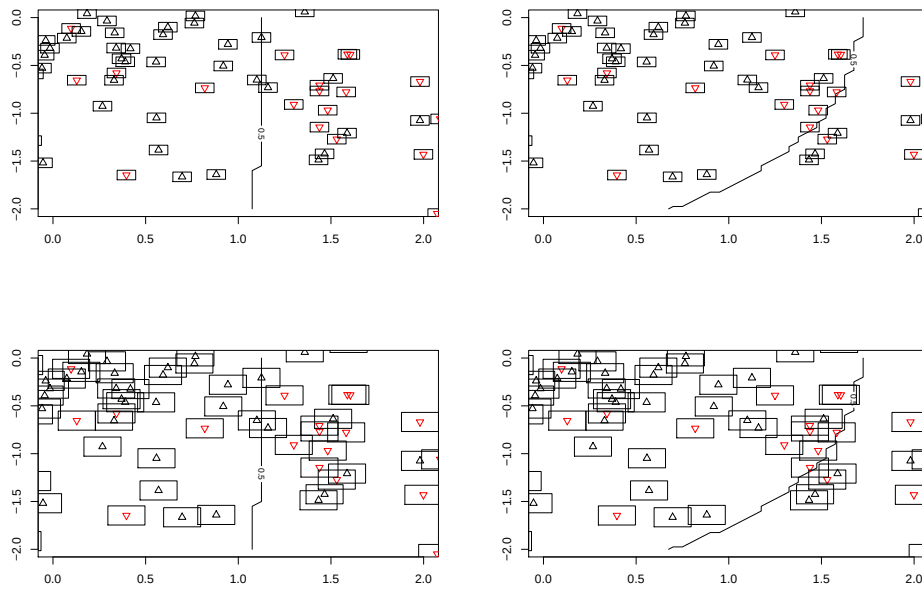


Figura 1: Ejemplo gráfico de la influencia de la tolerancia en el conjunto de test. Arriba tolerancia 0.05, abajo tolerancia 0.01. Derecha clasificador cuadrático, izquierda clasificador lineal. Los puntos están etiquetados con triángulos que apuntan hacia arriba o hacia abajo según la clase a la que pertenezcan. Obsérvese como al aumentar la tolerancia el número de puntos imprecisos susceptibles de intersecar la superficie de decisión aumenta.

Tabla 1: Valores de los parámetros utilizados en los experimentos

Rep. Mont.	50
Rep. 10-cv	50
Var. Gaus.	1.0 y 2.0
Centros Gaus.	(0,0) y (2.0,0), (2.1,0), (2.2,0)
Tolerancia	0.05, 0.06, 0.07, 0.08, 0.09, 0.10
Tam. dataset	500+500

se dispone.

Si se examinan las primeras cuatro columnas de la tabla 2, se puede observar como el intervalo obtenido para la potencia se hace gradualmente mayor. Este comportamiento es coherente con la variación de la tolerancia, como se dijo en la sección anterior, cuanto mayor es la tolerancia más fácil es (durante el proceso de optimización que calcula los p-valores) que los datos (imprecisos) del conjunto de test intersequen con la superficie de decisión.

En cuanto al error de Tipo I, no se aprecia variación alguna relacionada con las combinaciones de parámetros tolerancia vs. distancia entre centros. Como es habitual, se aprecia un incremento del error de Tipo I

cuando el nivel de significación aumenta.

Si se examina la variación de los resultados a través de las cuatro primeras filas, se observa como aumenta el tamaño del intervalo para la potencia a medida que las gaussianas están más separadas. Parece coherente con la experimentación planteada ya que las diferencias entre los clasificadores lineal y cuadrático disminuyen al estar las gaussianas más separadas. De forma dual a este efecto, cuanto más separadas están las gaussianas, menos errores de Tipo I se producen. Los efectos comentados se degradan rápidamente cuando la tolerancia de los datos aumenta, obsérvese las siguientes cuatro filas.

Todos estos efectos son más difíciles de observar cuando se examinan los resultados variando simultáneamente filas y columnas. Es posible que los resultados no sean concluyentes debido a que el número de repeticiones en el análisis de Montecarlo del error máximo y mínimo deba de ser más elevado.

4 CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha mostrado un estudio experimental preliminar en el que las técnicas más recientes, que

Tabla 2: Potencias y Errores de Tipo I, máximos y mínimos para cada combinación de parámetros, a los niveles 0.1, 0.05 y 0.01

t=0.03											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.840	1.000	0.000	0.380	0.960	1.000	0.000	0.300	0.260	0.980	0.000	0.100
1.000	1.000	0.000	0.380	1.000	1.000	0.000	0.300	0.760	1.000	0.000	0.100
1.000	1.000	0.000	0.920	1.000	1.000	0.000	0.880	0.920	1.000	0.000	0.820
t=0.04											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.9800	1.000	0.000	0.440	1.000	1.000	0.000	0.100	0.260	0.960	0.000	0.040
1.000	1.000	0.000	0.440	1.000	1.000	0.000	0.100	0.760	1.000	0.000	0.040
1.000	1.000	0.000	0.920	1.000	1.000	0.000	0.700	0.920	1.000	0.000	0.420
t=0.05											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.033	1.000	0.000	0.100	0.133	1.000	0.000	0.400	0.000	1.000	0.000	0.1667
0.667	1.000	0.000	0.100	0.900	1.000	0.000	0.400	0.433	1.000	0.000	0.1667
0.800	1.000	0.000	0.800	0.933	1.000	0.000	0.400	0.700	1.000	0.000	0.867
t=0.06											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.533	1.000	0.000	0.233	0.000	0.933	0.000	0.067	1.000	1.000	0.000	0.300
0.967	1.000	0.000	0.233	0.000	1.000	0.000	0.067	1.000	1.000	0.000	0.300
1.0000	1.000	0.000	0.900	0.000	1.000	0.000	0.933	1.000	1.000	0.000	0.833
t=0.07											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.067	1.000	0.000	0.233	0.400	1.000	0.000	0.300	0.000	1.000	0.000	0.267
0.567	1.000	0.000	0.233	0.867	1.000	0.000	0.300	0.033	1.000	0.000	0.267
0.767	1.000	0.000	0.733	0.967	1.000	0.000	0.800	0.233	1.000	0.000	0.767
t=0.08											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.000	1.000	0.000	0.133	0.000	1.000	0.000	0.067	0.000	1.000	0.000	0.167
0.233	1.000	0.000	0.133	0.000	1.000	0.000	0.067	0.233	1.000	0.000	0.167
0.233	1.000	0.000	0.9	0.167	1.000	0.000	0.733	0.600	1.000	0.000	0.867
t=0.09											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.000	1.000	0.000	0.133	0.267	1.000	0.000	0.100	0.000	1.000	0.000	0.367
0.200	1.000	0.000	0.133	0.800	1.000	0.000	0.100	0.167	1.000	0.000	0.367
0.400	1.000	0.000	0.867	0.967	1.000	0.000	0.833	0.333	1.000	0.000	0.933
t=0.10											
xc1=2.0				xc1=2.1				xc1=2.2			
Potencia		Err. I		Potencia		Err. I		Potencia		Err. I	
min	max	min	max	min	max	min	max	min	max	min	max
0.967	1.000	0.000	0.133	0.267	1.000	0.000	0.267	0.200	1.000	0.000	0.200
1.000	1.000	0.000	0.133	0.733	1.000	0.000	0.267	0.767	1.000	0.000	0.200
1.000	1.000	0.000	0.933	0.800	1.000	0.000	0.767	0.933	1.000	0.000	0.900

extienden los test de hipótesis habituales a datos intervalares, se aplican a una experimentación de Machine Learning con datos sintéticos. Variando de forma controlada los parámetros de la experimentación se ha comprobado (informalmente) que los resultados obtenidos, en lo que se refiere a error de Tipo I y potencia, parecen ser coherentes con la variación en tolerancia de los datos de test y con la dificultad del problema en si, el grado de solapamiento entre las gaussianas. De esta forma, se puede afirmar que un análisis estadístico de este tipo puede complementar las propuestas de los trabajos en los que se usan datos de baja calidad con técnicas de Soft computing [12] [13].

En relación a los métodos usados para las estimaciones del error de test máximo y mínimo y para los p-valorés máximos y mínimos, es necesario afinar lo más posible estos, ya que la estimación de la potencia y del error de Tipo I se cimentan en estas estimaciones. En este sentido se baraja sustituir el método de optimización "L-BFGS-B" por otro basado en un algoritmo genético [11]. Otro factor a estudiar es el número de particiones empleadas en la validación cruzada.

Agradecimientos

Este trabajo ha sido subvencionado por el Ministerio de Ciencia y Tecnología en el marco del proyecto TIN2008-06681-C06-04.

Referencias

- [1] C. Bertoluzza, M. A. Gil, D. A. Ralescu (Eds.) Statistical Modeling, Analysis and Management of Fuzzy Data. Series: Studies in Fuzziness and Soft Computing , Vol. 87 2002, XIV
- [2] R. H. Byrd, P. Lu and J. Nocedal. A Limited Memory Algorithm for Bound Constrained Optimization, (1995), SIAM Journal on Scientific and Statistical Computing , 16, 5, pp. 1190-1208.
- [3] O. Cordon, F. A. C. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, Genetic fuzzy systems. New developments, *Fuzzy Sets Syst.*, vol. 141, no. 1, pp. 173, 2004.
- [4] I. Couso, L. Sánchez. Defuzzification of Fuzzy p-Values. *Advances in Soft Computing* (2008) 126-132
- [5] I. Couso, L. Sánchez. Inner and outer fuzzy confidence regions determined by low quality data. Eurofuse Workshop 09: Preference Modelling and Decision Analysis
- [6] I. Couso. Como Hacer Inferencia Estadística a partir de datos de baja calidad. ESTYLF 2010.
- [7] T. Denoeux, M. H. Masson, P.A. Hbert, Nonparametric Rank-Based Statistics and Significance Test for Fuzzy Data, *Fuzzy Sets and Systems*, 153 (2005) 1-28
- [8] S. Ferson, V. Kreinovich, J. Hajagos, W. Oberkampf, L. Ginzburg. Experimental Uncertainty Estimation and Statistics for Data Having Interval Uncertainty. Technical Report SAND2007-0939. Sandia National Laboratories, Albuquerque, New Mexico
- [9] S. Haykin. Neural Networks, A Comprehensive Foundation. Prentice Hall. 1999
- [10] S. García, F. Herrera. An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. *Journal of Machine Learning Research* 9 (2008) 2677-2694
- [11] W. R. Mebane Jr., J. S. Sekhon. The rgenoud Package. 2002.
- [12] L. Sánchez, I. Couso. Advocating the use of Imprecisely Observed Data in Genetic Fuzzy Systems. em IEEE Transactions on Fuzzy Systems 15:4 (2007) 551-562
- [13] L. Sánchez, J. Otero. Learning Fuzzy Linguistic Models from Low Quality Data by Genetic Algorithms. Proceedings of the 16th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE07). London (United Kingdom, 2007) 1921-1926
- [14] S.H. Ong, Rahul Mukerjee. Interval estimation of the population mean under model uncertainty: Robust versus empirical statistics. *Journal of Statistical Planning and Inference*, Volume 138, Issue 12, 1 December 2008, Pages 3696-3704.

STATISTICAL PREFERENCE AS A COMPARISON METHOD OF TWO IMPRECISE FITNESS VALUES

Ignacio Montes¹ Davide Martinetti¹ Susana Díaz¹ Susana Montes¹

¹ Dpto. Estadística e I.O. y D.M., Universidad de Oviedo,
{uo182551,martinettidavide.uo,diazsusana,montes}@uniovi.es

Abstract

The fitness value of a knowledge base (KB) can be unknown and only some imprecise information about it can be obtained. In some cases this information is given by means of an interval where we know the fitness is contained. Thus, the comparison of two randomly distributed intervals is necessary in this context in order to be able to determine the preferences among individuals.

This contribution is a first approach to the use of statistical preference as a tool to compare this kind of intervals. We consider the probabilistic relation associated to the stochastic comparison of every pair of intervals and we study the cycle-transitivity of this relation. The defuzzification of this probabilistic relation, that is, the statistical preference relation, is studied and some properties are obtained. Our studies are particularly detailed for the case of the uniform distribution.

Keywords: vague data, fitness value, probabilistic relation, transitivity, statistical preference.

1 INTRODUCTION

Genetic algorithms are a powerful tool, capable of solving complex optimization tasks such as learning or tuning fuzzy rule bases and their corresponding fuzzy partitions (see [1]). When a genetic algorithm is used to generate or adapt a fuzzy system, the technique is referred to as a Genetic Fuzzy System (GFS) [2]. The use of GFSs has widely accepted, given that these algorithms are robust and can search efficiently large solution spaces (see [18]).

Although in this context the linguistic granules or information are represented by fuzzy sets, the input data and the output results are usually crisp [7]. However, some recent papers (see [12, 13, 14, 15]) have dealt with fuzzy-valued data to learn and evaluate GFS. In that approach the function that quantifies the optimality of a solution in the genetic algorithm, that is, the fitness function, is fuzzy-valued. In particular, in [15], the authors have considered that the fitness values are partly unknown, since there only exists interval valued information about them. In this context some kind of order between two fitness values is necessary for determining if one individual precedes the other. Since the information about the fitness values is imprecise and given by means of intervals, a procedure for comparing two intervals is required. Initially, this procedure was based on estimating and comparing two probabilities [15]. In this work we will consider a more general and flexible way to compare two intervals which is based on a probabilistic relation [5, 6, 9]. When the defuzzification is required, we will consider the statistical preference [6, 10], which is obtained as a cut of this relation.

In this contribution we study how these concepts are applied to compare two intervals, which represent an imprecise information about the fitness valued of two KBs. In particular, we will not assume knowledge of the joint distribution for the two fitness values and then the uniform distribution is used. Not only the independent, but also the countermonotone and comonotone cases (see [4]) are considered for defining the joint distribution. The assumption of a uniform distribution is not an artificial requirement and it can be considered in many situations as a consequence of lack of information (see, for instance, [15, 17]). When this distribution is considered, we will obtain the specific expression of the associated probabilistic and fuzzy relations. Since transitivity is a very simple, but also very important property when comparing elements, we will study whether and how the transi-

tivity propagates between these two types of relations. Moreover, we will generalize the concept of statistical preference, as a defuzzification of the probabilistic relation and we will study the general connection between this way to compare random elements and the classical stochastic dominance. Finally we study in detail the case of the uniform distribution.

The work is organized as follows: Section 2 collects some definitions and notions involved in following sections. In Section 3 we include the results obtained about the transitivity of the fuzzy and the probabilistic relations associated to two imprecise fitness values. In Section 4 we study the generalization of the concept of statistical preference, its connections with stochastic dominance and a particular analysis for the uniform distribution. The last section contains some final remarks.

2 BASIC CONCEPTS

Let us consider two fitness values θ_1 and θ_2 of two Fuzzy Systems in a regression problem. For instance, θ_1 and θ_2 are the mean squared errors of two Fuzzy Rule Based Systems (FRBS) on the same training set. In many situations, these values θ_1 and θ_2 are unknown, but we have some imprecise information about them. Thus, we cannot know the value of θ_1 and θ_2 , but we know two intervals where each of them is contained. These intervals can be obtained by means of a fuzzy generalization of the mean squared errors (for a more detailed explanation, see Sections 4 and 5 in [15]) and they will be denoted by $FMSE_1$ and $FMSE_2$, respectively.

If these two intervals are disjoint, then we have not any problem for determining the preferred interval and therefore the decision is trivial. The problem arises when the intersection is non-empty. In that case, a prior knowledge about the probability distribution of the fitness $P(\theta_1, \theta_2)$ can be considered. In that situation, a decision rule considered in [15] was to decide that $FMSE_1 \succ FMSE_2$ if and only if

$$P((\theta_1, \theta_2) : \theta_1 < \theta_2) > P((\theta_1, \theta_2) : \theta_1 \geq \theta_2).$$

However, in that approach we consider a crisp order between the intervals, but if we are in a fuzzy context, with imprecise data, some kind of gradual comparison could be more appropriate as the starting point of the comparison. Thus, we will use multivalued (also called fuzzy) relations to compare the intervals, that is, binary relations whose image is the real interval $[0, 1]$. These relations express the stochastic dependence between the alternatives with any value in that interval $[0, 1]$. The closer the value to 0, the

weaker the stochastic dependence between the alternatives. In [9] a fuzzy relation was considered and it can be defined, by using the usual notation in this context, by $R(FMSE_1, FMSE_2) = 1$ if $P((\theta_1, \theta_2) : \theta_1 > \theta_2) \geq P((\theta_1, \theta_2) : \theta_1 < \theta_2)$ and $R(FMSE_1, FMSE_2) = 1 + P((\theta_1, \theta_2) : \theta_1 > \theta_2) - P((\theta_1, \theta_2) : \theta_1 < \theta_2)$ otherwise.

Let us remark that R is reflexive (for any $FMSE_i$ we have that $R(FMSE_i, FMSE_i) = 1$) and strongly complete ($R(FMSE_1, FMSE_2) = 1$ or $R(FMSE_2, FMSE_1) = 1$), so that, any pair of fitnesses can be compared.

A different way to obtain a gradual comparison between fitness values is by means of the probabilistic relations. Let us recall that given a set of alternatives A , a probabilistic relation in A is a mapping $Q : A \times A \rightarrow [0, 1]$ such that $Q(a, b) + Q(b, a) = 1$ for every pair of alternatives a and b in A . These relations are sometimes called reciprocal or ipsodual relations. The interpretation of a probabilistic relation is quite different from the interpretation of a fuzzy relation. If Q is a probabilistic relation, $Q(a, b) = 1$ expresses that alternative a is totally preferred to b . But the value 0 does not mean absence of connection. For a probabilistic relation Q , $Q(a, b) = 0$ is identified with a clear preference of b over a . It is equivalent to $R(b, a) = 1$ and $R(a, b) = 0$. It also holds that $Q(a, b) = \frac{1}{2}$, reflects indifference between both alternatives.

De Schuymer et al. introduced in [6] the probabilistic relation generated by a collection of dice. The collection of dice is called a discrete dice model for the probabilistic relation defined. The definition generalized to a set of random variables can be seen in [5]. In our context it can be expressed as follows:

$$Q(FMSE_1, FMSE_2) = P((\theta_1, \theta_2) : \theta_1 > \theta_2) + \frac{1}{2}P((\theta_1, \theta_2) : \theta_1 = \theta_2).$$

From this probabilistic relation, we can obtain a way to compare two fitness values (the general definition was given in [5]). Thus, we say that $FMSE_1$ is statistically preferred to $FMSE_2$ if $Q(FMSE_1, FMSE_2) > \frac{1}{2}$. We will denote it $FMSE_1 >_{SP} FMSE_2$. $FMSE_1$ and $FMSE_2$ are statistically indifferent if $Q(FMSE_1, FMSE_2) = \frac{1}{2}$. We will denote $FMSE_1 \geq_{SP} FMSE_2$ if $Q(FMSE_1, FMSE_2) \geq \frac{1}{2}$.

This way to compare two random elements can be seen as an alternative to the classical stochastic dominance [8]. Let us recall that stochastic dominance does not take into account the possible relationship between the random elements being compared, but statistical preference depends on the stochastic dependence between the random elements. We recall that for every joint distribution function F , there is a copula C that con-

nects the marginal probability distribution functions (F_1 and F_2): $F(x, y) = C(F_1(x), F_1(y))$. Therefore statistical preference depends on the copula that connects the random elements. We recall that a copula (see for example [11]) is an operation $C : [0, 1]^2 \rightarrow [0, 1]$ satisfying

- $C(x, 0) = C(0, x) = 0$ for all $x \in [0, 1]$,
- $C(x, 1) = C(1, x) = x$ for all $x \in [0, 1]$,
- the property of moderate growth:
 $C(x_1, y_1) + C(x_2, y_2) \geq C(x_1, y_2) + C(x_2, y_1)$
 for every $(x_1, x_2, y_1, y_2) \in [0, 1]^4$ such that $x_1 \leq x_2$
 and $y_1 \leq y_2$.

If C is the minimum operator, the two elements are said to be comonotonic and if $C(x, y) = \max(x + y - 1, 0)$ they are countermonotonic. The case C is the product is the usual case of independence.

In [9] it was proven that there is a one-to-one correspondence between these fuzzy relations (R) and these probabilistic relations (Q), but we should recall they are considering two different ways to measure the relationship degree. Thus, although an important property to provide coherence to relations is the transitivity, the definition has to be totally different for probabilistic and fuzzy relations.

For probabilistic relations, De Baets et al. [3] developed a general framework that includes as particular cases several types of transitivity for probabilistic relations. In [6] the same authors provide an example based on a game with dices where no one “classical” transitivity is satisfied, but a particular type of cycle-transitivity holds. Analogously to FG -transitivity developed by Switalski [16], cycle-transitivity is based on an upper and a lower bound function. But the value bounded is not the usual $Q(a, c)$, but the sum $Q(a, b) + Q(b, c) + Q(c, a)$.

Thus, in [3] they defined that a probabilistic relation Q is cycle-transitive with respect to the upper bound function U if for all a, b, c in the set of alternatives,

$$\alpha_{abc} + \beta_{abc} + \gamma_{abc} - 1 \leq U(\alpha_{abc}, \beta_{abc}, \gamma_{abc}),$$

where

$$\begin{aligned}\alpha_{abc} &= \min(Q(a, b), Q(b, c), Q(c, a)), \\ \beta_{abc} &= \text{median}(Q(a, b), Q(b, c), Q(c, a)), \\ \gamma_{abc} &= \max(Q(a, b), Q(b, c), Q(c, a)).\end{aligned}$$

and U is an upper bound function, that is, it fulfils

1. $U(0, 0, 1) \geq 0$,
2. $U(0, 1, 1) \geq 1$,
3. $U(\alpha, \beta, \gamma) + U(1 - \gamma, 1 - \beta, 1 - \alpha) \geq 1$.

The most important upper bounds are:

- $U_M(\alpha, \beta, \gamma) = \beta$,

- $U_P(\alpha, \beta, \gamma) = \alpha + \beta - \alpha\beta$,
- $U_L(\alpha, \beta, \gamma) = \min(\alpha + \beta, 1)$.

The notation is derived from the associated t-norms (see [3]). We briefly recall that a t-norm is a binary $[0, 1]^2 \rightarrow [0, 1]$ operator such that it is commutative, associative, non-decreasing in each argument and has 1 as neutral element. The three most common t-norms are the minimum defined as: $T_M(x, y) = \min(x, y)$, $\forall x, y \in [0, 1]$, the product t-norm defined as: $T_P(x, y) = x \cdot y$, $\forall x, y \in [0, 1]$, and the Łukasiewicz t-norm: $T_L(x, y) = \max(x + y - 1, 0)$, $\forall x, y \in [0, 1]$.

This topic is essential in the most usual definition of transitivity for a fuzzy relation: a fuzzy relation R is transitive with respect to the t-norm T (T -transitive, for short) if it holds that

$$T(R(a, b), R(b, c)) \leq R(a, c)$$

for any three alternatives.

3 TRANSITIVITY

In this section we will obtain the type of cycle-transitivity and the type of T -transitivity fulfilled by Q and R , respectively, if we assume $P(\theta_1, \theta_2)$ is obtained by copulation of two uniform distributions (see, for instance, [15, 17]). We will consider three different approaches to this problem: independence, comonotony and countermonotony. Moreover, by coherence with the studies developed in [4, 5], we will consider in this section that the two random elements are uniformly distributed in two intervals with the same width.

Thus, let $FMSE_1 = [a_1, a_1 + \lambda]$ and $FMSE_2 = [a_2, a_2 + \lambda]$ be two intervals where we know the fitnesses θ_1 and θ_2 of two KBs are content and let us consider a uniform distribution on each of them. As we commented in Section 2, the joint distribution is defined by means of a copula. In this work we will consider the minimum, product and Łukasiewicz copulas (which is the same of the minimum, product and Łukasiewicz t-norms) in order to define this joint distribution.

3.1 THE INDEPENDENT CASE

The case when the copula is the product was already studied in [5], where they obtained the expression of the probabilistic relation associated to this kind of random variables. Thus, in our context,

$$Q^P(FMSE_1, FMSE_2) = \begin{cases} 0 & \text{if } a_2 > a_1 + \lambda \\ \frac{(\lambda + a_2 - a_1)^2}{2\lambda^2} & \text{if } a_1 < a_2 < a_1 + \lambda \end{cases}$$

It also was proven in [5] that Q^P is cycle-transitive with respect to the upper bound function

$$U^P(\alpha, \beta, \gamma) = \begin{cases} \beta + \gamma - 1 + \frac{1}{2}(T_L(\sqrt{2(1-\beta)}, \sqrt{2(1-\gamma)}))^2, & \text{if } \beta \geq 1/2, \\ \alpha + \beta - \frac{1}{2}(T_L(\sqrt{2\alpha}, \sqrt{2\beta}))^2, & \text{if } \beta < 1/2. \end{cases}$$

In this case, by using [9], the expression of the associated fuzzy relation is:

$$R^P(FMSE_1, FMSE_2) = \begin{cases} 0 & \text{if } a_2 > a_1 + \lambda \\ 2(1 - \frac{(\lambda + a_1 - a_2)^2}{2\lambda^2}) & \text{if } a_2 < a_1 < a_2 + \lambda \\ 1 & \text{otherwise} \end{cases}$$

3.2 THE COMONOTONIC CASE

In the case the joint distribution function is obtained as the minimum between both marginal distribution functions, the expression of the probabilistic relation is crisp. In such situation, we can assure that it is always cycle-transitive with respect to any upper bound function, since it is transitive in the classical sense.

PROPOSITION 3.1 *Let $FMSE_1 = [a_1, a_1 + \lambda]$ and $FMSE_2 = [a_2, a_2 + \lambda]$ be two intervals where we know the fitnesses θ_1 and θ_2 of two KBs are content and let us consider a uniform distribution on each of them. If the joint distribution is obtained by means of the minimum copula, then the associated probabilistic relation Q^M is crisp with*

$$Q^M(FMSE_1, FMSE_2) = \begin{cases} 0 & \text{if } a_1 \leq a_2, \\ 1 & \text{if } a_1 > a_2. \end{cases}$$

Moreover Q^M is transitive.

In this case, the associated fuzzy relation R^M is also crisp (see [9]) and it is transitive.

3.3 THE COUNTERMONOTONIC CASE

Let us now consider the case the joint distribution is obtained by means of the Łukasiewicz copula.

PROPOSITION 3.2 *Let $FMSE_1 = [a_1, a_1 + \lambda]$ and $FMSE_2 = [a_2, a_2 + \lambda]$ be two intervals where we know the fitnesses θ_1 and θ_2 of two KBs are content and let us consider a uniform distribution on each of them. If the joint distribution is obtained by means of the Łukasiewicz copula, then the associated probabilistic relation Q^L is*

$$Q^L(FMSE_1, FMSE_2) = \begin{cases} \frac{a_1 - a_2 + \lambda}{2\lambda} & \text{if } a_2 < a_1 + \lambda, \\ 0 & \text{if } a_1 + \lambda \leq a_2. \end{cases}$$

Moreover, Q^L is cycle-transitive with respect to the upper bound function

$$U^L(\alpha, \beta, \gamma) = \max(\beta, 1/2).$$

In [9] it was proven that the cycle-transitivity with respect to U^L is equivalent to the T_L -transitivity of the associated fuzzy relation R^L . Thus,

COROLLARY 3.3 *Let $FMSE_1 = [a_1, a_1 + \lambda]$ and $FMSE_2 = [a_2, a_2 + \lambda]$ be two intervals where we know the fitnesses θ_1 and θ_2 of two KBs are content and let us consider a uniform distribution on each of them. If the joint distribution is obtained by means of the Łukasiewicz copula, then the associated fuzzy relation R^L is T_L -transitive.*

4 STATISTICAL PREFERENCE

As we showed in Section 2, the classical statistical preference [5] is defined as the 0.5-cut of the probabilistic relation Q . This defuzzification is very strict and we could consider appropriate any other cut as the minimum level of relationship. Thus, we could generalize this definition as follows.

DEFINITION 4.1 *Let X and Y be two random variables and let Q be its associated probabilistic relation. X is said to be α -statistical preferred or indifferent to Y , $X \geq_{\alpha SP} Y$, if it holds that $(X, Y) \in Q_\alpha$, where $Q_\alpha = \{(X, Y) : Q(X, Y) \geq \alpha\}$.*

It is immediate that α -statistical preference implies α' -statistical preference if $\alpha \geq \alpha'$, but the converse is not true.

The relationship of this concept with the stochastic dominance (see [8]) is developed in the following result.

PROPOSITION 4.2 *If $\alpha \leq 1/2$, then*

- *the first degree stochastic dominance implies α -statistical preference but the converse is not true;*
- *there is not any implication between the second degree stochastic dominance and the α -statistical preference.*

In the case $\alpha > 1/2$, even the first implication is not fulfilled and any relationship can be established.

Although we have considered here a more general definition, the most usual case (see, for instance, [5, 6, 10]) is the case of the 1/2-cut, which is simply called statistical preference. In that case, the meaning of this definition can be clarified by the following proposition.

PROPOSITION 4.3 *Let X and Y be two random variables. In general, it holds that*

$$X \geq_{SP} Y \implies P(X < Y) \leq \frac{1}{2}.$$

For continuous variables, the converse implication is also fulfilled.

Let us now present two results which can be use as a tool to manage statistical preference under translations and dilations. Both of them are natural properties for this relation.

PROPOSITION 4.4 *Let X and Y be two random variables. It holds that*

- $X \geq_{SP} Y \iff X - Y \geq_{SP} \underline{0}$.
- $X + Y \geq_{SP} X \iff Y \geq_{SP} \underline{0}$.
- $\lambda X \geq_{SP} \underline{\mu} \iff \begin{cases} X \geq_{SP} \underline{\mu/\lambda} & \text{if } \lambda > 0 \\ \underline{\mu/\lambda} \geq_{SP} X & \text{if } \lambda < 0 \\ \mu \leq 0 & \text{if } \lambda = 0 \end{cases}$

As a consequence of this result, we have

COROLLARY 4.5 *Let X be a random variable and let α be a constant. It holds that*

- $X \geq_{SP} X + \alpha \iff \alpha \leq 0$.
- $\lambda X \geq_{SP} \alpha X \iff \begin{cases} \underline{0} \geq_{SP} X & \text{if } \alpha > 1 \\ X \geq_{SP} \underline{0} & \text{if } \alpha < 1 \end{cases}$

In the following of the section, we will apply this definition in our context. Thus, we will consider two imprecise fitness values and we will obtain the requirement to be able to assure one of them is statistical preferred to the other. We will consider again a uniform distribution, that is, no any prior information about the distribution over the observed interval, but in this case we will not restrict our study to intervals with the same width, but we will consider the most general case.

Thus, $FMSE_1 = [a, b]$ and $FMSE_2 = [c, d]$ will denote now two intervals where we know the fitnesses θ_1 and θ_2 of two KBs are content. Let us assume a uniform distribution on each of them. We will consider again three possible ways to obtain the joint distribution: minimum, product and Łukasiewicz copulas. In these three cases we will obtain the condition over a , b , c and d to assure the statistical preference of the interval $FMSE_1$ to the interval $FMSE_2$. To do that, the expression of Q^P , Q^M and Q^L obtained in the previous section will be an essential part of the proof.

4.1 THE INDEPENDENT CASE

PROPOSITION 4.6 *Let $FMSE_1 = [a, b]$ and $FMSE_2 = [c, d]$ be two uniformly distributed intervals which represent the information we have about the fitnesses θ_1 and θ_2 of two KBs. If the joint distribution is obtained by means of the product copula, then $FMSE_1 \geq_{SP} FMSE_2$ if, and only if,*

$$\begin{cases} (b-c)^2 \geq (b-a)(d-c) & \text{if } a \leq c < b \leq d, \\ a+b \geq c+d & \text{if } a \leq c < d < b \\ a+b \geq c+d & \text{if } c < a < b \leq d, \\ (d-c)(b-a) \geq (d-a)^2 & \text{if } c < a < d \leq b. \end{cases}$$

4.2 THE COMONOTONIC CASE

PROPOSITION 4.7 *Let $FMSE_1 = [a, b]$ and $FMSE_2 = [c, d]$ be two uniformly distributed intervals which represent the information we have about the fitnesses θ_1 and θ_2 of two KBs. If the joint distribution is obtained by means of the minimum copula, then*

- *If $a \leq c < b \leq d$, then $FMSE_1 \not\geq_{SP} FMSE_2$.*
- *If $a \leq c < d < b$, $FMSE_1 \geq_{SP} FMSE_2$ if, and only if, $b(d-b+c) \geq a(c+d-a)$.*
- *If $c < a < d \leq b$, then $FMSE_1 \geq_{SP} FMSE_2$.*
- *If $c < a < b \leq d$, $FMSE_2 \geq_{SP} FMSE_1$ if, and only if, $3ad + (a-b)^2 + 2c^2 + bc \geq 2cd + 3ac + bd$.*

4.3 THE COUNTERMONOTONIC CASE

PROPOSITION 4.8

Let $FMSE_1 = [a, b]$ and $FMSE_2 = [c, d]$ be two uniformly distributed intervals which represent the information we have about the fitnesses θ_1 and θ_2 of two KBs. If the joint distribution is obtained by means of the Łukasiewicz copula, then $FMSE_1 \geq_{SP} FMSE_2$ if, and only if, $c(c-b-a) \geq d(d-a-b)$.

5 CONCLUDING REMARKS

In this work we have developed a complete study of a method to compare two random intervals, which can be applied in particular to compare the imprecise information about two fitness values of two KBs. This comparison is fuzzy and it is established by means of a probabilistic relation. When the appropriate cut is considered, we obtain the statistical preference as a defuzzification of this relation. In all our studies the particular expressions and results for the usual case of uniform distributions are obtained. Thus, for the probabilistic and fuzzy relations associated to any set

of interval ($FMSE_i$) we have characterized the transitivity fulfilled for these relations. For statistical preference, the requirements to be ordered two intervals are totally detailed.

As a future work, we would like to extend our studies in two different directions. The first one and simpler is an extension to other families of distributions. The second one would be to develop some parallel method to compare other kind of imprecise information, which could be determined by any fuzzy set instead of an interval.

Acknowledgements

The research reported on in this communication has been partially supported by Project FEDER-MEC-MTM2007-61193.

References

- [1] O. Cordon, F. Herrera, F. Gomide, F. Hoffmann, L. Magdalena. Ten years of genetic-fuzzy systems: a current framework and new trends. In: *Proceedings of Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, Vancouver - Canada, 2001, pp. 1241–1246.
- [2] O. Cordon, F. Herrera, F. Hoffmann, L. Magdalena. Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases. World Scientific Publishing, Sigapore, 2001.
- [3] B. De Baets, H. De Meyer, B. De Schuymer, S. Jenei. Cyclic evaluation of transitivity of reciprocal relations. *Social Choice and Welfare* **26** (2006) 217–238.
- [4] H. De Meyer, B. De Baets, B. De Schuymer. On the transitivity of the comonotonic and countermonotonic comparison of random variables. *Journal of Multivariate Analysis* **98** (2007) 177–193.
- [5] B. De Schuymer, H. De Meyer, B. De Baets. Cycle-transitive comparison of independent random variables. *Journal of Multivariate Analysis* **96** (2005) 352–373.
- [6] B. De Schuymer, H. De Meyer, B. De Baets, S. Jenei. On the cycle-transitivity of the dice model. *Theory and Decision* **54** (2003) 261–285.
- [7] F. Herrera. Genetic fuzzy systems: taxonomy and current research trends and prospects. *Evolutionary Intelligence* **1** (2008) 27–46.
- [8] H. Levy. *Stochastic dominance*. Kluwer Academic Publishers. 1998.
- [9] D. Martinetti, I. Montes, S. Díaz. Transitivity of the fuzzy and the probabilistic relations associated to a set of random variables. In: *Proc. EUROFUSE 2009 Workshop: Preference modelling and decision analysis*, Pamplona, Spain, 2009.
- [10] I. Montes, D. Martinetti, S. Díaz. On the statistical preference as a pairwise comparison for random variables. In: *Proc. EUROFUSE 2009 Workshop: Preference modelling and decision analysis*, Pamplona, Spain, 2009.
- [11] R. Nelsen. *An Introduction to Copulas*. Lecture Notes in Statistics. Vol. 139. Springer, New York.
- [12] L. Sánchez, I. Couso. Advocating the use of imprecisely observed data in genetic fuzzy systems. *IEEE Transactions on Fuzzy Systems* **15** (2007) 551–562.
- [13] L. Sánchez, I. Couso, J. Casillas. A multiobjective genetic fuzzy system with imprecise probability fitness for vague data. In: *Proc. 2006 IEEE Internat. Conf. on Evolutionary Fuzzy Systems*, Ambleside, UK, 2006, pp.131–137.
- [14] L. Sánchez, I. Couso, J. Casillas. Modeling vague data with genetic fuzzy systems under a combination of crisp and imprecise criteria. In: *Proc. 2007 IEEE Sympos. on Computational Intelligence in Multicriteria Decision Making*, Honolulu, USA, 2007, pp.30–37.
- [15] L. Sánchez, I. Couso, J. Casillas. Genetic learning of fuzzy rules based on low quality data. *Fuzzy Sets and Systems* **160** (2009) 2524–2552.
- [16] Z. Switalski. General transitivity conditions for fuzzy reciprocal preference matrices. *Fuzzy Sets and Systems* **137** (2003) 85–100.
- [17] J. Teich. Pareto-front exploration with uncertain objectives. In: *Proc. First Internat. Conf. on Evolutionary Multi-Criterion Optimization*, 2001, pp. 314–328.
- [18] Y. Yuan, H. Zhuang. A genetic algorithm for generating fuzzy classification rules. *Fuzzy Sets and Systems* **84** (1996) 1–19.

Representación de relaciones causales a través de un proceso semi-automático de recuperación

Cristina Puente¹

Alejandro Sobrino²

José Ángel Olivas³

¹ Dpto. de Sistemas Informáticos, ETS de Ingeniería, Universidad Pontificia Comillas, Madrid.
cpuente2@upcomillas.es

² Dpto. de Lógica y Filosofía Moral, Universidad de Santiago de Compostela, La Coruña.
alejandro.sobrino@usc.es

³ Dpto. de Tecnologías y Sistemas de la Información, Universidad de Castilla La-Mancha, Ciudad Real.
joseangel.olivas@uclm.es

Resumen

En este trabajo se presenta un sistema automático de recuperación de oraciones condicionales y causales para su posterior análisis y representación. El resultado del proceso es un esquema que puede servir para establecer nuevas conexiones entre conceptos, como por ejemplo a través de un grafo causal.

Palabras Clave: Sentencias causales y condicionales, grafos causales, representación de la causalidad.

1 INTRODUCCIÓN

La causalidad es un tema fundamental en cualquier campo del ámbito científico. En las ciencias empíricas, como la física, las relaciones causales son una manera típica de generar conocimiento y aportar explicaciones. Por ejemplo, cuando un físico cuántico calcula la probabilidad de que un átomo absorba un protón, está en realidad determinando la causa por la que el átomo salta a un nivel de energía. Es decir, intenta establecer una relación causa-efecto [7]. En las ciencias sociales, como en el derecho, la causalidad se utiliza para establecer una conexión entre conducta y responsabilidad, p. ej., como justificación en la imposición de una pena. Así, un jurado puede determinar si la negligencia de un médico en el tratamiento de la víctima fue la causa inmediata de su muerte y, en función de eso, condenarle o no.

Generalmente la causalidad es un tipo de relación entre dos entidades: causa y efecto. La causa provoca un efecto, y el efecto se deriva de una causa. Dado que la causalidad puede ser enunciada en términos de causas suficientes o necesarias, recuerda y tiene ciertos paralelismos con la

relación lógica de implicación, tradicionalmente definida en términos de condiciones necesarias y suficientes.

Condicionalidad y causalidad presentan analogías y divergencias. Hay muchas oraciones causales que pueden ponerse en términos condicionales. Las leyes científicas son un caso paradigmático: p. ej., la primera ley de la termodinámica dice que la causa de que la energía interna de un sistema cambie es que se realice trabajo sobre ese sistema o intercambie calor con otro. Esta ley puede parafrasearse en términos condicionales diciendo que si se realiza trabajo sobre un sistema o bien éste intercambia calor con otro, la energía interna del sistema cambiará. Pero hay enunciados causales que no son condicionales. P. ej., ‘La causa de la muerte de Kennedy fue un disparo’ no admite paráfrasis en términos condicionales (Si se dispara a Kennedy, se le mata), porque introduce una posibilidad que puede contradecir la factualidad de un hecho verdadero ya sucedido. Se le puede disparar a alguien y no matarle o morir por otra circunstancia (un infarto, p. ej.), pero no se puede hacer hipótesis sobre un hecho pasado, como el que atañe a este ejemplo. Los enunciados condicionales tampoco tienen siempre naturaleza causal. Muchas veces, sí, como ocurre en el enunciado condicional de la primera ley de la termodinámica antes aludida. Pero otras veces, no. P. ej., ‘si salto, me caigo’ expresa una relación entre un acto y una consecuencia más débil que una relación causal. En una relación causal, siempre que se da la causa, se da el efecto. Pero en el enunciado arriba proferido, por ser conjetural, deja abierta la posibilidad a que salte y caiga, pero también a que no salte y caiga o a que no salte y no caiga. Sólo si a posteriori se vincula con seguridad la caída al salto, y de manera predominante a él, saltar sería causa de la caída. Además, con las partículas *si... entonces* se pueden hacer extraños enunciados, muy infrecuentes en el lenguaje (casi experimentales, se podría decir) donde no hay relación alguna entre antecedente y consecuente, como ocurre, p. ej., en ‘si 2 es par, Madrid es capital de España’. Son los condicionales materiales,

que en ningún caso, obviamente, expresan vínculos causales.

Por tanto, aunque no se puede identificar completamente condicionalidad y causalidad, ambas nociones están bastante relacionadas. Dado que el objetivo de este trabajo tiene que ver con el rescate de oraciones causales en textos, será relevante detectar oraciones condicionales en textos, será relevante detectar oraciones condicionales que expresen posibles relaciones causales y oraciones típicamente causales en tanto incluyan léxico causal que las caracterice como tales. Para lograr este objetivo, este artículo está dividido en tres bloques. En el primero se presenta un método de extracción y clasificación automático de oraciones condicionales y causales a través de un analizador morfológico. En el segundo bloque se presenta un proceso relacionado con el anterior capaz de seleccionar del total de frases extraídas un conjunto relacionado con un concepto en particular, dando como resultado un esquema de representación. En el tercer bloque se describe la representación de este esquema a modo de grafo para permitir establecer hipótesis y relaciones que a priori no aparecen en el texto.

2 DETECCIÓN DE SENTENCIAS CONDICIONALES Y CAUSALES

Distintos textos procedentes de ámbitos novelísticos, periodísticos o científicos ofrecerán un banco de pruebas en el que identificar a las oraciones causales. Para ello se ha desarrollado un proceso específico para su extracción, análisis, tratamiento y representación.

Dado que en los textos científicos muchas sentencias condicionales son causales, el estudio de la causalidad que se hace en este trabajo se sirvió de las frases rescatadas con el programa descrito en [1], aunque su análisis excede el marco teórico allí considerado. Para ello se utilizó el analizador morfológico de libre distribución *Flex*¹ junto con el lenguaje de programación C, dando lugar a un proceso capaz de seleccionar oraciones de un texto que encajen dentro de unos patrones (figura 1).

Structure e2: if + present simple + future simple
 Structure e3: if + present simple + may/might
 Structure e4: if + present simple + must/should
 Structure e6: if + past simple + would + infinitive
 Structure e7: if + past simple + might/could
 Structure e8: if + past continuous + would + infinitive
 Structure e9: if + past perfect + would + infinitive
 Structure e10: if + past perfect + would have + past participle
 Structure e11: if + past perfect + might/could have + past participle
 Structure e12: if + past perfect + perfect conditional continuous
 Structure e13: if + past perfect continuous + perfect conditional
 Structure e16: if + past perfect + would + be + gerund
 Structure e23: for this reason, as a result
 Structure e24: due to, owing to
 Structure e26: provided that
 Structure e31: have something to do, a lot to do
 Structure e34: so that, in order that
 Structure e38: although, even though
 Structure e43: in case that, in order that
 Structure e44: on condition that, supposing that

Figura 1: Conjunto de patrones implementados.

En un principio se analizaron las formas en inglés pertenecientes al condicional en su forma más típica, *if x then y*, así como otras fórmulas equivalentes que pudieran dar lugar a este tipo de oraciones.

Para comprobar el nivel de fiabilidad de la aplicación, se analizaron manualmente una serie de documentos pertenecientes a diferentes ámbitos (50 páginas por categoría), y se calcularon las medidas de relevancia y precisión (en inglés, *recall*, *precision*). La relevancia *R*, es el número de oraciones causales correctamente clasificadas por el sistema dividido por el número de oraciones causales clasificadas manualmente. La precisión *P*, es el número de oraciones causales correctamente clasificadas por el proceso dividido por el total de sentencias devueltas. La *F-Measure*, que determina la relación entre la relevancia y precisión, viene dado por la fórmula $F = (2 * P * R) / (P + R)$, y sirve para controlar la importancia relativa de uno u otro parámetro.

Tabla 1: Resultados de clasificación y detección

Tipo de texto	Devueltas (Programa)	Clasificadas (Programa)	Clasificadas (Manual)
Científicos	62	52	80
Médicos (Medlars) ²	11	10	13
Novelas	22	12	37
Noticias (Reuters) ³	14	11	19
Evangelios	30	21	42

¹ <http://www.gnu.org>.

² <http://www.nlm.nih.gov/bsd/mmshome.html>

³ <http://www.daviddlewis.com/resources/testcollections/reuters21578/>

Tabla 2: Resultados de relevancia, precisión y *F-Measure*

Tipo de texto	Relevancia	Precisión	F-Measure
Científicos	0,65	0,84	0,73
Médicos (Medlars) ²	0,77	0,91	0,83
Novelas	0,32	0,55	0,41
Noticias (Reuters) ³	0,58	0,79	0,67
Evangelios	0,50	0,70	0,58

Los resultados indican que los textos científicos y médicos tienen unos mejores resultados en relevancia y precisión que el resto de dominios tratados, por lo que los experimentos a partir de este punto se efectuaron dentro del ámbito médico. En particular se seleccionaron cinco libros relacionados con el campo de la ginecología [2], [3], [5], [6], [8].

3 ESQUEMATIZACIÓN AUTOMÁTICA DEL GRAFO CAUSAL

Una vez que han sido seleccionadas y etiquetadas todas las oraciones condicionales de un documento, se ha diseñado un proceso que permita extraer aquellas relacionadas con un concepto en particular. Este concepto de entrada podrá proceder de una pregunta (por ejemplo en un sistema pregunta-respuesta), o ser formulado directamente al proceso.

El siguiente paso consistirá en la localización de todas aquellas oraciones condicionales previamente seleccionadas en las que se mencione el concepto solicitado. Se ha creado otra aplicación mediante el analizador morfológico *Flex* que se encarga de generar un fichero de texto plano con las frases seleccionadas.

Cada una de estas oraciones pasará directamente a una aplicación programada en C, donde se analizará si el concepto introducido en el proceso está dentro del contexto condicional de la oración. En caso afirmativo se presentará un esquema con las conjunciones, modificadores y demás términos relevantes para su posterior representación en el grafo.

Los pasos que sigue el algoritmo son los siguientes:

- Procesar una frase mediante un programa part-of-speech-tagger (POST) [4], para dividirla en lexemas.
- Localizar el concepto buscado dentro de la frase analizada.
- Localizar la partícula condicional o causal más próxima al concepto buscado y evaluar si pertenecen al mismo contexto (en base al número de palabras que los separan). Si no, eliminar la sentencia.

- Determinar si el concepto es causa o efecto dentro de la sentencia analizada para establecer la dirección de la flecha dentro del grafo.
- Localizar atributos y modificadores como adjetivos y adverbios cerca del concepto.
- Si el concepto buscado está antes de la conjunción causal/condicional, etiquetar el nodo como efecto y buscar el nodo causal junto con sus modificadores. En caso contrario etiquetar el nodo como causa y buscar el nodo efecto junto con sus modificadores.

El proceso desarrollado es capaz de simplificar una sentencia compleja, extrayendo los conceptos y modificadores que pudieran estar relacionados con el buscado.

La salida del programa consiste en un fichero plano que contiene un resumen esquematizado del contexto en el que se encuentra el concepto buscado dentro del ámbito causal. Por ejemplo, la siguiente sentencia: *Preterm rupture of membranes at 20 weeks or earlier is associated with a poor prognosis for miscarriage within five days of membrane rupture due to chorioamnionitis, and in the remaining 60 % of pregnancies, more than 50% of neonates die due to pulmonary hypoplasia*, daría como resultado el siguiente diagrama:

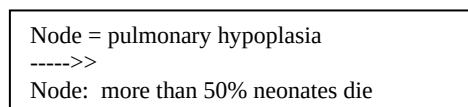


Figura 2: Salida esquematizada del programa.

La figura 2 refleja la manera más simple de extracción de una sentencia. La representación correspondiente a esta salida del programa se corresponde con la figura 3, donde el nodo con el concepto *pulmonary hypoplasia* apunta al nodo con el concepto *more than 50% neonates die*, o en otras palabras, *pulmonary hypoplasia* causa o provoca que *more than 50% neonates die*.

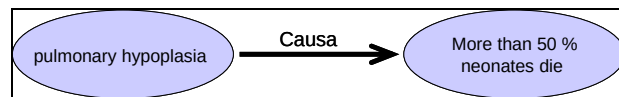


Figura 3: Parte del grafo correspondiente a la figura 2.

Generalmente, el proceso de representación es bastante más complicado que el reflejado en las figuras 2 y 3, por los modificadores asociados a cada uno de los nodos como se puede ver en los ejemplos expuestos a continuación.

4 REPRESENTACIÓN DEL GRAFO CAUSAL

Para probar el proceso de esquematización y representación del grafo, se introdujo en el sistema el concepto “*pulmonary hypoplasia*”, dando como resultado 28 sentencias condicionales (con contenido causal), de las cuales se han seleccionado cuatro para representar gráficamente:

In bilateral forms the prognosis is unfavorable in the majority of cases neonatal death is **due to** pulmonary hypoplasia as a consequence of the severe oligohydramnios. Meckel Gruber syndrome is invariably lethal **due to** the severe pulmonary hypoplasia induced by the oligohydramnios. Bilateral renal agenesis is always lethal **due to** the associated severe pulmonary hypoplasia. Doppler studies in fetuses with pulmonary hypoplasia **due to** multicystic kidneys or obstructive uropathy reported increased impedance to flow in the branches of the pulmonary artery consistent with high peripheral pulmonary vascular resistance.

Figura 4: Ejemplo de sentencias recuperadas.

A continuación, se procesaron estas oraciones con el POST [4], para posteriormente servir de entrada al proceso de esquematización de sentencias condicionales/causales, dando lugar a los resultados mostrados en la figura 5:

Node = pulmonary hypoplasia || **associated modifiers:** as consequence of severe oligohydramnios ----->> in majority of cases neonatal death //
Before node modifier: severe **Node** = pulmonary hypoplasia **associated modifiers:** by oligohydramnios -->> Gruber syndrome is invariably lethal //
Before node modifier: severe **Node** = pulmonary hypoplasia ----->> renal agenesis is always lethal //
 Doppler studies fetuses with **Node** = pulmonary hypoplasia <<----- **Node:** multicystic kidneys or : obstructive uropathy //

Figura 5: Salida del proceso de esquematización de las frases obtenidas en la figura 4.

Desde un punto de vista causal, las sentencias esquematizadas se pueden representar teniendo en cuenta la causa principal (en este caso *pulmonary hypoplasia*), el efecto que provoca, y en algunos casos como en el que se presenta, el mecanismo que se desencadena entre causa y efecto, como muestra la figura 6.

4.1. INTERPRETACIÓN DEL GRAFO

Según el grafo representado en la figura 6, se puede decir, de forma condicional que “*si la hipoplasia pulmonar es alta, puede provocar agenesia renal, la cual es siempre letal. La hipoplasia pulmonar como consecuencia de un nivel alto (severe) de oligohidramnios provoca el síndrome de Gruber, el cual es invariablemente letal. Como consecuencia de un nivel alto de oligohidramnios, la hipoplasia pulmonar causa en la mayoría de los casos la muerte neonatal.*”. O se puede elaborar una teoría de forma causal: “*La hipoplasia pulmonar severa causa el síndrome de Gruber, el cual es invariablemente letal. Al estar causada por un nivel alto de oligohidramnios (severe), la hipoplasia pulmonar provoca en la mayoría de los casos la muerte neonatal.*”.

5 CONCLUSIONES Y TRABAJO FUTURO

El proceso introducido en este trabajo, presenta un modelo automático de recuperación de oraciones causales y condicionales relacionadas con un determinado concepto. Este modelo será mejorado en trabajos futuros para establecer relaciones entre varios conceptos de manera automática, incluyendo la fase de representación, bien sea mediante un grafo causal u otro modelo más adecuado que permita ponderar este tipo de relaciones. También se pretende ampliar el ámbito de aplicación a otras áreas como por ejemplo el diagnóstico médico o medioambiental, ámbitos legales, educación, o incluso perfiles de usuario.

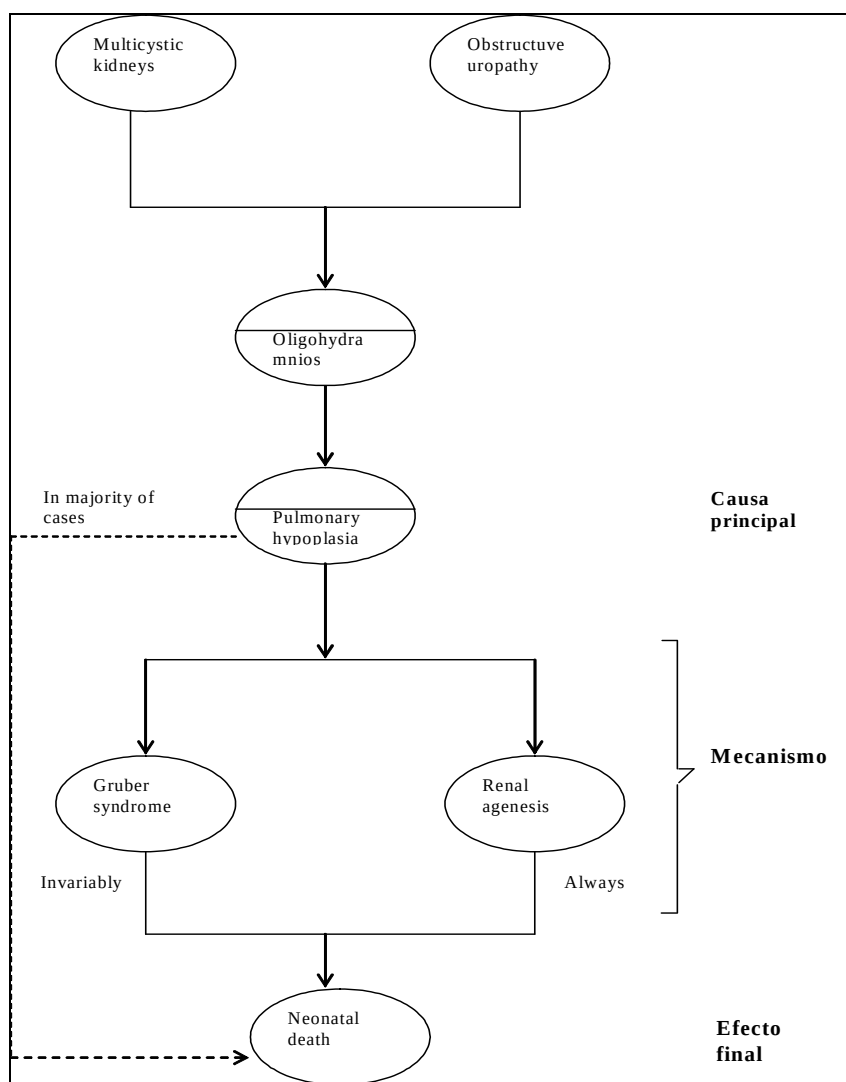


Figura 6: Representación del grafo causal con la salida extraída en la figura 5.

Agradecimientos

Proyecto TIN2007-67494 F-META, MEC-FEDER, España, y proyecto PEIC09-0196-3018 SCAIWEB-2, JCCM, España.

Referencias

- [1] C. Puente, J. A. Olivas, Analysis, detection and classification of certain conditional sentences in text documents. Proceedings of the 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU'08, Torremolinos, Spain, (2008).
- [2] D. Paladini, P. Volpe, Ultrasound of congenital fetal anomalies, differential diagnosis and prognostic indicators. Informa healthcare (2007).
- [3] G. Pilu, K. Nicolaides, Diagnosis of fetal abnormalities, the 12-23 weeks scan. Isoug & The Fetal Medicine Foundation, (2002).
- [4] H. Schmid, Probabilistic Part-of-Speech Tagging Using Decision Trees. Proceedings of International Conference on New Methods in Language Processing, Sep. (1994)
- [5] K. Nikolaides, The 11-13 weeks scan. The Fetal Medicine Foundation, (2002).
- [6] K. Nikolaides, G. Rizzo, Doppler in obstetrics. The Fetal Medicine Foundation, (2002).
- [7] M. Bunge, Causality and modern sciences, Dover, (1979).
- [8] R. Barbieri, S. Berga, Precipis: an update in Obstetrics & Gynecology. American College of Obstetricians and Gynecologists, (2002).

FOG: ARQUITECTURA FLEXIBLE PARA LA GENERACIÓN AUTOMÁTICA DE ONTOLOGÍAS

Emilio F. Viñas¹, Jesús Serrano-Guerrero², José A. Olivas², Javier de la Mata², Andrés Soto³,

Departamento de Tecnologías y Sistemas de Información, Universidad de Castilla-La Mancha,
Paseo de la Universidad 4, 13071, Ciudad Real

¹ fdez.emilio@gmail.com

² jesus.serrano, joseangel.olivas, javier.mata@{uclm.es}

³ soto_andres@yahoo.com

Resumen

Tras realizar una consulta a un sistema de recuperación de información (SRI), se obtienen datos de los cuales, sólo algunos tienen relación con la consulta y otros datos relacionados contextualmente con la consulta no son recuperados. Una de las causas de este problema es que los sistemas de recuperación de información actuales no conocen qué relaciones guardan unas palabras con otras y, por tanto, no son capaces de analizar el contexto del dominio en el que se formuló la consulta.

Una posible solución al problema planteado anteriormente es la creación de ontologías que establezcan relaciones entre palabras creando conceptos capaces de definir el dominio. En este artículo se presenta la arquitectura de una plataforma llamada Fuzzy Ontology Generator (FOG). Esta plataforma establece las relaciones existentes entre palabras generando conceptos que definan un dominio de forma automática.

Palabras Clave: Ontología, Lógica borrosa, Sistema de Recuperación de Información.

1 INTRODUCCIÓN

Un sistema de recuperación de información, se basa en la representación, almacenamiento, organización y acceso a la información. Un SRI almacena información mediante la indexación de datos y cuando el usuario del SRI expresa una necesidad, el sistema establece un ranking de elementos en base a dicha consulta.

Formalmente, un SRI [1] se puede describir como una cuádrupla $SRI = [D, Q, F, S]$ en la cual:

- D : es el conjunto de documentos que representa a los elementos almacenados.

- Q : es el conjunto de consultas que representan la consulta del usuario al sistema.
- F : es la herramienta que modela la representación de documentos, las consultas del usuario y sus relaciones.
- $S: Q \times D \rightarrow \mathcal{R}$ es una función que expresa como de relevante es un documento respecto a la consulta de un usuario.

Más importante que la cantidad de datos que se puedan almacenar en un sistema de recuperación de información, es la obtención de datos útiles tras una consulta a dicho sistema. Existen sistemas de recuperación de información basados en conocimiento, que aumentan la eficacia de la recuperación de información y dan más flexibilidad a la consulta.

La incorporación de conocimiento a los sistemas de recuperación de información provoca un salto cualitativo y cuantitativo en la búsqueda de información. La extracción de conocimiento y su posterior incorporación a un sistema de recuperación de información, se basa en el trabajo de grupos de expertos que extraen conceptos y establecen relaciones entre ellos. En el caso de que el dominio a modelar sea amplio, se requiere de un gran esfuerzo que convierte la obtención de las reglas que forman el conocimiento, en una tarea imposible de abordar. Para poder añadir a los sistemas de recuperación de información la capacidad de analizar el contexto de una consulta, bastaría con la incorporación de un conocimiento que describiese las relaciones existentes entre los términos que forman el dominio.

En este artículo se presenta FOG, una herramienta capaz de crear ontologías de forma automática. La idea es desarrollar una herramienta que permita la posibilidad de albergar distintas estrategias para la generación de ontologías aplicando distintos algoritmos a la hora de establecer las relaciones y atribuirle un grado de similitud semántica a dichas relaciones mediante técnicas de lógica borrosa. Todas estas variantes en el momento de decidir la estrategia a seguir proporcionan a la herramienta una gran flexibilidad.

1. 1. ESTADO DEL ARTE

Son muchos los trabajos propuestos con el objetivo de poder modelar la información y que ésta pueda ser utilizada por máquinas. Este es el caso de Lucarella et al. [6], que en 1991, presentó un ejemplo de un sistema borroso de recuperación de información basado en conocimiento, en el que se aplicaba la teoría de conjuntos borrosos de Zadeh [10]. En este sistema, el conocimiento se estructuraba mediante una red de conceptos. Pero dicha red de conceptos fue establecida por expertos, haciendo la creación de este sistema inviable en el caso de dominios amplios. Esta problemática contribuyó a la idea de extraer el conocimiento de una forma automática.

En 1999, Liang [5] presentó un trabajo en el cual se creaba una red de conceptos de tres niveles de forma automática. En esta red se representan los documentos en el nivel inferior de la red conceptual y el resto de los niveles representan a los conceptos.

Tanto Lucarella como Liang, entre otros autores, formulaban que las relaciones entre nodos fuesen únicas y fijas. En 2003, Horng et al. [4], propuso que las relaciones entre conceptos dependen de la perspectiva, es decir, según el contexto en el que se formule una pregunta, se puede elegir una relación u otra entre un par de nodos. Horng, establece cuatro tipos de relaciones entre nodos: relación borrosa positiva, relación borrosa negativa, relación borrosa generalización y relación borrosa de especificación. A cada una de estas relaciones se les atribuye un valor, proporcionando así semántica a la unión entre nodos.

Como resultado de la extracción automática de conocimiento, se propusieron herramientas como FOGA [9], que genera ontologías de forma automática haciendo uso de lógica borrosa. Gracias a las ontologías se puede representar la semántica entre las palabras de una forma entendible por las máquinas. Según Gruber [3], una ontología es una especificación explícita de una conceptualización. Para nuestra arquitectura, se puede entender una ontología como una cuádrupla $O = [C, P, R, A]$ donde:

- C: es el conjunto de conceptos definidos en el dominio. Un concepto es a menudo considerado como una clase en una ontología.
- P: es el conjunto de propiedades de un concepto. Una $p \in P$ es definida como un instancia de una relación ternaria de la forma $p(c, v, f)$, donde $c \in C$ es un concepto de una ontología, v es una propiedad asociada con c y f define restricciones sobre v .
- R: es un conjunto de relaciones semánticas entre dos conceptos en C.

- A: es el conjunto de axiomas que definen la ontología.

Existen multitud de técnicas para el modelado de información. Por ejemplo, Gomide et al. [7], expone dos teorías. Una basada en la coocurrencia de términos, llamada *Keyword Connection* y otra basada en las relaciones múltiples, *Multirelationship Fuzzy Concepts*. Estas dos teorías, entre otras, ponen de manifiesto la multitud de estrategias para el modelado de información usando lógica borrosa.

1. 2. MOTIVACIÓN

La finalidad de FOG es la de crear ontologías que puedan ser utilizadas por un sistema de recuperación de información. La idea de este proyecto surge como elemento complementario para el framework BUDI [8], desarrollado en colaboración entre la empresa CSD¹ (Computer for Software and Development), con el fin de cubrir algunas de sus deficiencias como es la gestión de información externa útil para los distintos procesos que requieren de cierta información con carga semántica.

En los siguientes apartados de este documento se detalla más detenidamente el funcionamiento completo de la arquitectura de FOG. Más concretamente, en el apartado 2 se hará un repaso de la arquitectura completa de la herramienta y se comentará la parte de configuración de la que se hará uso en el flujo de ejecución de la aplicación, a continuación, en el apartado 3, se analizará cómo se realiza la extracción de datos, cuyo resultado pasará a la fase de modelado de la información de la que se hablará en el apartado 4 y, por último, en lo referente al funcionamiento de FOG, en el apartado 5, se explicarán las distintas representaciones que pueden hacerse sobre el grafo conceptual generado. Finalmente, algunas conclusiones y trabajos futuros serán comentados brevemente.

2 ARQUITECTURA

Se puede definir FOG (Fuzzy Ontology Generator) como una herramienta abierta a la incorporación de nuevos algoritmos con los que se puedan diseñar distintas estrategias para la creación de ontologías de forma automática.

El objetivo es poder realizar variaciones en las funcionalidades e incluso intercambiarlas para poder obtener distintos resultados representados visualmente mediante un grafo o en un lenguaje para la representación

¹ <http://www.csd-systems.com/>

de ontologías (RDF², OWL³, Fuzzy OWL [2]). Así queda patente que es una herramienta en continuo desarrollo al ser posible realizar mejoras de rendimiento, aumentar la funcionalidad de componentes e incorporar nuevas estrategias. FOG debe ser totalmente independiente de cualquier librería externa que pueda ser susceptible de ser utilizada (indexadores, procesadores léxicos o sintácticos, visualizadores...), de tal forma que cualquier componente pueda ser intercambiado sin problemas.

Antes de comentar los principales módulos que forman FOG, es necesario comentar un componente ajeno al flujo de datos. Este componente es el módulo de configuración (CONFIG, mostrado en la figura 1), en el cual el usuario de la herramienta establece los parámetros que determinan la estrategia a seguir para la creación de la ontología. El módulo de configuración lee los datos de un archivo XML, de forma que mediante reflexión se procede a la carga de los mismos definiendo así el flujo de ejecución del sistema.

A lo largo de un proceso intervienen distintos módulos que pueden actuar en tres fases principales:

- *Extracción de datos:* En esta fase se procede a la extracción de los datos desde la fuente indicada en el módulo de configuración. La extracción de datos realiza la indexación de todos los términos que contiene la colección a modelar. Esta colección puede ser una colección de documentos, una base de datos, etc. A lo largo de esta fase intervienen dos componentes principales: el componente básico de indexación de datos y el módulo de preprocesamiento de datos.
- *Modelado de información:* En esta fase se hace uso del concepto entidad. Se entiende por entidad, a la unidad base en la estructura final de FOG. Una entidad se puede definir como un elemento de la colección de documentos a procesar junto con las propiedades que posee en el contexto en el cual es analizada.

La elección de que es una entidad, es introducida en el módulo de configuración por el usuario. Más formalmente una entidad queda definida como una dupla $E = [V, P]$ donde: V es el elemento elegido en el módulo de configuración como unidad base y P es el conjunto de propiedades necesarias para el modelado.

Al cargar la información introducida por el usuario en el módulo de configuración, se

detectan las entidades, las relaciones existentes entre ellas y se ejecutan las funcionalidades que caracterizan el modelado final del grafo. Tales funcionalidades pueden ser: la fusión de entidades, la poda de entidades que no tengan relevancia, etc., dando como resultado un grafo que será la representación del dominio sobre el que se trabaja.

- *Representación de la ontología:* Es la fase final a la que llega el flujo de ejecución de FOG. En este módulo pueden ejecutarse distintos visualizadores y se da un formato concreto a la ontología resultante para poder ser utilizada por un sistema de recuperación de información.

En la figura 1 se indica la arquitectura general de FOG que a continuación será comentada.

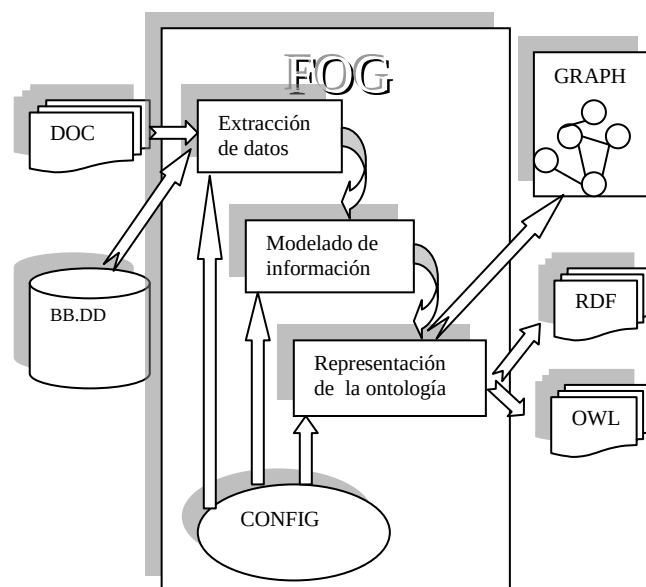


Figura 1: Arquitectura de FOG.

3 EXTRACCIÓN DE DATOS

El módulo de extracción de datos se encarga de extraer los términos contenidos en los documentos de la colección y crear un índice con ellos. Este índice es procesado para poder obtener todos los datos que se encuentran en un documento desde el punto de vista léxico y así, posteriormente, poder darles un valor semántico.

3.1. INDEXACIÓN DE DATOS

Dentro de la fase de extracción de datos, un primer proceso es crear un índice en memoria local con el

² RDF <http://www.w3.org/RDF/>

³ <http://www.w3.org/2004/OWL/>

contenido de la colección, de forma que luego pueda ser utilizado. Con este proceso se consigue independizar la fuente documental de los datos útiles para el proceso de FOG.

3.2. PREPROCESO DE DATOS

El análisis léxico y sintáctico de los datos que contiene el documento se realiza en esta fase. Los datos que se pueden obtener, y que nos interesan para la estrategia de modelado, se obtienen de la configuración introducida por el usuario. También el usuario introduce en el módulo de configuración la herramienta de procesamiento de lenguaje natural, como puede ser GATE⁴ (General Architecture for Text Engineering) entre otras, que permitan operaciones básicas para el procesamiento de lenguaje natural como:

- Tokenizador: analiza un texto en el que identifica palabras, números, símbolos, símbolos de puntuación, espacios en blanco, etc.
- Formateador de oraciones: identifica las oraciones, especificando el inicio y el final de las mismas.
- Etiquetador: identifica para cada palabra el tipo, es decir, indica si es sustantivo, nombre propio, verbo transitivo, adjetivo calificativo, etc.
- Analizador sintáctico: se obtiene un análisis sintáctico del documento a procesar.

4 MODELADO DE LA INFORMACIÓN

En la fase de modelado de la información intervienen distintas funcionalidades necesarias para la generación automática de la ontología. Estas funcionalidades se pueden clasificar entre funcionalidades obligatorias como la funcionalidad de procesado de entidades o detección de relaciones y otras opcionales como la fusión de entidades, la poda de entidades.

FOG es una herramienta que se encuentra en sus inicios, por lo que se puede aumentar el número de funcionalidades opcionales que intervienen en el proceso.

4.1. PROCESADO DE ENTIDADES

El procesado de entidades se basa en un algoritmo capaz de detectar los elementos del documento que inicialmente se consideran entidad. Las características de una entidad

son introducidas por el usuario en el módulo de configuración. Este algoritmo detecta las entidades y extrae del documento toda la información necesaria para la detección de relaciones.

Para la obtención de la información de una entidad se tiene que hacer uso de los datos obtenidos durante el preprocesamiento de datos. Por tanto, alguna de las propiedades que componen una entidad son:

- Valor de la entidad: es el identificador de la entidad, es decir, palabra que forma la entidad.
- Peso de la entidad: este peso se puede definir a nivel de colección. Indicando la relevancia de la entidad en la colección.
- Identificador de oración: identifica a la oración que contiene la entidad.
- Identificador de documento: identifica al documento que contiene la oración en la que se encuentra la entidad.

4.2 DETECCIÓN DE RELACIONES

El siguiente paso en el flujo de ejecución es la detección de las relaciones. En esta fase se relacionan las entidades según la estrategia diseñada por el usuario. En la estrategia diseñada se pueden tener en cuenta entre otros factores, si las relaciones entre entidades se realizan a nivel de oración, a nivel de documento, en función de una distancia entre palabras establecida por el usuario.

Todas las relaciones detectadas tienen un grado de relación semántica. Por ejemplo, si el usuario determina que las relaciones se establecen a nivel de oración y luego a nivel de documento, para todos los documentos de una colección se crearán arcos entre todas las entidades que formen parte de la misma oración y luego se crearán arcos entre todas las oraciones del documento para terminar uniendo los documentos y estructurar jerárquicamente toda la colección.

El siguiente paso es el cálculo del grado de relación semántica que existe en las conexiones detectadas para cada par de entidades. Para realizar dicho cálculo, una de las técnicas propuestas es la lógica borrosa tal y como ya proponen otros autores comentados en el estado del arte.

Por ejemplo, Pereira [7] menciona el cálculo del grado de relación semántica entre términos en base a la coocurrencia de términos:

$$W_{ij} = \frac{N_{ij}}{N_i + N_j - N_{ij}}$$

⁴ <http://gate.ac.uk/>

Donde N_i (N_j) es el número de documentos que contienen a los términos t_i (t_j) y N_{ij} es el número de documentos que contienen el término t_i y el término t_j .

Esta técnica para establecer el grado de las relaciones, otras propuestas en el estado del arte y nuevas técnicas que se realizarán en trabajos futuros, tienen cabida en FOG. El usuario decide que técnica aplicar introduciendo las variaciones en el módulo de configuración.

4.3 FUSIÓN DE ENTIDADES

Es normal que en un documento existan palabras que al compartir una estrecha relación semántica, pertenezcan a un mismo subconjunto del dominio que estamos tratando. Por tanto, en toda una colección documental perteneciente al mismo dominio, el número de elementos que componen este subconjunto aumentan. Es por este hecho que el grafo resultante tras el proceso de detección de relaciones puede llegar a ser ingobernable. De aquí surgió la idea de la fusión. Esta funcionalidad se basa en algoritmos que detectan las entidades que tienen un determinado grado de relación semántica y las agrupa formando una única entidad de mayor nivel de abstracción. Al finalizar este proceso se incorporan al grafo conceptual nuevas entidades catalogadas como conceptos. Se puede definir concepto como el conjunto de entidades que comparten un cierto grado de similitud semántico.

4.4 PODA DE ENTIDADES

Cuando la colección documental alcanza un tamaño considerable y toda la colección pertenece a un mismo dominio, el grado de relevancia a nivel de colección de algunas entidades, en comparación con el grado de relevancia obtenido por otras entidades, se puede encontrar fuera del umbral establecido por el usuario. Este hecho indica que esta entidad es poco significativa para el dominio del que se trata y se descarta para la concepción final del grafo conceptual.

5 REPRESENTACIÓN DE LA ONTOLOGÍA

Tras la conclusión de los procesos anteriores se tiene como resultado una ontología. Pero su utilización en otras aplicaciones depende de la representación final. Para sistemas de recuperación de información como BUDI [8], la representación de un grafo conceptual de un dominio facilitaría la desambiguación de términos especificados en una consulta y así poder precisar más la respuesta del sistema a una consulta de un usuario basándose en el contexto.

FOG permite la visualización de la ontología en distintos formatos y cada uno pone de manifiesto una serie de características a destacar. Como ejemplo de visualización se presentan las siguientes figuras. Estas muestran las relaciones entre las entidades detectadas en los siguientes documentos que son oraciones de la colección MEDLARS⁵:

- Doc1: *correlation between maternal and fetal plasma levels of glucose and free fatty acids.*
- Doc2: *free fatty acid concentration in maternal plasma and fetal body fat content.*
- Doc3: *the concentration of non-esterified fatty acids in maternal and fetal plasma in intact, alloxan-diabetic and x-ray-irradiated rats.*
- Doc4: *7 patients who had biopsy of the bone marrow, all had abnormal types or numbers of plasma cells.*

En la figura 2, se pueden apreciar la expansión de conceptos. El usuario puede seleccionar el nivel de abstracción que desea visualizar.

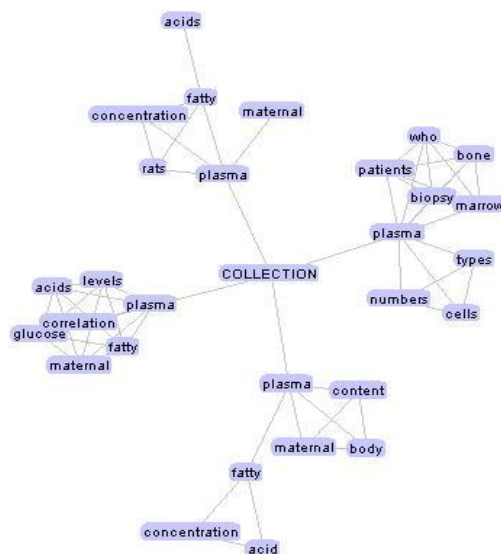


Figura 2: Representación visual de la ontología en función de nivel jerárquico.

En la figura anterior se puede ver la relación existente entre la entidad *acids* y la entidad *fatty*, y por consiguiente, existe un grado de relación semántica establecido por la fórmula aplicada en la etapa de detección de relaciones. Un ejemplo de cálculo del grado de relación semántica puede ser la utilización de la fórmula comentada en el apartado 4.2.

⁵ <ftp://ftp.cs.cornell.edu/pub/smart/>

En la figura 3, se representa la proximidad entre entidades en función de una entidad elegida como entidad central. En el caso del ejemplo, la entidad central es COLLECTION.

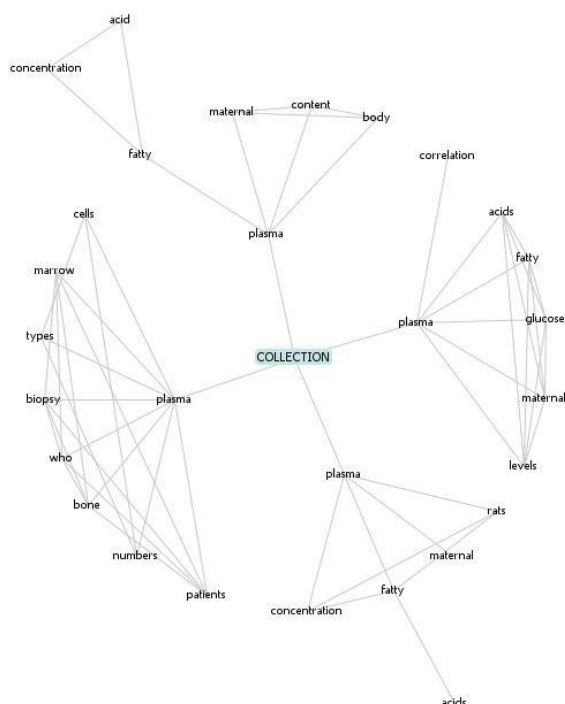


Figura 3: Representación visual de la ontología en función de una entidad central.

Para que un sistema de recuperación de información pueda hacer uso del grafo conceptual, éste debe estar representado en un lenguaje que el sistema de recuperación de información pueda manejar. La representación final del grafo conceptual podría ser formateada en un lenguaje como RDF, OWL ó FUZZY-OWL [2]. Se hará uso de una representación u otra en función de lo especificado por el usuario y de las restricciones que se incluyan dentro de la ontología.

6 CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se ha presentado la arquitectura de la plataforma FOG, la cual está orientada al proceso de generación automática de ontologías en colecciones de datos tales como colecciones documentales, bases de datos relacionales, etc.

Actualmente la plataforma está preparada para la incorporación de algoritmos, así pues el siguiente paso es la incorporación de algoritmos que permitan generar los distintos componentes de una ontología así como incorporar algunas de las propuestas de los autores que se

han citado en el estado del arte para poder comparar sus resultados con propuestas propias.

De igual forma otro de los próximos trabajos será la creación de las interfaces necesarias para conectar la plataforma BUDI con FOG, con el fin de aprovechar sus resultados para distintas fases como la expansión de consultas o la desambiguación de significados en las distintas fases del proceso de metabúsqueda.

Agradecimientos

La investigación realizada está financiada por el proyecto TIN2007-67494 del Ministerio de Educación y Ciencia, el proyecto PEIC-09-0196-3018 de la Junta de Comunidades de Castilla-La Mancha y el proyecto TSI-020302-2009-55 financiado por el Ministerio de Industria, Turismo y Comercio.

Referencias

- [1] Baeza-Yates, R.A., Ribeiro-Neto, B., Modern Information Retrieval, Boston, MA, USA, Addison Wesley, 1999.
- [2] Calegari, S., Ciucci, D. (2007). Fuzzy Ontology and Fuzzy-OWL in the KAON Project. Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International. 1-6.
- [3] Gruber, T. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5 (2), 199-220.
- [4] Horng, Y., Chen, S., & Lee, C. (2003). Automatically constructing multi-relationship fuzzy concept networks for document retrieval. Applied Artificial Intelligence, 17(4), 303-328.
- [5] Liang, T., & Chang, C. (1999). Chinese textual retrieval based on fuzzy concept network. Proceedings of National Computer Symposium, 1, 61-67.
- [6] Lucarella, D., & Morara, R. (1991). FIRST: Fuzzy information retrieval system. Journal of Information Science, 17(1), 81-91.
- [7] Pereira, R., Ricarte, I., & Gomide, F. (2009). Information Retrieval with FROM: The Fuzzy Relational Ontological Model. International Journal, 24, 340-356.
- [8] Serrano-Gurrero, J., Romero, F., Olivas, J., de La Mata, J., & Soto, A. (2009). BUDI: Architecture for Fuzzy Search in Documental Repositories. Mathware and Soft Computing, 16, 71-85.
- [9] Tho, Q., Hui, S., Fong, A., & Cao, T. (2006). Automatic fuzzy ontology generation for semantic web. IEEE Transactions on knowledge and data engineering, 18 (6), 842-856.
- [10] Zadeh, L. (1965). Fuzzy sets. Information and Control, 8, 338-353.

Representación Múltiple de Consultas: Una alternativa a la Expansión de Consultas en Sistemas de Recuperación de Información

José Ruiz-Morilla¹ Jesús Serrano-Guerrero² José A. Olivas² Emilio F. Viñas³
Departamento de Tecnologías y Sistemas de Información, Universidad de Castilla-La Mancha,
Paseo de la Universidad 4, 13071, Ciudad Real

¹ joaquin.ruiz@panel.com

² jesus.serrano, joseangel.olivas@uclm.es

³ fdez.emilio@gmail.com

Resumen

A lo largo de los años se ha intentado mejorar la calidad de información obtenida por los sistemas de recuperación de información teniendo en cuenta la interacción con el usuario. Para ello es necesario usar métodos que permitan dirigir el proceso de búsqueda como puede ser la expansión de consultas, con el fin de obtener una búsqueda más precisa y eficiente. Pero la puesta en funcionamiento de estas técnicas no siempre es posible, por lo que hay que optar por medios alternativos.

En este artículo se presenta una metodología para la representación de una consulta desde distintos puntos de vista aplicable especialmente en aquellos ambientes donde otras técnicas clásicas no pueden ser aplicables.

Palabras Clave: Representación múltiple de información, reformulación de consultas, lógica borrosa.

1. INTRODUCCIÓN

El proceso de búsqueda en Internet se ha transformado en un trabajo muy costoso especialmente desde el punto de vista temporal y computacional debido a la gran cantidad de datos que hay actualmente en la red. Esto es visible sobre todo cuando un usuario realiza una búsqueda sobre un sistema y se comprueba que la mayor parte de los resultados obtenidos tienen poca o ninguna relevancia con respecto a la consulta del usuario.

Para recuperar información existen diferentes sistemas que llevan a cabo procesos de representación,

almacenamiento, organización y acceso a elementos de información [3]. Estos sistemas con el fin de optimizar el proceso de búsqueda recurren a estrategias como la expansión/reformulación de consultas [1, 5, 14, 15] que pueden resultar interesantes.

Sin embargo estas técnicas no siempre son aplicables como ocurre en el caso del metabuscador BUDI [7] donde la expansión de consultas, entendida principalmente como la adición de términos a una consulta, no puede ser implementada dado que la mayoría de las librerías que proporcionan los buscadores web (Google, Yahoo, etc.) configurables en el sistema tienen restringido el número de términos de cada consulta a una cifra muy baja, 10 términos por ejemplo en el caso de Google.

Así pues es necesario plantear una alternativa a la expansión de consultas con el fin de poder guiar el proceso de búsqueda. Este es el objetivo principal del trabajo aquí presente, presentar una alternativa para aquellos casos en los que no es posible expandir una consulta añadiendo un número considerable de términos.

Para ello tomaremos como base la representación múltiple de información, en este caso las consultas, que implica la representación de la misma información desde distintos puntos de vista. Esta forma de representación de la información permite abrir el proceso de búsqueda para mejorar los resultados de los sistemas de recuperación de información.

Según Ingwersen [12] el principio de representación múltiple o polirepresentación se basa en la hipótesis de que contra más interpretaciones se tenga de un mismo conocimiento mayor es la probabilidad de que ese conocimiento pueda ser encontrado como relevante cuando sea requerido. Belkin [11] sugiere que cada representación de la información debe representar una diferente interpretación del problema por parte del usuario. En recuperación de información pueden encontrarse algunos trabajos relacionados con la

representación múltiple de conocimientos como los de Kekalainen [8, 9], quien propuso la representación de una misma consulta cambiando los operadores de búsqueda, o los trabajos de Skov [10], aplicados en este caso para la representación de documentos.

Así pues el objetivo de este trabajo es presentar una metodología que permita generar un conjunto de consultas desde distintos puntos de vista de forma automática con respecto a una consulta introducida por un usuario. A partir de esta consulta inicial se fijará una estrategia que generará de forma automática una serie de consultas alternativas mediante el intercambio de términos procedentes de la consulta original y términos procedentes de fuentes de información externas que guarden relación con el dominio de aplicación de la consulta. Un ejemplo de estas posibles fuentes puede ser una ontología temática o un diccionario electrónico que pueda proporcionar términos relacionados con los de la consulta original: sinónimos, hipónimos, etc.

Una vez generadas las consultas alternativas con respecto a la original, se llevará a cabo un proceso de comparación entre las consultas generadas automáticamente y la original con el fin de filtrar aquellas consultas que menos parecido guarden con la original. El resultado final del proceso será un conjunto de consultas que representan a la original pero utilizando diferentes términos, este hecho provoca que nuevos documentos puedan ser recuperados por un sistemas de recuperación de información. Las consultas generadas pueden lanzarse en paralelo en sistemas como BUDI [7] y sus resultados finalmente mezclados generando la lista final de resultados esperada por el usuario.

A continuación se describe la metodología aplicada en la ejecución de las consultas, detallando cada uno de los pasos. En la sección 3 se detallan los resultados de los experimentos realizados para demostrar la utilizada de la metodología previamente descrita. Se finalizará comentando los posibles trabajos futuros y conclusiones acerca del trabajo realizado.

2. METODOLOGÍA

La entrada del sistema es una consulta booleana expresada por el usuario en la cual él mismo podrá asignar a cada componente (términos, subexpresiones y la consulta entera) de la misma un valor semántico asociado que guardará relación con su correspondiente componente reformulada (términos, subexpresiones o consulta completa) de las nuevas consultas. Así por ejemplo la consulta:

$$q = 0.3 \ t_1 \text{ AND } 0.3 \ t_2$$

podría indicar que el usuario quiere cada término reformulado guarde al menos un relación de 0.3 con el original. Además el usuario podría indicar el mínimo grado de relación R de la consulta final reformulada con respecto a la original, por ejemplo, $R(q, q_{\text{reformulada}}) = 0.55$. En este caso se ha interpretado el componente semántico como un umbral. A lo largo del tiempo distintas interpretaciones se han podido dar de los pesos en una consulta, un resumen de ellas puede encontrarse en [2].

Así partiendo de la consulta inicial del usuario y con el fin de poder generar un conjunto de consultas relacionadas con respecto a una principal y dado que las nuevas consultas no pueden exceder en demasía el número de términos de la original, se proponen los siguientes pasos:

a) **Detección de los elementos fundamentales que conforman la consulta original mediante la aplicación de técnicas de procesamiento de lenguaje natural.**

Este paso consiste en la extracción de los términos y subexpresiones que conforman la consulta original con el fin de tener información acerca de los datos originales para buscar elementos que guarden alta relación con estos y así poder crear nuevas consultas a partir de ellos. Para ello se pueden utilizar técnicas como el stemming con el fin de buscar palabras con la misma raíz o el etiquetado POS (Part-Of-Speech) para detectar la función que realiza cada palabra, si es un verbo, un término puede ser sustituido por otro verbo en lugar de un nombre en el siguiente paso.

b) **Reformulación de consultas.** Una vez han sido preprocesadas las consultas en el paso anterior, el proceso de reformulación se define como un procedimiento en el cual parte o la totalidad de los diferentes términos extraídos de una consulta son reemplazados por otros elementos con significado similar para obtener una nueva consulta con un cierto grado de relación con respecto a la consulta original.

Para llevar a cabo este paso es necesario tener disponible una fuente que proporcionará los datos relacionados con respecto a los originales. Esta fuente puede ser una ontología temática, un diccionario electrónico, etc.

Para llevar a cabo el proceso de reformulación es necesario fijar una estrategia que seguir, por ejemplo, dada la consulta $q = t_1 \text{ AND } t_2$ y un conjunto de términos $\{t_n, t_m\}$ relacionados con t_1 , podría dejarse fijo el valor t_2 aunque se dispusiera de términos relacionados y reemplazar únicamente el término t_1 por sus términos

relacionados, generando así las siguientes dos nuevas consultas:

$$q_n = t_n \text{ AND } t_2$$

$$q_m = t_m \text{ AND } t_2$$

c) **Pesado de los nuevos elementos de la consulta.**

Los nuevos términos que se han reformulado tienen que ser pesados respecto al grado de relación que mantienen con los términos originales. Es decir, es necesario establecer una función que mida el grado de relación existente entre el término original y el término reformulado.

Por ejemplo supondremos que el grado de relación entre t_1 y t_n es 0.6, y entre t_1 y t_m es 0.5, así las consultas reformuladas quedarán como sigue:

$$q_n = 0.6 t_n \text{ AND } 1.0 t_2$$

$$q_m = 0.5 t_m \text{ AND } 1.0 t_2$$

Donde al no reemplazarse el término t_2 su grado de relación consigo mismo es máximo (1.0). Como puede verse las nuevas consultas no aumentan en tamaño como era deseado.

d) **Calcular el grado de relación entre la consulta reformulada y la original.** Si en el paso anterior realizamos la medición de la relación entre los distintos términos básicos, en este caso realizamos una medición del grado de relación entre la consulta original y la consulta completa reformulada gracias a estos pesos asignados a cada término.

Para medir este grado de relación es necesario una evaluación jerárquica de la consulta, de forma análoga a como realiza Yager [13] pero en este caso teniendo en cuenta el grado de relación entre términos y no el grado de satisfacción de cada término con respecto a un documento.

Así la consulta anterior, reemplazando el operador AND por el operador mínimo, los grados de relación R de cada nueva consulta serían:

$$R(t_1, t_n) = \min(0.6, 1.0) = 0.6$$

$$R(t_1, t_m) = \min(0.5, 1.0) = 0.5$$

e) **Filtrar las nuevas consultas dependiendo de su grado de relación.** Aquellas consultas que menor parecido guarden con la inicial deben ser rechazadas.

Para ello es necesario conocer cuál ha sido la semántica asignada a los pesos de la consulta

inicial del usuario. Por ejemplo en este caso se puede utilizar un valor umbral marcado por el usuario. Así por ejemplo si el usuario hubiera fijado cuando formuló su consulta un grado de relación mínimo entre consultas de 0.55, es decir, un peso a la consulta de 0.55, la consulta t_n sería aceptada mientras la consulta t_m sería rechazada para formar parte del conjunto de consultas alternativas a la inicial que debería ser devuelto.

3. RESULTADOS

Para comprobar el funcionamiento de esta metodología, se llevará a cabo un ejemplo. Se recibirá una consulta booleana original donde un usuario habrá asignado un valor borroso a cada término, subconsulta y consulta completa que indicará el grado de importancia para él de cada uno de esos componentes y como consecuencia el umbral mínimo que se exigirá a cada término o expresión reformulada.

La consulta se reformulará mediante la substitución de términos uno a uno en función de los datos provisto por WordNet [4] los cuales reciben un peso asignado de forma automática mediante el software WordNetSimilarity¹ y la medida propuesta por Wu [16]. Por cada sinónimo, hipónimo o hiperónimo encontrado irán generándose todas las posibles combinaciones con los términos originales y los relacionados, siempre reemplazando un término original por uno relacionado.

Para llevar a cabo el proceso de filtrado vamos a asociar una semántica umbral a cada uno de los pesos asignados por el usuario para cada término, expresión o consulta completa, por esa razón vamos a llamar a este proceso método “*Umbral*”. Los principales puntos que pueden resumir su funcionamiento son:

- Cada uno de los términos de la consulta reformulada tiene que superar un determinado umbral respecto a la consulta original.
- Cada una de las subexpresiones que conforman la consulta tienen asignado también un peso que funcionará como el umbral que deben superar la subexpresión reformulada correspondiente. En caso de que el umbral no sea superado por la subexpresión reformulada se termina la evaluación y se rechaza la consulta como posible consulta alternativa a la original, en caso contrario la ejecución prosigue hasta alcanzar la raíz del árbol, es decir, hasta evaluar completamente la consulta.

¹ <http://wn-similarity.sourceforge.net/>

Así tomaremos como ejemplo la consulta “table and computer” valorada por un usuario (fig. 1), y la consulta reformulada a través de WordNetSimilarity mostrada en la fig. 2.

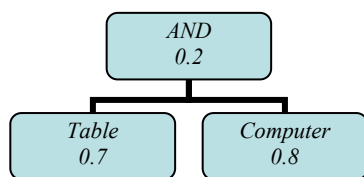


Figura 1: Estructura en árbol de la consulta original

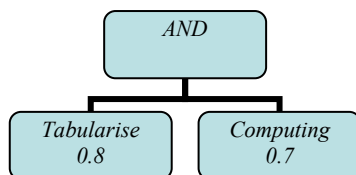


Figura 2: Estructura en árbol de la consulta reformulada

Primeramente comprobaremos si los umbrales de la consulta reformulada son mayores que los de la consulta original, en este caso el grado semántico (0.8) que determina la importancia del término “*Tabularise*” dentro de la consulta es mayor que el grado semántico (0.7) que determina la importancia del término “*Table*” de la consulta original, por lo tanto el proceso continúa con el siguiente término. Analizando el término reformulado “*Computing*” podemos observar que su grado de relación con el original (0.7) es menor que el exigido por el usuario (0.8) por lo que se rechaza la consulta, es decir, no sería seleccionada como consulta alternativa a la original dado que no se ajusta a las exigencias mínimas exigidas por el usuario.

A continuación pondremos un ejemplo de funcionamiento real donde a partir de una consulta dada, se generarán una serie de consultas alternativas bajo las condiciones comentadas al principio de esta sección. Una vez filtradas las consultas que menos relación guarden con la original, el resto de consultas serán lanzadas de forma paralela usando el framework BUDI [7] trabajando únicamente con el buscador Yahoo. Una vez se hayan recibido los resultados de todas las consultas se llevará a cabo un proceso de fusión mediante el método de Borda [6] para generar la lista de resultados final a devolver al usuario.

Partimos de una consulta donde a cada término un usuario le ha asignado un peso borroso cuya semántica asociada es una semántica umbral. La consulta completa se subdivide en subconsultas y a su vez a estas subconsultas se le asigna también un determinado peso borroso. Así, la consulta a evaluar sería la siguiente:

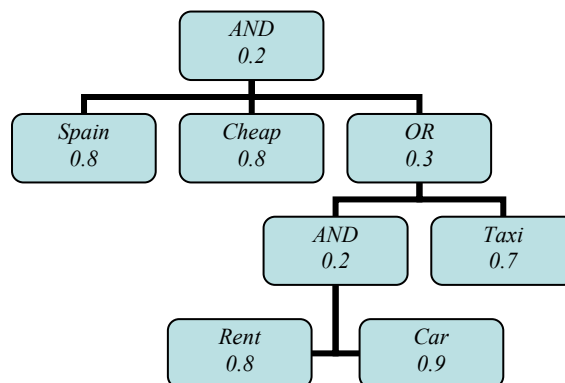


Figura 3: Estructura en árbol de la consulta

Mediante esta consulta se intenta buscar coches alquilables o taxis baratos en España. Cada uno de los términos tiene un grado semántico que indica la importancia de cada elemento con respecto a los otros y el umbral mínimo que se exigirá en el proceso de reformulación de la consulta. Trabajaremos en tres escenarios distintos: el primero ejecutará la consulta sin más sobre un único buscador (Yahoo). En el segundo caso aplicaremos la metodología para generar un conjunto de consultas alternativas pero no aplicaremos ningún método de filtrado de los presentados anteriormente. En el tercer caso haremos lo mismo que en el caso anterior pero aplicando el método de filtrado “*Umbral*”, es decir, asumiremos los pesos del usuario como valores límite que deben cumplir los grados de relación de cada elemento reformulado.

Los resultados del primer caso son mostrados en la tabla 1. Como puede observarse los resultados obtenidos son muy generales, casi todos ellos son páginas de tiendas de alquiler de coches y no tienen en cuenta otros posibles significados de la consulta.

De igual manera los resultados del segundo caso pueden observarse en la tabla 2. En este caso al realizar la representación múltiple de consultas, los resultados obtenidos son más variados, es decir, tiene en cuenta otros posibles significados de la consulta. Así aparecen nuevas páginas relacionadas con temas como blogs o noticias sobre viajes o sistemas de navegación útiles para realizar viajes. El problema de este escenario son la multitud de consultas obtenidas y por lo tanto también el alto coste en términos de tiempo para mezclar los resultados proporcionados por cada consulta. Por esta razón en el último caso se aplica el método “*Umbral*” de filtrado a las consultas con el fin de obtener solamente los resultados de aquellas consultas que guardan una mayor similitud con respecto a la consulta original.

En la tabla 3 podemos ver los resultados de dicho tercer caso una vez han sido filtradas las consultas con menor grado de relación.

En este caso se muestra una dispersión mucho menor que en el segundo caso, es decir, en este caso se eliminan webs como “*John Dulaney-Ezine Articles.com Expert Author*”, “*Forthcoming Archive*”, “*Sept/Oct Diplomat; Lemay, den Ouden, Grimes Consultants For Sale \$749,500 / For Rent \$3,750 p/m ...*”, cuya relación con la consulta original es más que dudosa.

En los tres casos los resultados en las primeras posiciones son muy similares dado que sólo se está usando un buscador (Yahoo), pero a partir de las 20 ó 30 primeras posiciones los resultados son bastante diferentes por los efectos de la fusión de las consultas. En el tercer escenario se ha observado que resultados a partir de dichas posiciones están más relacionados contra mayor sea el umbral exigido, es decir, contra mayor grado de relación guarden las nuevas consultas con la original.

4. CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se ha presentado una metodología para representar consultas desde distintos puntos de vista como alternativa a técnicas como la expansión de consultas. La metodología ha sido aplicada utilizando como interpretación de los pesos de la consulta del usuario una semántica umbral que determina el límite mínimo aceptable de parecido entre cada término o subexpresión de la consulta reformulada y la original.

Los resultados obtenidos no son concluyentes dada la escasa experimentación realizada hasta el momento, por lo que en el futuro se realizarán más experimentos con el fin de concretar el efecto de los distintos filtros, además de evaluar la bondad del sistema mediante colecciones estándares de datos que aporten una mayor objetividad a este proceso.

Agradecimientos

La investigación realizada está financiada por el proyecto TIN2007-67494 del Ministerio de Educación y Ciencia, el proyecto PEIC-09-0196-3018 de la Junta de Comunidades de Castilla-La Mancha y el proyecto TSI-020302-2009-55 financiado por el Ministerio de Industria, Turismo y Comercio.

Referencias

[1] A. H. Alsaar, J. S. Deogun, V. Raghavan, H. Sever. Enhancing concept-based retrieval based on minimal term sets. *Journal of Intelligent Information Systems*, 14, pág. 155-173, 2000.

[2] E. Herrera-Viedma. Modeling the retrieval process for an information retrieval system using an ordinal fuzzy linguistic approach. *Journal of the American Society for*

Information Science and Technology, 52, pág. 460-475, 2001.

[3] G. Salton, M. J. McGill: *Introduction to Modern Information Retrieval*, McGrawHill, Book Company, New York, 1983.

[4] G. A. Miller. "WordNet: a lexical database for English." *Communications of the ACM* 38, pág. 39-41, 1995.

[5] J. C. French, D. E. Brown, N. Kim. A classification approach to boolean query reformulation. *Journal of American Society Information Science*, 48, pág. 694-706, 1997.

[6] J. C. De Borda. *Memorie sur les Elections au Scrutin*. Paris, France: Histoire de l'Academie Royale des Sciences; 1781.

[7] J. Serrano-Guerrero, Francisco P. Romero, Jose A. Olivas, Javier de la Mata. "BUDI: Architecture for Fuzzy Search in Documental Repositories." *Mathware and Soft Computing* 16, pág. 71-85, 2009.

[8] J. Kekalainen, K. Jarvelin. The impact of query structure and query expansion on retrieval performance. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pág. 130-137, New York, NY, USA, 1998.

[9] J. Kekalainen, K. Jarvelin. The co-effects of query structure and expansion on retrieval performance in probabilistic text retrieval. *Information Retrieval*, 1, pág. 329-344, 2000.

[10] M. Skov, B. Larsen, and P. Ingwersen. Inter and intra-document contexts applied in polyrepresentation for best match ir. *Information Processing and Management*, 44, pág. 1673-1683, 2008.

[11] N. J. Belkin, P. Kantor, E. A. Fox, J. A. Shaw. Combining the evidence of multiple query representations for information retrieval. *Information Processing and Management*, 31, pág. 431-448, 1995.

[12] P. Ingwersen, K. Jarvelin. *The turn: integration of information seeking and retrieval in context (The Information Retrieval Series)*. Springer-Verlag, Secaucus, NJ, USA, 2005.

[13] R. Yager. A hierarchical document retrieval language. *Information Retrieval*. 3(4), pág. 357-377, 2000.

[14] S. Gauch, J. B. Smith. An expert system for automatic query reformulation. *Journal of the American Society for Information Science*, 44, pág. 124-136, 1993.

[15] Y. Chang, I. Ounis, M. Kim. Query reformulation using automatically generated query concepts from a document space. *Information Processing and Management*, 42, pág. 453-468, 2006.

[16] Z. Wu. (1994). Verb semantics and lexical selection. In *Proceedings of the 32nd annual meeting of the association for computational linguistics*, pág. 133-138, Las Cruces, Nuevo Mexico, USA.

Tabla 1: Resultados obtenidos en el primer escenario.

TÍTULO	URL
<i>Discount Spain rent a car from 51 GBP per week</i>	http://www.rentalcargroup.co.uk/Spain/index.htm
<i>Discount Valencia Spain rent a car from 60 GBP per week all ...</i>	http://www.rentalcargroup.co.uk/Spain/Valencia_Airport.htm
<i>Car Hire Spain - Cheap Spain car rentals Arguscarhire.com</i>	http://www.arguscarhire.com/car-hire/spain.html
<i>Spain car hire, Spain car rental from 8 EUR Per day all inclusive</i>	http://www.spaincarrental.net
<i>Cheap Car Rental in Spain with Kemwel</i>	http://www.kemwel.com/rental-car/Spain.cfm
<i>Cheap Car Rental Spain - Car Hire España</i>	http://www.simpleautorentals.co.uk/cheapest_car_hire_spain.shtml
<i>Car Rental Spain - Cheap car hire</i>	http://www.placesonline.com/europe/spain/car_rental.asp
<i>Cheap Spain Car Rental - Spain Driving Information from Kemwel</i>	http://www.kemwel.com/rental-car/Spain-driving.cfm
<i>Spain, Malaga, Rental Car, Rent Cars, Cheap Rentals Car</i>	http://www.spainrentacar.net/index.htm
<i>Spain car hire from 70 EUR / 61 GBP / 100 USD per week...</i>	http://spain.rentalcargroup.com/index.htm

Tabla 2: Resultados obtenidos en el segundo escenario.

TÍTULO	URL
<i>Some Damn Good Life Experience (through the eyes of Stephanie)</i>	http://datadriventravels.typepad.com
<i>Car Hire Spain - Cheap Spain car rentals Arguscarhire.com</i>	http://www.arguscarhire.com/car-hire/spain.html
<i>Free spain newspapers Information - spain newspapers Articles</i>	http://travel.ezineseeker.com/spain-newspapers.htm
<i>It is a low rent area, and the Captain kept insisting that no one with my ...</i>	http://www.maravot.com/The_Romance_Anaisfille.html
<i>Travel : BlogSandwich News Blog</i>	http://www.blogsandwich.com/category/travel
<i>John Dulaney - EzineArticles.com Expert Author</i>	http://ezinearticles.com/?expert=John_Dulaney
<i>Forthcoming Archive</i>	http://www.btinternet.com/~bookcase/FC2008.htm
<i>Sept/Oct Diplomat; Lemay, den Ouden, Grimes Consultants For Sale \$749,500 / For Rent \$3,750 p/m ...</i>	http://issui.com/diplomatonline.com/docs/diplomat_sept_08_final_2?mode=embed&documentId=081027180039-e77c28e740a24aacb2be40ff85cd97fe&layout=grey
<i>Free Information Using Our Detailed Navigation System</i>	http://www.liberated-press-releases.com/catagorydetail.php?Cat=342&level=1&title=Travel
<i>Mustang : Deanne Stillman : ISBN 9780618454457</i>	http://www.buy.com/prod/mustang/q/loc/106/206286089.html

Tabla 3: Resultados obtenidos en el tercer escenario.

TÍTULO	URL
<i>Facts about Spain; Cheap Hostels</i>	http://www.spanish-town-guides.com/Facts.htm
<i>Free spain newspapers Information - spain newspapers Articles</i>	http://travel.ezineseeker.com/spain-newspapers.htm
<i>Car Hire Spain - Cheap Spain car rentals Arguscarhire.com</i>	http://www.arguscarhire.com/car-hire/spain.html
<i>Rental Car / Taxi - Palma de Mallorca Transportation ...</i>	http://www.virtualtourist.com/travel/Europe/Spain/Balearic_Islands/Palma_de_Mallorca-258946/Transportation-Palma_de_Mallorca-Rental_Car_Taxi-BR-1.html
<i>Discount Valencia Spain rent a car from 60 GBP per week all ...</i>	http://www.rentalcargroup.co.uk/Spain/Valencia_Airport.htm
<i>AM General to produce Standard Taxi cabs? — Autoblog</i>	http://www.autoblog.com/2008/07/14/am-general-to-produce-standard-taxi-cabs
<i>Cheap Car Rental Spain - Car Hire España</i>	http://www.simpleautorentals.co.uk/cheapest_car_hire_spain.shtml
<i>; Rental Cars - Travel at Your Own Pace</i>	http://ezinearticles.com/?Rental-Cars---Travel-at-Your-Own-Pace&id=2791420
<i>Isla Verde Hotels: Coqui Inn - Isla Verde, Puerto Rico;</i>	http://www.coqui-inn.com/gettinghere.html
<i>Moscow: Facts, Discussion Forum, and Encyclopedia Article</i>	http://www.absoluteastronomy.com/topics/Moscow
<i>.....</i>	<i>.....</i>
<i>Tom Waits True Confessions</i>	http://www.antilabelblog.com/?p=288
<i>It is a low rent area, and the Captain kept insisting that no one with my ...</i>	http://www.maravot.com/The_Romance_Anaisfille.html

Ontologías borrosas para representar perfiles de usuario en una herramienta de gestión de objetos de aprendizaje

Mateus Ferreira-Satler¹, Victor H. Menéndez², Francisco P. Romero³,
Alfredo Zapata², Manuel E. Prieto⁴

¹Escuela Superior de Informática, Universidad de Castilla La Mancha, msatler@gmail.com,

²Universidad Autónoma de Yucatán, Mérida, México, {mdoming, zgonzal}@uady.mx

³E.U. Ing. Tec. Industrial de Toledo, Univ. de Castilla La Mancha, FranciscoP.Romero@uclm.es

⁴Dept. de Tecnologías y Sistemas de Información, Univ. de Castilla La Mancha, ManuelE.Prieto@uclm.es

Resumen

Este trabajo propone una aplicación específica de la utilización de ontologías borrosas con el fin de representar perfiles de usuario, que permitan mejorar el rendimiento en tareas como el filtrado y la recuperación de información. La construcción de estas ontologías se realiza de forma automática utilizando para ello técnicas relacionadas con la lógica borrosa. Con el fin de probar la validez de esta aproximación, se han integrado en una herramienta de gestión de objetos de aprendizaje, en la cual, el perfil de cada uno de los usuarios se construye a partir de los objetos de aprendizaje publicados por él mismo. Según los experimentos realizados, el método de construcción y representación se adecua con gran precisión y exhaustividad a lo requerido por el usuario.

Palabras Clave: Ontologías Borrosas, Perfiles de Usuario, Objetos de aprendizaje.

1 INTRODUCCIÓN

La proliferación de sitios y repositorios dedicados a ofrecer recursos destinados a la educación es una prueba del constante desarrollo del e-learning en la actualidad. Es por ello que numerosas instituciones transforman y publican recursos educativos para que sean aprovechados y reciclados.

A pesar de los esfuerzos de normalización y estandarización de los recursos, un problema latente en toda aplicación e-Learning y especialmente en los repositorios de recursos educativos (LCMS, Learning Content Management System) es poder ofrecer los recursos apropiados a las necesidades de los usuarios [4]: profesores, estudiantes, diseñadores instruccionales, etc. Cada uno de estos usuarios posee metas específicas y está

dentro de un contexto distinto cuando busca alguna información. La personalización de los procesos de filtrado y recuperación de la información contribuirá a atenuar la sobrecarga de información, ajustando los resultados presentados a las necesidades individuales de cada usuario y mejorando la eficiencia en la gestión de los recursos educativos a su cargo.

Uno de los medios que soportan la mejora del proceso de filtrado y recuperación de información son las ontologías. El uso de ontologías ha recibido gran atención en la organización y distribución del conocimiento porque agrega un nivel conceptual en la recuperación de información, en contraste con la recuperación meramente sintáctica. Además, los perfiles de usuario basados en ontologías vienen siendo ampliamente aplicados en la representación del contexto y en la personalización de aplicaciones, de manera que éstas se ajusten a los requisitos del usuario.

En este trabajo se propone la integración de los perfiles de usuario y las ontologías como herramientas de soporte a los procesos de gestión de objetos de aprendizaje dentro de un LCMS, para la mejora tanto del filtrado y recuperación de la información como la clasificación y gestión de estos elementos. La plataforma AGORA [5] pretende apoyar a los profesores en el desarrollo de objetos de aprendizaje acordes a estrategias de diseño educativo y empleando un enfoque de recomendación. Dentro de este contexto, se ha abordado la definición, aplicación y validación de la construcción automática de perfiles de usuario basados en ontologías borrosas.

El resto del artículo está organizado de la siguiente manera. En la siguiente sección se presenta la herramienta AGORA, base de nuestro estudio experimental. La sección 3 está dedicada a la representación del perfil del usuario mediante ontologías borrosas y el proceso de construcción de este perfil. En la sección 4 se presentan los resultados del caso de estudio realizado sobre el uso de AGORA, dedicando la última sección a las conclusiones y al trabajo futuro.

2 AGORA: GESTIÓN DE OBJETOS DE APRENDIZAJE

Un objeto de aprendizaje es una unidad básica de información en e-Learning, que combina uno o más recursos educativos con los metadatos que los describen. Esta estructura permite su intercambio y reutilización entre distintos sistemas e-Learning sin que exista pérdida de su funcionalidad o características.

Los objetos de aprendizaje pueden ser cualquier recurso que el profesor considere relevante para la instrucción de sus alumnos. Como ejemplos de estos recursos se pueden citar los objetos de aprendizaje basados en documentos con información textual, que fueran utilizados como fuente de información para construir el perfil de usuario en este trabajo; y los objetos de aprendizaje basados en multimedia, como videos o audios.

La gestión de objetos de aprendizaje implica un conjunto de actividades requeridas para controlar su ciclo de vida. Su finalidad es el desarrollo de objetos de aprendizaje de calidad que puedan ser reutilizados en distintas soluciones e-Learning. Dicha gestión puede realizarse por medio de un entorno que integre las características de edición, transformación, almacenamiento y colaboración presentes de forma individual en las herramientas existentes para el desarrollo de objetos.

La plataforma AGORA (Ayuda para la Gestión de Objetos Reusables de Aprendizaje) [5] surge como una solución para los procesos de gestión de objetos de aprendizaje realizados de forma integral. Uno de sus objetivos es emplear un enfoque de asistencia y recomendación para la ejecución de procesos como el etiquetado (generación de metadatos), la búsqueda y la secuenciación de objetos de aprendizaje. Para ello, se consideran factores como el diseño instruccional, la gestión del conocimiento, su historial de acciones, etc. Un papel clave para lograr la máxima eficiencia en esta labor de gestión es la definición concreta y correcta del perfil del profesor, punto en el cual se centra este trabajo.

AGORA está formado por cinco módulos, cada uno de ellos especializado en una función concreta: Un módulo de gestión del conocimiento y su representación basada en ontologías instruccionales y de dominio; un módulo de gestión de objetos de aprendizaje para la administración de los recursos y metadatos propios; un módulo de metabúsqueda para buscar objetos de aprendizaje almacenados en otros repositorios, un módulo extracción de conocimiento para el análisis de acciones realizadas por el usuario para la generación de perfiles y modelos de comportamiento. Por último, el sistema cuenta con un módulo de recomendación que utiliza los módulos anteriores para dar respuesta a las necesidades de los usuarios.

Las principales características de AGORA pueden resumirse en:

- Una red de repositorios distribuidos en las instituciones participantes.
- Permite gestionar cualquier tipo de recurso utilizado en un LMS – Learning Management System (presentaciones, documentos, videos).
- Incluye un generador-etiquetador de objetos, un meta-buscador, un mecanismo de evaluación de la calidad y un entorno de presentación de contenidos.
- Emplea asistentes y recomendadores para la realización de las actividades.
- Fomenta la colaboración, al ofrecer mecanismos de comunicación e intercambio de conocimiento entre los usuarios.

Dentro de AGORA, los usuarios pueden crear, publicar, agregar, manipular, descubrir y almacenar objetos de aprendizaje. También pueden realizar una búsqueda sobre los objetos publicados. Para optimizar estas búsquedas un factor clave es el aprovechamiento del conocimiento inherente al entorno y al usuario. Con este propósito, se pretende incorporar el perfil de cada uno de los profesores. Para ello, la construcción automática de este perfil se implementa dentro del módulo de extracción de conocimiento. El módulo utiliza un componente que permite definir el perfil de cada usuario a partir de la base que viene conformada por los objetos de aprendizaje que él haya publicado en el sistema. De esta manera, se puede disponer de una representación del perfil de acuerdo con los intereses de trabajo del propio profesor.

Dentro de este contexto, es importante considerar la incorporación de características semánticas al perfil, mediante técnicas de identificación de preferencias de usuario y organización de la información. Las ontologías y la lógica borrosa proporcionan los medios adecuados para representar la información en formato digital y permiten describir una vista general del dominio de interés tratado.

3 PERFILES DE USUARIO BASADOS EN ONTOLOGÍAS BORROSAS

La propuesta presentada en este trabajo incluye varias etapas de procesamiento de datos para generar automáticamente perfiles de usuario basados en ontologías borrosas, a partir de un conjunto de documentos previamente seleccionados. Para lograr dicho objetivo, el proceso de construcción ha sido dividido en cinco etapas: pre-procesamiento lingüístico, indexación de términos (lo que llamamos pre-ontología), extracción de términos relevantes, construcción de la ontología del usuario y actualización del perfil del usuario. La Figura 1 muestra la estructura del proceso.

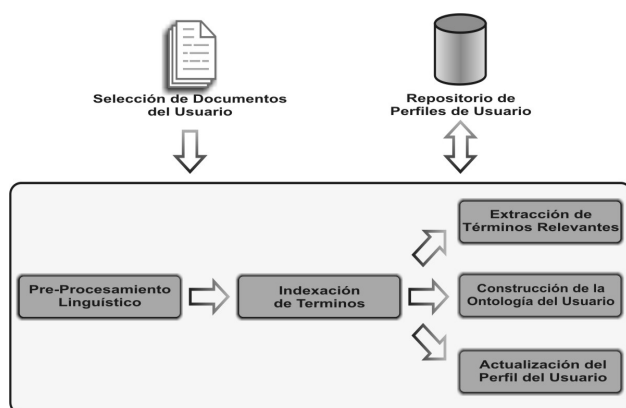


Figura 1: Proceso de Generación Automática de Perfiles de Usuario basados en Ontologías Borrosas.

En las siguientes secciones se describen en detalle cada una de las etapas del proceso.

3.1. PRE-PROCESAMIENTO LINGÜÍSTICO

El principal objetivo de esta etapa es transformar los documentos de los objetos de aprendizaje colocados por un usuario de AGORA en una representación apropiada, para que los mismos puedan ser procesados en las próximas fases.

Primero, se realiza la transformación previa de los documentos para tratar los diferentes tipos de formato (.doc, .txt, .pdf, etc.). Después, se eliminan todas las informaciones no textuales como dígitos, fechas, signos de puntuación, entre otros (análisis léxico) y se identifica el idioma de los textos, a través de una técnica que utiliza diccionarios de palabras frecuentes [1]. También se realiza la corrección ortográfica para sustituir las palabras que no estén escritas correctamente, utilizando diccionarios de corrección automática de ortografía.

El siguiente paso tiene por objeto reducir el vocabulario y generar una representación textual más significativa. Se utilizan dos técnicas conocidas como *Stoplists* [3] y *Stemming* [2]. Estas técnicas, aplicadas de manera adecuada, reducen el tamaño del vocabulario y muchas veces mejoran el desempeño del proceso.

En esta fase, una secuencia de caracteres define un término. Los términos están separados unos de los otros por un delimitador. El proceso prevé la existencia de términos formados por una o más palabras. De cada documento propiedad del usuario se extrae un conjunto de términos. Para que un término sea considerado como válido, el mismo debe aparecer por lo menos en tres documentos. Se utiliza esta estrategia para eliminar las palabras que aportan poca carga semántica al contexto del dominio. Solamente los términos válidos son procesados en las etapas posteriores.

3.2. INDEXACIÓN DE TÉRMINOS

En esta etapa se construye una estructura de índice, llamada *pre-ontología*, que contiene información sobre todos los términos generados en la fase anterior. La *pre-ontología* está formada por un conjunto de objetos del tipo "term". A su vez, cada objeto "term" posee un identificador, que en este caso es el propio término, y una lista de objetos que contienen informaciones relacionadas con los documentos en los cuales el término aparece.

Cada objeto de esta lista de documentos contiene las siguientes informaciones: *Identificador del Documento*; *Número de Ocurrencias del término más frecuente en el documento*; *Lista de posiciones del término en el documento*. La lista de posiciones contiene la posición física del término en el documento. La Figura 2 muestra la estructura de la *pre-ontología*.

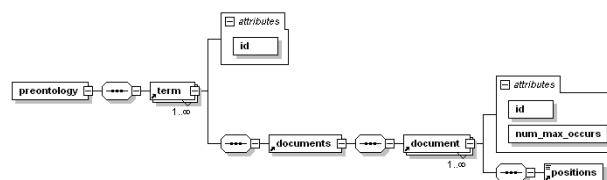


Figura 2: Estructura de la Pre-Ontología.

La *pre-ontología* sirve como base, tanto para la extracción de términos relevantes como para la construcción de la ontología. También sirve como soporte al proceso de actualización de resultados.

3.3. EXTRACCIÓN DE TÉRMINOS RELEVANTES

Con el fin de generar una representación semántica del contexto del usuario, se deben extraer los términos más relevantes dentro de los documentos que ayudan a definir el perfil. Con este fin, es necesario establecer un método para cuantificar el valor de representatividad de un término con relación a un conjunto de documentos.

En este trabajo han sido implementadas dos medidas de relevancia diferentes para calcular el peso de los términos. La primera de ellas es una ampliación de la estrategia propuesta en [6] basada en la medida clásica *tf-idf* con el fin de obtener el peso relativo (pr_i) de cada término dentro de los documentos del perfil:

$$pr_i = \frac{docs(i) + \sum_{j \in D} \frac{ft(i,j)}{|mf_j|}}{|D|} \quad (1)$$

Donde se utiliza el número de documentos en que el término i aparece ($docs(i)$) y su frecuencia de ocurrencia en estos documentos ($ft(i,j)$). La normalización del valor se obtiene utilizando el número de documentos procesados ($|D|$) y el número de ocurrencias del término más frecuente del documento ($|mf_j|$).

La segunda medida de relevancia es una variación de la medida propuesta en [8] llamada KF-IDF. De acuerdo con esta fórmula, el peso de un término crece logarítmicamente inversamente proporcional al número de pre-ontologías de usuarios en las que aparece. En otras palabras, se considera un término como relevante si el mismo ocurre más frecuentemente que otros términos en una determinada pre-ontología de usuario, y ocasionalmente dentro de las restantes. Como se muestra en la fórmula (2), KF-IDF considera la relevancia de un término en base a un grupo o conjunto de usuarios. La nueva medida del peso, llamada *kmod*, se define de la siguiente manera:

$$kmod_i = \sum_{j \in D} tf(i, j) \times \left(1 + \frac{docs(i)}{|D|}\right) \times \ln\left(\frac{|users|}{users(i)} + 1\right) \quad (2)$$

donde:

- $tf(i, j)$ = frecuencia del término i en el documento j .
- $docs(i)$ = número de documentos en que el término i aparece.
- $|D|$ = número total de documentos.
- $|users|$ = número total de usuarios.
- $users(i)$ = número de usuarios cuya pre-ontología tiene indexado el término i .

En este caso, el valor calculado no se ofrece directamente como relevancia, sino que se normaliza dividiéndolo por la media de los valores obtenidos. De este modo, el perfil del usuario contiene los términos con mayor relevancia (del 75% al 100%).

3.4. CONSTRUCCIÓN DE LA ONTOLOGÍA DEL USUARIO

En esta etapa, la construcción automática de la ontología del usuario tiene por objeto generar una clasificación jerárquica de los términos indexados en la pre-ontología. La ontología, en este contexto, puede ser considerada como un conjunto de árboles relacionados donde cada nodo representa un ítem. En estos árboles, el nodo A es descendiente del nodo B si el ítem A es “más concreto que” (*narrower than*) el ítem B , o si el ítem B es “más general” (*broader than*) el ítem A . La propuesta para generar automáticamente ontologías borrosas, se basa en el algoritmo propuesto en [7]. Este algoritmo permite calcular medidas borrosas del grado de generalidad (*GD – Generality Degree*) entre parejas de palabras extraídas de una colección de documentos.

Formalmente, se define $C = (d_1, d_2, \dots, d_n)$ como el conjunto de documentos d_j s, donde cada documento $d = (t_1, t_2, \dots, t_m)$ está formado por un conjunto de términos t_i . Dado un término t_i , la ocurrencia de t_i en un documento d se representa por $ocurr(t_i, d)$ y su valor se obtiene de acuerdo con la siguiente fórmula:

$$ocurr(t_i, d) = \frac{f(t_i, d)}{f_{\max}(d)} \quad (3)$$

donde:

- $f(t_i, d)$ = frecuencia de ocurrencia del término t_i en el documento d .
- $f_{\max}(d)$ = número de ocurrencias del término más frecuente del documento d .

La relación $GD(t_i, t_j)$ expresa que t_i es “más concreto” que t_j . El grado de esta relación está definido por:

$$GD(t_i, t_j) = \frac{\sum_{d \in D} occurr(t_i, d) \otimes occurr(t_j, d)}{\sum_{d \in D} occurr(t_i, d)} \quad (4)$$

donde $\otimes$ se refiere a un operador de conjunción borroso. En este trabajo han sido implementadas seis t-normas conocidas que ejercen esta función. Son las siguientes:

- 1) Mínimo(M_i): $\min\{x, y\}$
- 2) Producto de Hamacher (PH): $\frac{x \cdot y}{x + y - (x \cdot y)}$
- 3) Producto Algebraico (PA): $x \cdot y$
- 4) Producto de Einstein (PE): $\frac{x \cdot y}{2 - [x + y - (x \cdot y)]}$
- 5) Producto Acotado (PC): $\max\{0, x + y - 1\}$
- 6) Producto Drástico (PD): $\begin{cases} x & \text{if } y = 1 \\ y & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases}$

Para construir la ontología de usuario, primero se calcula el grado de generalidad (*GD*) para una pareja de términos distintos de la pre-ontología, es decir, $GD(t_i, t_j)$ y $GD(t_j, t_i)$. A continuación, se elimina el valor más pequeño. El valor mayor se utiliza para generar la relación que se va a incluir en la ontología. El valor más pequeño se descarta para eliminar relaciones redundantes entre los términos. Con esta estrategia de decisión, se elige la instancia positiva del término si la diferencia entre $GD(t_i, t_j)$ y $GD(t_j, t_i)$ es grande, o se elige la relación más fuerte si la diferencia es pequeña. El proceso se repite para cada pareja de términos de la pre-ontología de todos los usuarios. Finalmente, se eliminan de la ontología las relaciones cuyo valor de *GD* es menor que un determinado umbral, dado que se considera información no relevante. El valor del umbral se ha definido empíricamente en 0.6, tras análisis de las relaciones generadas.

La ontología borrosa generada contiene una descripción taxonómica de las relaciones *GD*, y su estructura se muestra en la Figura 3.

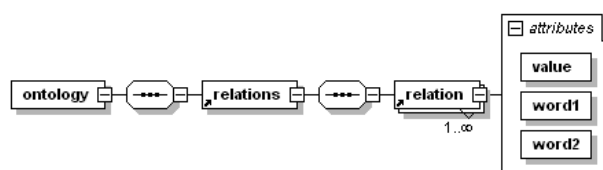


Figura 3: Estructura de la Ontología.

Como se puede observar en la Figura 3, la ontología está compuesta por un conjunto de relaciones. Cada relación tiene dos identificadores (word1 y word2) que representan los términos que se relacionan, y un cuantificador (value) que indica el grado o valor de la relación.

3.5. ACTUALIZACIÓN DEL PERFIL DE USUARIO

Con el objetivo de que el proceso de construcción del perfil de usuario sea más dinámico, es necesario permitir que se pueda incorporar información de nuevos objetos de aprendizaje seleccionados o creados por el usuario directamente en el perfil, si éste ya existe. Dicho procedimiento debe reproducir los mismos resultados que se obtendrían si los objetos de aprendizaje estuvieran disponibles en el inicio del proceso de construcción del perfil.

Para realizar dicha tarea, los nuevos documentos son previamente sometidos a las rutinas de pre-procesamiento lingüístico e indexación, de acuerdo con los procesos descritos en las secciones 2.1 y 2.2. Como resultado, se genera una nueva pre-ontología. El próximo paso consiste en calcular el peso de los términos de la pre-ontología, de acuerdo con las métricas descritas en la sección 2.3 y, posteriormente, actualizar los términos relevantes del usuario. Por último, se actualiza la ontología del usuario siguiendo un procedimiento semejante al de su construcción. Primero se calcula el Grado de Generalidad (*GD*), relacionando cada uno de los términos de la nueva pre-ontología con todos los demás términos, lo que genera como resultado un conjunto de relaciones borrosas. Posteriormente, se usan estas relaciones para actualizar la ontología, de acuerdo con las siguientes reglas: 1) Si la relación generada no existe en la ontología del usuario, entonces se incorpora esta nueva relación. 2) Si la relación ya existe, entonces se actualiza su valor o se excluye la relación, si este es el caso.

4 PERFILES DE USUARIO EN AGORA

En este apartado se presenta el estudio experimental realizado con el fin de representar el perfil de usuarios de la plataforma AGORA, a partir de los documentos proporcionados por los usuarios durante los últimos meses en el servidor situado en la Universidad de Castilla-La Mancha. El sistema cuenta con algo más de 200 usuarios de distintos niveles educativos, instituciones y países.

Únicamente se ha construido el perfil de los usuarios con un número de documentos aportados suficientemente significativo, en concreto más de 5, ya que los propios administradores de AGORA consideran que con una aportación menor, los perfiles generados no representarían correctamente a un usuario concreto. Para este caso se seleccionaron 40 usuarios, quienes han aportado 180 documentos de diversa naturaleza y formato (HTML, PDF, etc.).

En la Tabla 1 se puede observar el conjunto de términos obtenidos como relevantes por parte del algoritmo explicado en el apartado anterior sobre los documentos asociados al usuario 174. Este usuario es el que posee más documentos colocados en el sistema (20 documentos).

Tabla 1: Términos más relevantes para un usuario.

<i>Término</i>	<i>Peso</i>	<i>Término</i>	<i>Peso</i>
aprendizaje	15,42	aplicación	7
dato	10,64	biblioteca	5,68
objeto	9,51	estudiante	5,5
información	9,1	instructor	4,83
objeto aprendizaje	7,22	patrón	4,79
sistema	7,1	alumno	,47

Este usuario ha indicado que los términos seleccionados por el algoritmo incluyen a todos los relevantes a su juicio (100% de exhaustividad) aunque también incluyen algún término demasiado general, por ejemplo, *dato* o *información* (75% en precisión). Los resultados son similares (90% en exhaustividad y 70% en precisión) en cuanto a las relaciones más importantes entre conceptos que se reflejan en la Figura 4. Para el resto de usuarios los resultados han sido similares, con un alto grado de exhaustividad (89,1%) y un grado algo menor de precisión (74,8%).

Por último, en la Tabla 2 se muestran las estadísticas sobre el número de relaciones de las ontologías de los usuarios más significativos generadas según los diferentes operadores de conjunción borrosos. Aquí *Mi* es el Mínimo, *PH* es el Producto de Hamacher, *PA* es el Producto Algebraico, *PE* es el Producto de Einstein, *PC* es el Producto Acotado y *PD* es el Producto Drástico.

Tabla 2: Estadísticas de las Ontologías de los Usuarios.

<i>Usuario</i>	<i>Mi</i>	<i>PH</i>	<i>PA</i>	<i>PE</i>	<i>PC</i>	<i>PD</i>
139	253	187	0	0	0	0
154	946	789	0	0	0	0
158	15952	10539	250	169	72	68
166	9662	3619	321	218	104	92
170	1212	745	26	23	20	20
171	5089	2484	134	124	107	104
172	872	341	81	75	39	38
173	12562	5485	339	278	169	165
174	18688	6658	430	342	221	186

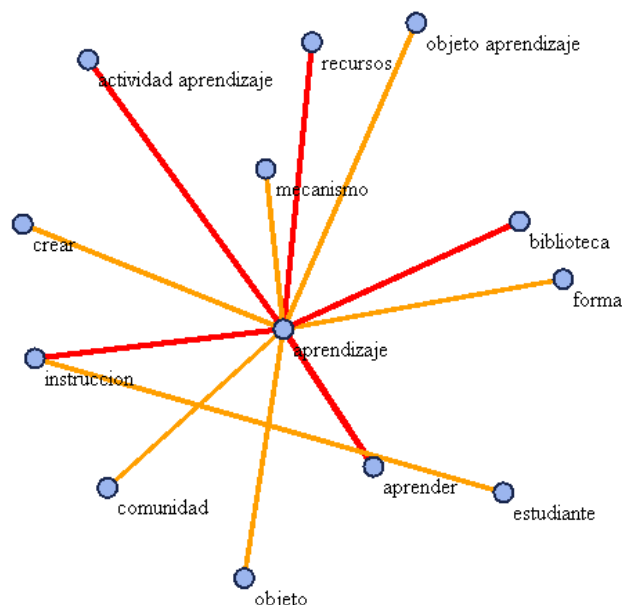


Figura 4: Relaciones más relevantes entre conceptos.

5 CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha propuesto la utilización de ontologías borrosas con el fin de representar perfiles de usuario. Esta aproximación ha sido probada en la herramienta de gestión de objetos de aprendizaje AGORA, cuyos usuarios y expertos han validado que el perfil representa de forma confiable sus temas de interés y preferencias de trabajo.

Como líneas a desarrollar en el futuro se propone la experimentación con un número mayor de usuarios y de documentos, dentro de la misma plataforma. Esto será posible ya que el sistema está siendo adoptado por varias universidades y redes de investigación en Iberoamérica.

Por otro lado, nos planteamos mejorar la calidad del perfil de usuario mediante la realización de procesos de poda que eviten la aparición de términos pocos significativos dentro del perfil, así como incorporar la información proporcionada por el usuario en su descripción.

Agradecimientos

La investigación realizada está financiada por los proyectos TIN2007-67494 del Ministerio de Ciencia e Innovación, PEIC09-0196-3018 de la Junta de Comunidades de Castilla-La Mancha y A/016625/08 de la Agencia Española de Cooperación Internacional para el Desarrollo (AECID), así como por el Consejo Nacional de Ciencia y Tecnología CONACYT, México y el Consejo de Ciencia y Tecnología del Estado de Yucatán CONCyTEY, México.

Referencias

- [1] K.R. Beesley. Language Identifier: A Computer Program for Automatic Natural-Language Identification of On-Line Text. *Languages at Crossroads: Proceedings of the 29th Annual Conference of the American Translators Association*. Pág. 47-54, 1998.
- [2] T. Korenius, et al.. Stemming and lemmatization in the clustering of finnish text documents, in *Proceedings of the thirteenth ACM international conference on Information and knowledge management*. ACM: Washington, D.C., USA. Pág. 625-633, 2004.
- [3] R.R. Korfhage. Information storage and retrieval. *John Wiley & Sons, Inc.* Pág. 349, 1997.
- [4] N. McLean, C. Lynch. Interoperability between Information and Learning Environments: Bringing the Gaps. *A joint White Paper on behalf of the IMS Global Learning Consortium and the Coalition for Networked Information*. Disponible en http://www.imsglobal.org/dlims_white_paper_public/publ_1.pdf. Pág. 13, 2003.
- [5] M.E. Prieto, et al.. A Recommender System Architecture for Instructional Engineering. In *Proceedings of the 1st World Summit on the Knowledge Society: Emerging Technologies and information Systems For the Knowledge Society*. LNIAI, vol. 52 (88). Springer, Heidelberg. Pág. 314-321, 2008.
- [6] F. Romero et al.. Automatic extraction of the main terminology used in empirical software engineering through text mining techniques. In *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*. ACM: Kaiserslautern, Germany. Pág. 357-358, 2008.
- [7] D.H. Widyantoro and J. Yen. Incorporating fuzzy ontology of term relations in a search engine. In *BISC International Workshop on Fuzzy Logic and the Internet*. Berkeley: University of California. Pág. 564-576, 2001.
- [8] F. Xu et al. A Domain Adaptive Approach to Automatic Acquisition of Domain Relevant Terms and their Relations with Bootstrapping. In *Proceedings of LREC 2002, the 3rd International Conference on Language Resources an Evaluation*. Las Palmas, Spain. 2002.

Una aproximación declarativa a la clasificación de documentos

Francisco P. Romero¹ Pascual Julián Iranzo² Mateus Ferreira-Satler² Juan Gallardo-Casero²

¹ Escuela Universitaria de Ingeniería Técnica Industrial de Toledo,
Universidad de Castilla La Mancha,
FranciscoP.Romero@uclm.es

² Escuela Superior de Informática de Ciudad Real
Universidad de Castilla La Mancha,
Pascual.Julian@uclm.es, msatler@gmail.com, Juan.Gallardo@alu.uclm.es

Resumen

En este trabajo se presenta una aproximación a la clasificación de documentos sin necesidad de realizar un proceso de entrenamiento previo. En este caso el clasificador se define a partir de las descripciones semánticas de los conceptos relacionados con cada una de las categorías pre-existentes. Para ello se utilizan relaciones de proximidad entre conceptos extraídas de ontologías y tesauros de propósito general. La base declarativa del proceso de clasificación es un mecanismo de unificación borrosa que facilita un proceso de búsqueda flexible de los conceptos en los documentos independiente de la base de conocimiento semántica que sirva de soporte.

Palabras Clave: Clasificación de Documentos, Unificación Borrosa, Búsqueda Flexible, Relaciones de proximidad.

1. INTRODUCCIÓN

La clasificación es el proceso de búsqueda de una serie de modelos o funciones que permiten distinguir entre una serie de clases o conceptos establecidos a priori sobre los datos. Este modelo será utilizado para identificar la clase de los objetos que aún no hayan sido clasificados [5]. La clasificación de documentos, ubicada dentro del aprendizaje supervisado, es una tarea que consiste en determinar la clase o clases a las que pertenece un documento [11].

Dada la naturaleza de la clasificación de documentos, el proceso se realiza habitualmente en varios pasos. En primer lugar, se cuenta con ejemplos dados de documentos pertenecientes a las clases predefinidas. Todos estos ejemplos son etiquetados con respecto a una de estas clases para formar el conjunto documental de en-

trenamiento. El algoritmo de aprendizaje correspondiente realizará un proceso de inducción obteniendo como resultado una descripción de cada una de las clases definidas. Esta descripción será posteriormente utilizada para clasificar de forma automática los nuevos documentos, por lo que de ella depende el grado de exactitud del sistema clasificador. Uno de los problemas que presentan estos sistemas se refiere al aumento exponencial del volumen del conjunto de entrenamiento cuando se pretende lograr un grado de exactitud suficiente ante una estructura compleja y ante ejemplos difíciles de clasificar a priori.

En este trabajo se propone prescindir del procedimiento de entrenamiento del clasificador. Con esta finalidad se utilizará como base la descripción semántica de los conceptos vinculados a cada una de las categorías predefinidas, es decir, a partir del concepto que define cada categoría se obtendrán las relaciones semánticas positivas que tenga con otros conceptos. La utilización de un proceso de recuperación de información basado en conceptos nos permitirá superar las dificultades inherentes a la recuperación basada únicamente en aspectos lexicográficos [3] siempre y cuando se cuente con una definición correcta y concreta del propio concepto para lo cual se utilizarán tesauros y ontologías de propósito general como Wordnet [12] y Concept Net [10]. Estas descripciones conceptuales serán la entrada a un mecanismo operacional de unificación borrosa que permite la búsqueda flexible de conceptos en documentos [8].

El resto del artículo está organizado de la siguiente manera. La sección 2 incluye una breve semblanza sobre relaciones de proximidad entre conceptos. En la sección 3 se presenta el lenguaje Bousi~Prolog que proporciona los mecanismos necesarios para poder realizar la clasificación de los documentos de forma declarativa. En la sección 4 se presenta el experimento realizado con el fin de verificar el funcionamiento de esta aproximación. Por último, se finaliza con las conclusiones y el trabajo futuro en la sección 5.

2. RELACIONES DE PROXIMIDAD ENTRE CONCEPTOS

La proximidad entre conceptos permite definir cuán cerca o cuán similar es un concepto en relación a otro, es decir, permite identificar si dos conceptos se relacionan semánticamente. Normalmente, los valores de similitud varían dentro de un rango de 0 y 1.

Existen muchas propuestas y métodos en la literatura que tratan de estimar un valor de similitud entre conceptos. En [9] se presenta un estudio comparativo de algunos de estos métodos, destacando las ventajas y desventajas de cada uno de ellos. Para comparar las metodologías, los autores propusieron una taxonomía de propuestas, como muestra la Figura 1.

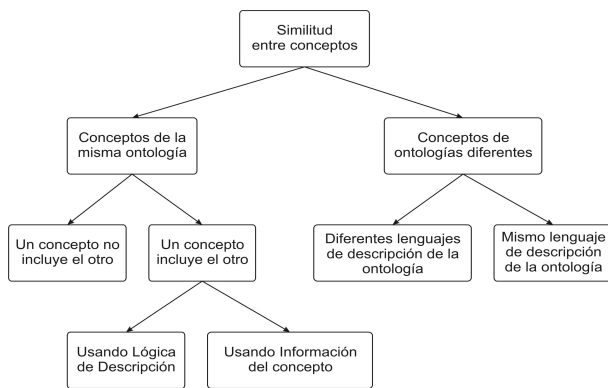


Figura 1: Taxonomía de propuestas para medir la similitud entre conceptos.

Dado que todo conjunto de conceptos relacionados puede ser incluido en una ontología, las propuestas se dividen en métodos para conceptos de la misma ontología y métodos para conceptos en ontologías diferentes. El primer tipo de métodos se basa en la idea de que un concepto incluye el otro, es decir, un concepto es una subclase o una superclase del otro en la jerarquía de la ontología. Si esto ocurre, las propuestas se subdividen en métodos basados en Lógica de Descripción y métodos basados en Información del concepto. Las propuestas que soportan conceptos de diferentes ontologías se separan en métodos que soportan ontologías descritas en el mismo lenguaje y métodos que soportan ontologías descritas en diferentes lenguajes.

En [16], los autores proponen un modelo para medir la similitud semántica entre conceptos con base en estudios filosóficos y de inteligencia artificial. El modelo usa características de conjuntos para describir conceptos. Un concepto consta de muchas características, al que llaman conjunto de características (features set). A estos conjuntos se aplican determinadas reglas y fórmulas para derivar la similitud de los conceptos.

Wang et al. [14] propone un modelo de similitud basado en el Análisis de Conceptos Formales (Formal Concept Analysis). La idea es que cuanto más objetos y atributos sean compartidos por dos conceptos, más similitud hay entre ellos. El método presentado combina características e informaciones estructurales en la toma de decisión, para intentar acercarse a la forma del discernimiento humano.

En [2], Formica presenta un método para evaluar la similitud de conceptos que tiene en cuenta la taxonomía y la estructura de los conceptos. Se basa en la asociación entre probabilidades y conceptos de la jerarquía. Se define la idea de similitud entre contenido de información, que se combina con la estructura de los conceptos y con una variable de peso definida por un experto en el dominio para generar un valor de similitud entre conceptos.

Para obtener la distancia o proximidad semántica entre conceptos será necesario recurrir a tesauros y ontologías como WordNet o Concept Net. Las relaciones de proximidad utilizadas en este trabajo están basadas en relaciones entre conceptos que proveen de forma directa o indirecta estos recursos de uso no específico.

En concreto, las medidas de proximidad conceptual que van a ser utilizadas como base de conocimiento para realizar las tareas de búsqueda flexible dentro del sistema Bousi~Prolog van a ser las siguientes.

- *Analogía estructural*[10]: No es una medida estrictamente semántica, sino que indica un grado de similitud en cuanto a las características del concepto representado por la palabra. Dos conceptos serán análogos si tienen características y funcionalidades similares.
- *Proximidad contextual*[10]: Refleja la proximidad entre dos conceptos en base a sus contextos habituales. Dado que Concept Net está concebida como una red semántica, la vecindad contextual alrededor de una palabra o nodo viene dada por la extensión del radio desde el nodo origen a los diferentes nodos relacionados. Se calcula contabilizando los caminos entre los dos nodos y la dirección de los arcos. Como resultado se aporta una medida de mutua información entre dos nodos.
- *Relaciones de similitud basadas en la sinonimia*: En [13] se presenta una fórmula basada en el coeficiente de Jaccard con el fin de calcular la similitud entre dos términos a partir de los conceptos que representan utilizando como base las relaciones semánticas existentes en WordNet.

3. EL LENGUAJE BOUSI~PROLOG Y LA BÚSQUEDA FLEXIBLE DE SOLUCIONES

Bousi~Prolog (BPL, para abreviar) [6, 7, 8] es un lenguaje de programación lógica donde la noción de “aproximación” se gestiona a un nivel sintáctico por medio de relaciones de proximidad. El objetivo principal de su diseño ha sido flexibilizar el proceso de repuesta a preguntas. Actualmente existen dos implementaciones de BPL:¹ una implementación de alto nivel [6], desarrollada mediante un metaintérprete escrito en Prolog, y otra de bajo nivel [7] que esencialmente es una adaptación de la Máquina Abstracta de Warren (WAM — Warren Abstract Machine) [15] (implementada en Java) para que pueda incorporar un algoritmo de unificación borroso.

Desde el punto de vista operacional, Bousi~Prolog extiende el lenguaje Prolog mediante un mecanismo de *resolución SLD débil* que es una adaptación del principio de resolución SLD clásico donde la unificación sintáctica se reemplaza por un algoritmo de unificación borrosa (o débil) basado en relaciones de proximidad. Informalmente este algoritmo de *unificación débil* establece que dos términos $f(t_1, \dots, t_n)$ y $g(s_1, \dots, s_n)$ unifican débilmente con respecto a una relación de proximidad $\mathcal{R}$ si f y g son próximos (esto es, si $\mathcal{R}(f, g) > 0$) y cada uno de sus argumentos t_i y s_i unifican débilmente. Por consiguiente, el algoritmo de unificación no produce un fallo cuando se confrontan dos símbolos sintácticamente distintos, siempre que sean próximos, sino un éxito con un cierto grado de aproximación que es el mínimo entre el grado de aproximación que es el mínimo entre el grado de aproximación, $\mathcal{R}(f, g)$, de los símbolos en cabeza y los grados de aproximación con que unifican débilmente cada uno de sus componentes. Así pues, Bousi~Prolog computa tanto respuestas computadas (sustituciones) como grados de aproximación.

En esencia, la sintaxis de Bousi~Prolog es como la de Prolog pero enriquecida con el símbolo predefinido “~~” empleado para describir las relaciones de proximidad mediante lo que denominamos una *ecuación de proximidad* de la forma: $\langle \text{symbol} \rangle \sim \sim \langle \text{symbol} \rangle = \langle \text{degree} \rangle$. Aunque, formalmente, una ecuación de proximidad representa una entrada de una relación borrosa binaria arbitraria, su lectura intuitiva es que dos constantes, símbolos de función n -arios o símbolos de predicado n -arios son próximos con un cierto grado. Informalmente, empleamos el símbolo predefinido “~~” como una notación comprimida para la clausura simétrica de una relación borrosa binaria arbitraria (esto es, una ecuación de proximidad $a \sim \sim b = \alpha$ puede entenderse en ambos

sentidos: a es próximo a b y b es próximo a a con grado α). Por consiguiente, en su forma más simple, un programa Bousi~Prolog es una secuencia de hechos y reglas Prolog seguido por una secuencia de ecuaciones de proximidad.

En este trabajo se han aprovechado las notables características del lenguaje Bousi~Prolog para la búsqueda flexible de soluciones a la hora de implementar el programa `inspect.bpl` (véase la Figura 2), que inspecciona una secuencia de documentos almacenados en un fichero cuya estructura interna es conforme al formato estándar SMART (véase la Figura 3). Este programa comprende una ontología de términos, modelada mediante ecuaciones de proximidad, y una serie de predicados, donde el predicado `inspect/3` constituye la interfaz principal con el usuario. De manera resumida, puede decirse que el predicado `inspect/3` lee, palabra por palabra, un fichero, denominado `FileName`, buscando palabras que son próximas (según la ontología definida) a una palabra clave, denominada `Keyword`. El término `Keyword` es un tópico que caracteriza el documento. Como resultado de la inspección, el predicado `inspect` devuelve un registro con datos estadísticos, denominado `DataAccount`, con la siguiente estructura:

```
[[texNumber(1)|L1], [texNumber(2)|L2], ...]
```

Esto es, existe una sublista para cada texto almacenado en el fichero `FileName`. Cada sublista L_i almacena una secuencia de triples $\mathbf{t}(X, N, D)$, donde X es un término próximo o similar al término `Keyword`, con grado D , que ocurre N veces en el texto `texNumber(i)`. Para realizar esta búsqueda de términos próximos o similares a uno dado, el programa se apoya en el mecanismo de unificación borrosa implementado en el núcleo del lenguaje Bousi~Prolog. Más concretamente, el predicado `processRemainderText` (véase la Figura 2) utiliza un operador de unificación débil, denotado por “~~”, que es la contrapartida difusa al operador de unificación sintáctica del lenguaje Prolog.

Una vez terminada la inspección de los textos, los datos contenidos en el registro `DataAccount` se utilizan en el análisis de los textos, con objeto de lograr su clasificación.

4. EXPERIMENTOS

En este apartado se presenta el experimento que se ha realizado sobre una colección estándar con el fin de validar la aproximación propuesta. El experimento es útil a su vez para determinar qué relaciones de proximidad entre documentos representan mejor el conocimiento de un experto al clasificar un documento en su correspondiente categoría.

¹<http://www.inf-cr.uclm.es/www/pjulian/bousi.html>

Reuters² es una colección de noticias que han sido formateadas y preparadas para su utilización en la evaluación de sistemas de categorización de documentos. Esto significa que está orientada al descubrimiento de los criterios que ha seguido el experto humano para clasificar un documento en una o varias categorías.

En concreto, esta colección consiste en 21.758 artículos generados por la agencia de noticias Reuters durante el año 1987. De entre estos documentos, únicamente algo más de la mitad de ellos (aprox. 12.900) tienen asignada una categoría de las 118 posibles definidas por la agencia, siendo posible asignar un documento a más de una categoría. Cada una de las categorías tiene un nivel diferente de importancia, existiendo categorías con menos de 10 documentos asignados y categorías que engloban a más de 1.000 documentos.

Para este experimento se han seleccionado de forma aleatoria 6 categorías de entre las 118 posibles. Además de esta selección de categorías hemos seleccionado aquellos documentos más significativos siguiendo el criterio propuesto en [1], con lo cual se utilizan un total de 160 documentos que pueden estar clasificados en una o varias categorías de forma simultánea (ver Tabla 1).

Tabla 1: Distribución de los documentos por clases.

Categoría	Documentos
wheat	71
sugar	36
rice	24
cotton	20
copper	18
lead	14

4.1. Procedimiento y resultados

Con la finalidad de realizar el experimento de clasificación de este conjunto de documentos a partir de las relaciones de proximidad conceptual anteriormente especificadas se definen los siguientes dos conjuntos de referencia.

- El conjunto T contiene las etiquetas que representan a cada una de las clases, de esta forma $T = \{wheat, sugar, rice, cotton, copper, lead\}$. Cada una de las etiquetas individuales es identificada mediante t_i .
- El conjunto D contiene los documentos que van a ser clasificados. Se representará con d_j a cada uno de los documentos del conjunto, mientras que t_{ij} representará la etiqueta teórica correspondiente

al documento d_j . Por otro lado, t'_{ij} representa a la etiqueta obtenida tras el proceso de búsqueda flexible obtenida en el paso 3 del proceso que se explica a continuación. Como ejemplo de uno de estos documentos, se dispone del siguiente texto (ya procesado previamente) que está vinculado a la etiqueta “wheat”:

agriculture,department,report,farm,own,
reserve,national,average,price,loan,
release,price,reserves,matured,bean,
enter,corn,sorghum,rates,bean,potato

Una vez establecidos estos conjuntos se realiza un proceso de clasificación de todos los documentos para cada una de las relaciones de proximidad especificadas en el subapartado anterior. Los pasos a seguir serán los siguientes:

1. Obtención de los grados de proximidad según la relación elegida para cada uno de las categorías predefinidas. Por ejemplo,

```
wheat~~bean=0.315.  bean~~corn=0.48.
wheat~~corn=0.315.  bean~~potato=0.5.
```

2. Se carga el programa resultante del proceso anterior en Bousi~Prolog dentro del programa `inspect.bpl` con el fin de que sea utilizada para realizar las búsquedas flexibles que permitirán clasificar los documentos.
3. Buscar dentro de cada documento d_j cada una de las etiquetas t_i , con ello se obtiene una serie de grados de presencia de t_i en el documento d_j que vendrá representado por w_{ij} ($w_{ij} = g(t_i, d_j)$). Siendo g el predicado de búsqueda flexible que permite a Bousi~Prolog encontrar t_i en d_j sin necesidad de que éste aparezca explícitamente.

```
BPL> sv g(wheat,L)
L= [ (potato, 1, 0.107),
      (bean, 2, 0.315),
      (corn, 1, 0.315)]
w = 0,864
```

4. La etiqueta ganadora t'_{ij} corresponderá a aquella que obtenga un mayor peso ($t'_{ij} = \arg \max(w_{ij})$). Ésta será comparada con la categoría teórica (t_{ij}) y si coincide será una clasificación correcta. El porcentaje de clasificaciones correctas nos arrojará el grado de bondad que ofrece el uso de la ontología en el proceso de clasificación.

²www.daviddlewis.com/resources/testcollections

El proceso se ha realizado tanto con cada una de las relaciones semánticas definidas anteriormente como sin utilizar aportación semántica alguna, es decir, buscando únicamente mediante la igualdad sintáctica. Los resultados obtenidos se pueden observar en el tabla 2. La utilización de la relación de analogía estructural ha arrojado un aceptable número de elementos no clasificados que pertenecen casi al 50 % a los documentos que pertenecen a la categoría “lead” lo que no es sorprendente dado que la descripción del concepto utilizada no es suficientemente completa. Algo similar sucede al utilizar la proximidad contextual. En este caso, tanto el número de no clasificados como de errores cometidos es aún mayor. Los mejores resultados los ha ofrecido la utilización de la relación descrita en [13] ya que cuenta con una definición de los conceptos de mayor calidad y completitud. Otra conclusión que se puede obtener es la mejora considerable que se obtiene en la eficacia del proceso de clasificación al utilizar relaciones semánticas ya que se logra prácticamente doblar el número de documentos correctamente clasificados con respecto a la utilización exclusiva de la igualdad sintáctica.

Tabla 2: Resultados obtenidos.

Relación	Correctos	Incorrectos
		No Clasif/Errores
Analogía	74 %	16 %/10 %
Prox. Context.	59 %	30 %/11 %
Sinonimia	83 %	6 % / 11 %
Igualdad Sint.	40 %	48 %/ 12 %

5. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha presentado una aproximación a la clasificación de documentos no basada en un proceso de entrenamiento sino en la definición mediante relaciones semánticas de los conceptos que describen a cada una de las clases objetivo. A partir de esta definición, la utilización de un lenguaje de programación lógica como Bousi~Prolog permite realizar una búsqueda flexible de los conceptos que representan a las categorías dentro de los diferentes textos a clasificar. Este proceso arroja como resultado la clasificación en una o más categorías de cada uno de los documentos. El rendimiento obtenido en los experimentos es satisfactorio siempre teniendo en cuenta la dependencia existente con respecto a las relaciones semánticas utilizadas. Esto ha generado la aparición de ciertos problemas cuando la definición del concepto correspondiente a la categoría no es suficientemente completa.

Como trabajo futuro, es necesario en primer lugar, abordar el problema de escalabilidad del sistema de búsqueda flexible, ya que no permite realizar búsquedas sobre colecciones con más de 200 documentos. Una vez resuelto este problema se podrán realizar experimentos en los que se cuente con un volumen tanto de documentos como de relaciones semánticas mucho mayor. A su vez, será interesante comprobar la eficacia de la utilización de relaciones semánticas más complejas como las proporcionadas por la librería WordNetSimilarity³ o las estudiadas en [4] basadas en la versión más reciente Concept Net. Otro proceso a llevar a cabo será la evaluación de la mejora de la eficacia en el proceso de clasificación mediante la utilización de las ontologías de propósito específico mezcladas con las ontologías de propósito general ya mencionadas.

Agradecimientos

La investigación realizada está financiada por los proyectos TIN2007-67494 y TIN2007-65749 del Ministerio de Ciencia e Innovación, los proyectos PEIC09-0196-3018 y PII109-0117-4481 de la Junta de Comunidades de Castilla-La Mancha.

Referencias

- [1] C. Apté and F. Damerau and S.M. Weis, Automated Learning of Decision Rules for Text Categorization, ACM Transactions on Information Systems, pp. 23-30, (1994).
- [2] A. Formica. Concept similarity by evaluating information contents and feature vectors: a combined approach. Commun. ACM. 52(3) p. 145-149. (2009).
- [3] P. Garces and Jose A. Olivas and F. P. Romero. Concept-Matching IR Systems Versus Word-Matching Information Retrieval Systems: Considering Fuzzy Interrelations for Indexing Web Pages Journal of the American Society for Information Science and Technology, 57(4):564-576, 2006.
- [4] S. Guadarrama and M. Garrido Concept-Analyzer: A tool for analyzing fuzzy concepts In: Proceedings of Information Processing and Management Uncertainty Conference (IPMU'2008). 1084-1089. (2008).
- [5] J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann. (2001).

³<http://www.d.umn.edu/mich0212/cgi-bin/similarity/similarity.cgi>

- [6] P. Julián, C. Rubio and J. Gallardo. Bousi~prolog: a Prolog extension language for flexible query answering. In: ENTCS, vol 248, pp. 131-147. Elsevier, Amsterdam (2009).
- [7] P. Julián and C. Rubio. A similarity-based WAM for Bousi~Prolog. In: LNCS, vol 5517, pp. 245–252. Springer, Heidelberg (2009).
- [8] P. Julián and C. Rubio. A Declarative Semantics for Bousi~Prolog. In: Proceedings of 11th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming (2009).
- [9] D.N. Le, A.E.S. Goh, Current Practices in Measuring Ontological Concept Similarity, in Proceedings of the Third International Conference on Semantics, Knowledge and Grid. IEEE Computer Society. p. 266-269. (2007).
- [10] H. Liu and P. Singh. ConceptNet: A Practical Commonsense Reasoning Toolkit. BT Technology Journal, Vol. 22(4). (2004).
- [11] D. Meretakis and D. Fragoudis and D. Lu and S. Likothanassis. Scalable associationbased text classification. In Proc. 9th ACM Int. Conf. Information and Knowledge Management, Washington, US, pp. 5-11. (2000).
- [12] G.A. Miller and R. Beckwith and C. Fellbaum. Introduction to wordnet: an on-line lexical database. International Journal of Lexicography 3(4):235–244. (1990).
- [13] A. Soto and J.A. Olivas and M.E. Prieto. Fuzzy Approach of Synonymy and Polysemy for Information Retrieval. Proceedings of the International Symposium on Fuzzy and Rough Sets, ISFUROS-2006, Santa Clara, Cuba. (2006).
- [14] L. Wang and X. Liu. A new model of evaluating concept similarity. Knowledge-Based Systems, 2008. 21(8): p. 842-846.
- [15] D. H. Warren. An Abstract Prolog Instruction Set. Technical note 309, SRI International, Menlo Park, CA., October (1983).
- [16] C. Wu and D. Dai and Y. Wan, Ontology Concept Similarity in Semantic Query, in Proceedings of the 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery. IEEE Computer Society. p. 24-28. (2008)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% inspect(FileName, Keyword, DataAccount),
%% inspect(i, i, o)

inspect(FileName, Keyword, DataAccount) :-
    seeing(OLD),
    see(FileName),
    write('Processing file, '), write(FileName),
    write('). This may take a while ...'), nl,
    processFile(Keyword, DataAccount),
    seen,
    see(OLD).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% processFile(Keyword, DataAccount)
%% processFile(i, o)

processFile(Keyword, DataAccount):-processDocs(Keyword, [], DataAccount).

/*
processDocs/3,
lee una palabra, comprueba que es un delimitador
de identificador de documento ".I",
lee el identificador del documento
y eventualmente llama a processText/4.

processText/4,
lee una palabra, comprueba que es un delimitador de texto
".W" y lee y analiza el texto.
*/
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% processText(Keyword, AccAccount, NewAccAccount, EOF)
%% processText(i, i, o, o)

processText(Keyword, AccAccount, NewAccAccount, EOF) :-
    read_word(Word1,EOF1),
    (EOF1 = false ->
        (isTextDelimiter(Word1) ->
            processRemainderText(Keyword, [], TextAccount, EOF2),
            NewAccAccount = [TextAccount|AccAccount], EOF = EOF2;
            write('This file does not conform the standards.'),
            nl, NewAccAccount = [['ERROR'|AccAccount], EOF = EOF1
        );
        write('End of file reached.'), nl,
        NewAccAccount = AccAccount, EOF = EOF1
    ).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% processRemainderText(Keyword, TextAccAccount, TextAccount, EOF)
%% processRemainderText(i, i, o, o)

processRemainderText(Keyword, TextAccAccount, TextAccount, EOF) :-
    read_word(Word1,EOF1),
    name(Wordname, Word1),
    (EOF1 = false ->
        (isDocIdentifierDelimiter(Word1) ->
            TextAccount = TextAccAccount,
            EOF = EOF1;
            (Keyword == Wordname = AD ->
                (insert(t(Wordname,1,AD), TextAccAccount, NewTextAccAccount),
                    processRemainderText(Keyword,NewTextAccAccount,TextAccount,EOF)
                );
            processRemainderText(Keyword, TextAccAccount, TextAccount, EOF)
        );
        TextAccount = TextAccAccount, EOF = EOF1
    ).

```

Figura 2: Fragmento del programa inspect.bp1.

```

.I 1
.W
<Texto Documento 1>
.I 2
.W
<Texto Documento 2>
.
.
.
.I N
.W
<Texto Documento N>

```

Figura 3: Estructura de los documentos analizados.

TRABAJANDO CON RELACIONES DE PREFERENCIA LINGÜÍSTICAS DIFUSAS INCOMPLETAS PARA RECOMENDAR RECURSOS DE INVESTIGACIÓN EN UNA BIBLIOTECA DIGITAL UNIVERSITARIA

Carlos Porcel¹ Enrique Herrera-Viedma²

¹ Universidad de Jaén, Dept. Informática, cporcel@ujaen.es

² Universidad de Granada, Dept. Ciencias de la Computación e IA, viedma@decsai.ugr.es

Resumen

En trabajos previos hemos propuesto el uso de sistemas de recomendaciones lingüísticos difusos para difundir recursos de investigación en bibliotecas digitales universitarias. El problema es que el proceso de adquisición de las preferencias de los usuarios es bastante complejo y requiere demasiado esfuerzo por parte de los mismos. En este trabajo presentamos un sistema de recomendaciones lingüístico difuso que facilita ese proceso permitiendo que los usuarios expresen sus preferencias mediante relaciones de preferencia lingüísticas difusas incompletas. Se incluyen herramientas para gestionar la información incompleta cuando los usuarios expresan sus preferencias, mejorando la adquisición de los perfiles de usuario.

Palabras Clave: sistemas de recomendaciones, modelado lingüístico difuso, relaciones de preferencia incompletas.

1 INTRODUCCIÓN

Las bibliotecas digitales son colecciones de información que tienen asociados una serie de servicios que ofrecen a su comunidad de usuarios [4, 14]. Se han aplicado en numerosos ámbitos, pero nosotros nos vamos a centrar en un contexto académico. El acceso a Internet está motivando el uso cada vez más extendido de estas bibliotecas y que sus contenidos sean más variados, por lo que los usuarios esperan servicios cada vez más sofisticados [4, 14]. Un servicio particularmente importante es la difusión selectiva de información, para ayudar a los usuarios a acceder a información interesante para ellos. Los usuarios desarrollan perfiles de

interés, y conforme se añade nuevo material (libros, artículos, informes, etc.) a la colección, los ítems relevantes son recomendados a los usuarios.

Los sistemas de recomendaciones evalúan y filtran la gran cantidad de información disponible y asisten a los usuarios en sus procesos de acceso a la información [3, 13]. En el marco de las Bibliotecas Digitales Universitarias (BDU), los sistemas de recomendaciones se pueden usar para ayudar a sus usuarios (profesores, estudiantes y personal de biblioteca) a encontrar y seleccionar información y fuentes de conocimiento [10].

En un sistema de recomendaciones, se usan las preferencias de los usuarios para definir perfiles de usuarios que son aplicados como filtros sobre el flujo de documentos [3, 13]. En [11, 12] desarrollamos sistemas de recomendaciones en un entorno académico. En [11] propusimos un sistema de recomendaciones lingüístico difuso para una oficina de transferencia de tecnología para ayudar a los investigadores y empresas a obtener automáticamente información sobre recursos de investigación en sus áreas de interés. En [12] propusimos un sistema de recomendaciones lingüístico difuso que recomienda recursos especializados pero también complementarios relacionados con las áreas de investigación. El problema de esos sistemas es que los usuarios deben especificar directamente sus preferencias, lo cual implica demasiado esfuerzo para ellos. En algunos casos son pocas las categorías que tienen que valorar, pero hay casos en los que el número podría ser mucho mayor; por ejemplo en [11] se trabaja con 248 categorías.

Sin embargo, un sistema de recomendaciones se puede ver como un problema de ayuda a la decisión, donde las posibles soluciones son los recursos de la BDU, y los criterios a satisfacer son los perfiles de los usuarios. Podríamos, pues, adoptar los formatos de representación típicos en toma de decisiones, como las relaciones de preferencias difusas que presentan un alto nivel de expresividad [6]. Sin embargo, hay situa-

ciones en las que los usuarios no son capaces de proporcionar todos los valores requeridos, por lo que tenemos que trabajar con *relaciones de preferencia incompletas* [1, 2, 9].

En este trabajo presentamos un nuevo sistema de recomendaciones lingüístico difuso definido en el entorno de una BDU en el que proponemos una forma alternativa de obtener conocimiento preciso y útil sobre las preferencias de los usuarios. El sistema permite a los usuarios expresar sus preferencias por medio de relaciones de preferencia incompletas [1], que se completan estimando los valores ausentes usando los métodos propuestos en [1, 2, 9], que se basan en la propiedad de consistencia aditiva y usando sólo valores de preferencia conocidos. Ello permite reducir la intervención por parte de los usuarios y mejorar el proceso de establecimiento de sus perfiles. El sistema propuesto recomienda recursos de investigación que mejor satisfacen las necesidades de los usuarios, así como posibilidades de colaboración para conocer investigadores de áreas relacionadas con los que podrán colaborar.

Como en [11, 12] aplicamos el Modelado Lingüístico Difuso (MLD) multi-granular [8], que aporta técnicas que permiten representar los distintos conceptos del sistema con diferentes conjuntos de etiquetas lingüísticas.

El trabajo se estructura de la siguiente forma. En la sección 2 se describe el MLD. En la sección 3 presentamos el nuevo modelo para filtrar recursos en una BDU. Por último, analizaremos las conclusiones obtenidas.

2 MODELADO LINGÜÍSTICO DIFUSO

El MLD es una herramienta basada en el concepto de variable lingüística [15] que ha dado muy buenos resultados para el modelado de información cualitativa en multitud de problemas.

2.1 MODELADO LINGÜÍSTICO DIFUSO 2-TUPLAS

El MLD 2-tuplas [7] es un modelo continuo de representación de la información que permite reducir la pérdida de información típica de otros enfoques lingüísticos (clásico y ordinal [5, 15]). Para definirlo, tenemos que establecer el modelo de representación y el modelo computacional de las 2-tuplas para representar y agregar respectivamente la información lingüística.

Consideremos que $\mathcal{S} = \{s_0, \dots, s_g\}$ es un conjunto de términos lingüísticos con cardinalidad impar, donde la semántica asociada con cada una de las etiquetas viene

dada por medio de funciones de pertenencia triangulares, y consideramos todos los términos distribuidos sobre una escala sobre la que hay establecida una relación de orden total. Si mediante un método de agregación de información lingüística, obtenemos un valor $\beta \in [0, g]$, y $\beta \notin \{0, \dots, g\}$, podemos representar β como una 2-tupla (s_i, α_i) , donde s_i representa la etiqueta lingüística, y α_i es un valor numérico que expresa la traslación de β con respecto al índice de la etiqueta más cercana, i , en el conjunto de términos lingüísticos ($s_i \in \mathcal{S}$). Este modelo define un conjunto de funciones de transformación entre valores numéricos y 2-tuplas: $\Delta(\beta) = (s_i, \alpha)$ y $\Delta^{-1}(s_i, \alpha) = \beta \in [0, g]$ [7].

Para establecer el modelo computacional, definimos operadores de negación, de comparación y de agregación. Usando las funciones Δ y Δ^{-1} , podemos extender cualquiera de los operadores de agregación ya definidos para trabajar con 2-tuplas [7]. Algunos ejemplos son:

Definición 2.1. *Media aritmética.* Sea $x = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ un conjunto de 2-tuplas, la 2-tupla que representa la media aritmética $\bar{x}^e$ se calcula como:

$$\bar{x}^e[(r_1, \alpha_1), \dots, (r_n, \alpha_n)] = \Delta\left(\sum_{i=1}^n \frac{1}{n} \Delta^{-1}(r_i, \alpha_i)\right) = \Delta\left(\frac{1}{n} \sum_{i=1}^n \beta_i\right).$$

Definición 2.2. *Operador de media ponderada lingüística.* Sea $x = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ un conjunto de 2-tuplas y $W = \{(w_1, \alpha_1^w), \dots, (w_n, \alpha_n^w)\}$ sus pesos asociados con valores 2-tupla. La media ponderada lingüística $\bar{x}_l^w$ es:

$$\bar{x}_l^w[((r_1, \alpha_1), (w_1, \alpha_1^w)) \dots ((r_n, \alpha_n), (w_n, \alpha_n^w))] = \Delta\left(\frac{\sum_{i=1}^n \beta_i \cdot \beta_{w_i}}{\sum_{i=1}^n \beta_{w_i}}\right),$$

con $\beta_i = \Delta^{-1}(r_i, \alpha_i)$ y $\beta_{w_i} = \Delta^{-1}(w_i, \alpha_i^w)$.

2.2 MODELADO LINGÜÍSTICO DIFUSO MULTI-GRANULAR

En cualquier enfoque, un parámetro importante que hay que determinar es la “granularidad de incertidumbre”, es decir, la cardinalidad del conjunto de términos lingüísticos $\mathcal{S}$. Según el grado de incertidumbre que un experto encargado de cualificar un fenómeno tenga

sobre el mismo, el conjunto de términos lingüísticos elegido para proporcionar ese conocimiento tendrá más o menos términos. Cuando distintos expertos tienen diferentes grados de incertidumbre sobre el fenómeno o cuando un experto tiene que valorar distintos conceptos, se necesitan varios conjuntos de términos lingüísticos de diferente granularidad [8]. En tales situaciones necesitamos herramientas que nos permitan gestionar la información lingüística multi-granular. En [8] se propuso un MLD 2-tuplas multi-granular basado en el concepto de jerarquía lingüística. Una *Jerarquía Lingüística, LH*, es un conjunto de niveles $l(t, n(t))$, donde cada nivel t es un conjunto de términos lingüísticos con una granularidad $n(t)$ diferente [8]. Los niveles están ordenados por granularidad, de manera que podemos definir un nivel a partir del anterior: $l(t, n(t)) \rightarrow l(t+1, 2 \cdot n(t) - 1)$. En [8] se definió una familia de funciones de transformación entre etiquetas de diferentes niveles. Para establecer el modelo computacional seleccionamos un nivel que usamos para uniformizar la información y así podemos usar los operadores definidos en el modelo 2-tuplas.

2.3 RELACIONES DE PREFERENCIA LINGÜÍSTICAS DIFUSAS INCOMPLETAS

Definición 2.3. Una relación de preferencia difusa P sobre un conjunto de alternativas $X = \{x_1, \dots, x_n\}$ es un conjunto difuso sobre $X \times X$, es decir, es caracterizado por una función de pertenencia $\mu_P: X \times X \rightarrow [0, 1]$.

Cuando la cardinalidad de X es pequeña, la relación de preferencia puede ser convenientemente representada mediante la matriz $P = (p_{ij})$, con $p_{ij} = \mu_P(x_i, x_j)$ ($\forall i, j \in \{1, \dots, n\}$) indicando el grado de preferencia de la alternativa x_i sobre x_j , donde: $p_{ij} = 1/2$ indica indiferencia entre x_i y x_j , $p_{ij} = 1$ indica que x_i es absolutamente preferida a x_j , y $p_{ij} > 1/2$ indica que x_i es preferida a x_j .

Sin embargo, nuestro sistema integra el MLD multi-granular basado en 2-tuplas, por lo que definimos una relación de preferencia de la siguiente manera.

Definición 2.4. Sea $X = \{x_1, \dots, x_n\}$ un conjunto de alternativas y S un conjunto de términos lingüísticos. Una relación de preferencia lingüística $P = p_{ij}$ ($\forall i, j \in \{1, \dots, n\}$) sobre X es:

$$\mu_P: X \times X \rightarrow S \times [0.5, 0.5)$$

donde $p_{ij} = \mu_P(x_i, x_j)$ es una 2-tupla que denota el grado de preferencia de la alternativa x_i respecto x_j .

Sin embargo, en problemas reales, los expertos no

siempre son capaces de proporcionar todos los valores que son requeridos. Para modelar estas situaciones, definimos el concepto de relación de preferencia difusa incompleta [1, 2].

Definición 2.5. Una función $f: X \rightarrow Y$ es *parcial* cuando no ocurre que cada elemento del conjunto X necesariamente se corresponde con un elemento del conjunto Y .

Definición 2.6. Una *relación de preferencia difusa incompleta* P sobre un conjunto de alternativas X es un conjunto difuso sobre el conjunto $X \times X$ que se caracteriza mediante una función de pertenencia *parcial*.

3 DISEÑO DEL SISTEMA

En esta sección presentamos un nuevo sistema de recomendaciones lingüístico difuso en el que los perfiles de los usuarios se obtienen a partir de sus preferencias representadas por medio de relaciones de preferencia lingüísticas difusas incompletas [1]. Este sistema facilita la expresión de las preferencias de los usuarios y reduce el esfuerzo realizado por parte de los mismos. El sistema se usa para recomendar a los usuarios de una BDU sobre los recursos de investigación que mejor satisfacen sus preferencias, así como sugerencias de colaboración con otros investigadores.

En la figura 1 podemos observar el esquema básico de funcionamiento, que será explicado en las siguientes secciones.

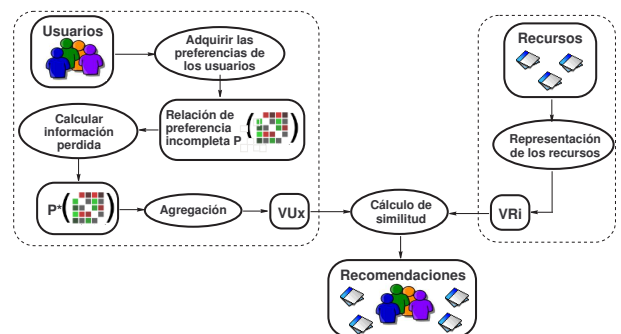


Figura 1: Esquema de funcionamiento.

3.1 REPRESENTACIÓN DE LA INFORMACIÓN

Para representar los distintos conceptos que tienen que ser valorados en la actividad del sistema, vamos a usar distintos conjuntos de etiquetas ($S_1, S_2, \dots$) seleccionados de una jerarquía lingüística [8]. Distinguimos cuatro conceptos que tienen que ser valorados:

- **Grado de importancia** (S_1) de una disciplina con respecto al ámbito de un recurso o las preferencias de un usuario.
- **Grado de relevancia** (S_2) de un recurso para un usuario.
- **Grado de compatibilidad** (S_3) entre dos usuarios.
- **Grado de preferencia** (S_4) de un recurso respecto de otro.

En nuestro sistema usamos conjuntos de etiquetas seleccionados de 2 niveles de una jerarquía lingüística de 3 niveles de 3, 5 y 7 etiquetas cada uno. En concreto, se usa el segundo nivel para asignar grados de importancia y grados de preferencia ($S_1 = S^5$ y $S_4 = S^5$), y el tercer nivel para asignar grados de relevancia y de compatibilidad ($S_2 = S^9$ y $S_3 = S^9$).

3.1.1 Representación de los recursos

Los recursos considerados son artículos de revista, contribuciones a congresos, capítulos de libro, libros o libros editados. El sistema obtiene una representación interna basada principalmente en el ámbito del recurso, para lo cual usamos una clasificación compuesta de 25 disciplinas (ver figura 2). Entonces, para representar un recurso i adoptamos un modelo vectorial, es decir, obtenemos un vector $VR_i = (VR_{i1}, VR_{i2}, \dots, VR_{i25})$, donde $VR_{ij} \in S_1$, con $j = \{1, \dots, 25\}$, indica el grado de importancia de la disciplina j con respecto al ámbito del recurso i .



Figura 2: Disciplinas para definir el ámbito de los recursos.

3.1.2 Perfiles de usuario

Los perfiles se componen de dos tipos de preferencias: sobre los tópicos de interés y sobre si desean o no cola-

borar con otros usuarios. La principal contribución de esta propuesta es la forma en que los usuarios proporcionan estas preferencias. Si usamos el modelo vectorial, como en trabajos previos, los usuarios deben insertar o modificar los valores asignados a las disciplinas, pero puede haber casos en los que el número sea muy elevado. Ello implicaría un esfuerzo aún mayor. En este modelo proponemos un método alternativo para obtener las preferencias de los usuarios sobre los tópicos de interés.

Pedimos a los usuarios que proporcionen sus preferencias sobre 5 recursos, usando una relación de preferencia lingüística difusa incompleta. Además, de acuerdo a los resultados presentados en [2], basta con que los usuarios proporcionen una fila de la relación. A continuación usamos el método propuesto en [2] para completar la relación, y entonces podemos obtener un vector que representa las preferencias de los usuarios sobre sus tópicos de interés. Veámos el proceso en detalle:

1. Adquirir las **preferencias de los usuarios** sobre un número limitado de recursos. El sistema muestra a los usuarios los 5 recursos más representativos, $R = \{r_1, \dots, r_5\}$ y les pide que expresen sus preferencias mediante una relación de preferencia lingüística difusa incompleta. En la relación de preferencia, cada valor $p_{ij} \in S_4$ representa el grado de preferencia de i sobre j . Bastaría con proporcionar una única fila:

$$P = \begin{pmatrix} - & p_{12} & p_{13} & p_{14} & p_{15} \\ x & - & x & x & x \\ x & x & - & x & x \\ x & x & x & - & x \\ x & x & x & x & - \end{pmatrix} \quad (1)$$

Entonces el sistema completa la relación P usando el método propuesto en [2], y obtenemos la relación P^* :

$$P^* = \begin{pmatrix} - & p_{12} & p_{13} & p_{14} & p_{15} \\ p_{21}^* & - & p_{23}^* & p_{24}^* & p_{25}^* \\ p_{31}^* & p_{32}^* & - & p_{34}^* & p_{35}^* \\ p_{41}^* & p_{42}^* & p_{43}^* & - & p_{45}^* \\ p_{51}^* & p_{52}^* & p_{53}^* & p_{54}^* & - \end{pmatrix} \quad (2)$$

donde $p_{1j} \in S_4$ son los grados insertados por el usuario sobre las preferencias del recurso x_1 con respecto a x_j , p_{ii} indica indiferencia, y cada p_{ij}^* es el grado estimado para el usuario sobre sus preferencias del recurso x_i con respecto a x_j .

2. Obtener el **vector de preferencias del usuario**. Para calcular los grados de preferencia del recurso i para un experto DG , podemos aplicar el operador de media aritmética definido en la definición

2.1 [7]: $DG_i = \bar{x}^e[p_{i1}^*, \dots, p_{i5}^*]$. Ahora podemos obtener el vector de preferencias del usuario x , es decir, $VU_x = (VU_{x1}, VU_{x2}, \dots, VU_{x25})$, a partir de la agregación de los vectores que representan las características de los recursos seleccionados ($\{VR_1, \dots, VR_5\}$) ponderados por los grados de preferencias $\{DG_1, \dots, DG_5\}$, usando el operador $\bar{x}_l^w$ definido en la definición 2.2 [7]: $VU_{xk} = \bar{x}_l^w[(VR_{1k}, DG_1), \dots, (VR_{5k}, DG_5)]$, con $k = \{1, \dots, 25\}$.

Completamos el perfil del usuario consultando sus preferencias de colaboración, es decir, si desea colaborar o no.

3.2 ESQUEMA DE RECOMENDACIÓN

En esta fase se generan las recomendaciones que se envían a los usuarios. Nos basamos en un proceso de cálculo de similitud entre los perfiles de los usuarios y la representación de los recursos. Como los componentes de los vectores son valores lingüísticos, la similitud entre dos vectores $(\sigma_l(V_1, V_2) \in S_1)$ la calculamos definiendo la medida del coseno en un entorno lingüístico:

$$\sigma_l(V_1, V_2) = \Delta(g \times \frac{\sum_{k=1}^n (\bar{h}_1 \times \bar{h}_2)}{\sqrt{\sum_{k=1}^n (\bar{h}_1)^2} \times \sqrt{\sum_{k=1}^n (\bar{h}_2)^2}})$$

donde g es la granularidad de S_1 , $\bar{h}_i = \Delta^{-1}(v_{ik}, \alpha_{vik})$ y (v_{ik}, α_{vik}) es el 2-tupla del término k en el vector V_i .

Cuando se inserta un nuevo recurso i , calculamos $\sigma_l(VR_i, VU_j)$, entre el vector que representa este nuevo recurso (VR_i) y los vectores de preferencias de todos los usuarios, $\{VU_1, \dots, VU_m\}$, donde m es el número de usuarios en el sistema.

Si $\sigma_l(VR_i, VU_j) \geq (s_6^9, 0)$ (umbral previamente definido para filtrar la información que se modifica en función de los resultados que se vayan obteniendo), el usuario j es seleccionado para recomendarle el recurso i . A continuación el sistema aplica a cada $\sigma_l(VR_i, VU_j)$ la función de transformación definida en [8], para obtener el grado de relevancia del recurso i para el usuario j , expresado usando una etiqueta del conjunto S_2 .

Las preferencias de colaboración se usan para clasificar a los usuarios en colaboradores $\mathcal{U}_C$ y no colaboradores $\mathcal{U}_N$. Para los usuarios de $\mathcal{U}_N$ el sistema les envía la información del recurso junto con su grado estimado de relevancia.

Para los usuarios de $\mathcal{U}_C$ el sistema calcula las posibilidades de colaboración. Para ello, entre cada dos usuarios $x, y \in \mathcal{U}_C$:

1. Calcular $\sigma_l(VU_x, VU_y)$.
2. Obtener el grado de compatibilidad lingüístico entre ambos usuarios, que será expresado como etiqueta de S_3 .

Por último se envía a los usuarios de $\mathcal{U}_C$ la información sobre el recurso, su grado de relevancia lingüístico estimado y las recomendaciones sobre colaboraciones junto con su grado de compatibilidad lingüístico.

3.3 EVALUACIÓN DEL SISTEMA

Actualmente tenemos implementada una versión de pruebas, en la que el sistema funciona sólo con unos pocos investigadores. Esta versión ha sido usada para probar la funcionalidad, pero estamos trabajando para obtener una versión más completa. Cuando los usuarios reciben recomendaciones, valoran la relevancia de dichas recomendaciones y usamos estas opiniones para evaluar el sistema, comparándolas con las sugerencias del personal de biblioteca.

Para este primer estudio consideramos un conjunto de datos con 30 recursos de distintas áreas que fueron insertados en el sistema. Inicialmente limitamos estos experimentos a 6 usuarios que completaron el proceso de registro e insertaron sus preferencias acerca de los 5 recursos más relevantes presentados por el sistema. Los perfiles obtenidos a partir de dicha información junto con los recursos previamente insertados, constituyeron el conjunto de entrenamiento. Entonces, añadimos 20 nuevos recursos que constituyeron el conjunto de datos de pruebas. El sistema filtró estos 20 recursos y los recomendó a los usuarios apropiados. Para obtener datos con los que comparar, los 20 recursos también fueron recomendados usando las sugerencias del personal de biblioteca.

Con esta información podemos calcular la precisión (proporción de ítems relevantes seleccionados con respecto al total de ítems seleccionados), recall (proporción de ítems relevantes seleccionados con respecto al total de ítems relevantes) y F1 (combinación de ambas: $F1 = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}}$), que son las métricas habitualmente usadas para evaluar la calidad de las recomendaciones [10]. Las medias para la precisión, recall y F1 son de 67.50%, 61.39% y 63.51%, respectivamente. Aunque se apliquen en ámbitos ligeramente distintos, y con distintos experimentos, si comparamos este valor F1 con los obtenidos en sistemas anteriores [11, 12], 54.49% y 65.05% respectivamente, observamos un buen rendimiento del sistema propuesto, a falta, eso sí, de una comparación más precisa con algún sistema que aplique un esquema similar, usando datos reales y planteando los mismos experimentos. Pero a pesar de estos resultados, destacamos que ahora la ob-

tención de perfiles de usuario requiere menor esfuerzo por parte de los mismos.

4 CONCLUSIONES

Hemos presentado un sistema de recomendaciones lingüístico difuso multi-granular para la difusión de recursos de investigación pero integrando un nuevo método para facilitar la adquisición de las preferencias de los usuarios. Éstos no proporcionan directamente sus vectores de preferencias, sino que proporcionan dichas preferencias por medio de relaciones de preferencia lingüísticas difusas incompletas. Los perfiles de los usuarios son completados con sus preferencias sobre si desean colaborar o no con otros usuarios. Así, este sistema de recomendaciones actúa como un sistema de ayuda a la decisión que toma decisiones sobre los recursos que más podrían interesar a un investigador, así como sobre posibilidades de colaboración con otros investigadores para formar grupos de trabajo. Los resultados experimentales obtenidos muestran la satisfacción de los usuarios con las recomendaciones recibidas. Como principal trabajo futuro, nos planteamos estudiar la viabilidad sobre la obtención de las preferencias de los usuarios de forma automática, lo cual facilitaría aún más dicho proceso a pesar del coste computacional que ello supondría.

Agradecimientos

Este trabajo ha sido desarrollado con la financiación de los proyectos Proyecto de Excelencia Andaluz TIC05299, FUZZYLING (TIN2007-61079) y PETRI (PET2007-0460).

Referencias

- [1] S. Alonso, F.J. Cabrerizo, F. Chiclana, F. Herrera, E. Herrera-Viedma. Group decision-making with incomplete fuzzy linguistic preference relations. *International Journal of Intelligent Systems*, 24(2), 201-222, 2009.
- [2] S. Alonso, F. Chiclana, F. Herrera, E. Herrera-Viedma, J. Alcalá-Fdez, C. Porcel. A Consistency-Based Procedure to Estimating Missing Pairwise Preference Values. *International Journal of Intelligent Systems*, 23, 155-175, 2008.
- [3] R. Burke. Hybrid Web Recommender System. *P. Brusilovsky, A. Kobsa, and W. Nejdl (Eds.): The Adaptive Web, LNCS 4321*, 377-408, 2007.
- [4] M. A. Gonçalves, E. A. Fox, L. T. Watson, N. A. Kipp. Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries. *ACM Transactions on Information Systems*, 22(2), 270-312, 2004.
- [5] F. Herrera, E. Herrera-Viedma. Aggregation operators for linguistic weighted information. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems*, 27, 646-656, 1997.
- [6] F. Herrera, E. Herrera-Viedma. Choice functions and mechanisms for linguistic preference relations. *European Journal of Operational Research*, 120, 144-161, 2000.
- [7] F. Herrera, L. Martínez. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 8(6), 746-752, 2000.
- [8] F. Herrera, L. Martínez. A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making. *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*, 31(2), 227-234, 2001.
- [9] E. Herrera-Viedma, F. Chiclana, F. Herrera, S. Alonso. Group decision making model with incomplete fuzzy preference relations based on additive consistency. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 37(1), 176-189, 2007.
- [10] A.W. Neumann. Recommender Systems for Information Providers: Designing Customer Centric Paths to Information. *Heidelberg: Physica-Verlag*, 2009.
- [11] C. Porcel, A.G. López-Herrera, E. Herrera-Viedma. A recommender system for research resources based on fuzzy linguistic modeling. *Expert Systems with Applications*, 36, 5173-5183, 2009.
- [12] C. Porcel, J.M. Moreno, E. Herrera-Viedma. A Multi-disciplinary Recommender System to Advice Research Resources in University Digital Libraries. *Expert Systems with Applications*, 36(10), 12520-12528, 2009.
- [13] P. Reisnick, H.R. Varian. Recommender Systems. *Special issue of Communications of the ACM*, 40(3), 56-59, 1997.
- [14] L. Ross, P. Sennyey. The Library is Dead, Long Live the Library! The Practice of Academic Librarianship and the Digital Revolution. *The Journal of Academic Librarianship*, 34(2), 145-152, 2008.
- [15] L.A. Zadeh. The concept of a linguistic variable and its applications to approximate reasoning. *Part I. Information Sciences*, 8, 199-249, 1975. *Part II, Information Sciences*, 8, 301-357, 1975. *Part III, Information Sciences*, 9, 43-80, 1975.

ORDERED WEIGHTED AVERAGING APPROACHES FOR AGGREGATING GRADUAL TRUST AND DISTRUST

Chris Cornelis¹ Patricia Victor¹ Enrique Herrera-Viedma²

¹ Dept. of Appl. Math. and Comp. Sci., Ghent University, Belgium, {Chris.Cornelis,Patricia.Victor}@UGent.be

² Dept. of Comp. Sci. and AI, University of Granada, Spain, viedma@decsai.ugr.es

Abstract

In this paper, we focus on the aggregation problem for (trust, distrust) couples in trust networks. In particular, we study approaches based on classical and induced ordered weighted averaging (OWA) operators.

Keywords: aggregation operators, trust networks, recommender systems, ordered weighted average

1 INTRODUCTION

In a (virtual) trust network, agents (nodes) can express their opinion about other agents through trust and distrust statements. In this paper, opinions are represented by means of trust scores, i.e., (trust, distrust) couples drawn from a bilattice [2]. When an agent a needs to establish an opinion about another, unknown agent x , it can inquire about x with one of its own trust relations, who in turn might consult a trust connection, etc., until an agent connected to x is reached. The process of predicting the trust score along the thus constructed path from a to x is called *trust propagation*. Since it often happens that a has not one, but several trust connections that it can consult for an opinion on x , we also require a mechanism for combining several trust scores originating from different sources. This process is called *trust aggregation* and it is very important to generate recommendations in recommender systems [3].

The problem of trust aggregation was first addressed in [1], in which we introduced a number of desirable criteria that a trust score aggregation operator should satisfy. We also presented a possible aggregation approach based on the use of Yager's Ordered Weighted Averaging (OWA, [4]) operators. In this paper, we

undertake a more general study of the use of OWA operators for aggregating trust scores; in particular, we consider approaches based on classical OWA operators, as well as on induced ones [5].

The remainder of this paper is structured as follows: in Section 2, we introduce the necessary preliminaries about OWA operators and the trust score space we use, while in Section 3, we focus on the trust aggregation problem: we first recall a number of properties for trust score operators that were introduced in [1] (Section 3.1), and then in Section 3.2 set out to define a number of procedures based on standard OWA approaches and induced ones. Along the way, it will turn out that some properties need to be sacrificed, or at least adjusted. Finally, we offer a brief conclusion and point out further work in Section 4.

2 PRELIMINARIES

2.1 ORDERED WEIGHTED AVERAGING OPERATORS

The traditional OWA operator [4] models an aggregation process in which a sequence A of n scalar values are ordered decreasingly and then weighted according to their ordered position by means of a weighting vector $W = \langle w_i \rangle$, such that $w_i \in [0, 1]$ and $\sum_i^n w_i = 1$. In particular, if c_i represents the i^{th} largest value in A ,

$$OWA_W(A) = \sum_{i=1}^n w_i c_i \quad (1)$$

The OWA's main strength is its flexibility, since it enables us to model a whole range of aggregation strategies. Moreover, the reordering of the arguments introduces an element of non-linearity into an otherwise linear process.

With the induced OWA operator (IOWA, [5]), the ordering of the arguments is not based on their value, but on that of an induced ordering variable which is

associated to them. In particular, let A and W be defined as above, and let I be a sequence of (not necessarily scalar) values, drawn from a linearly ordered space $(\mathcal{L}, \leq_{\mathcal{L}})$. If c_i represents the value in A associated with the i^{th} largest value in I ,

$$IOWA_W(A, I, \leq_{\mathcal{L}}) = \sum_{i=1}^n w_i c_i \quad (2)$$

Since the values of the induced ordering variable need not be scalars, IOWA operators offer even more flexibility than their standard counterparts.

2.2 TRUST AND DISTRUST: BILATTICE-BASED MODEL

In this paper, a trust network is represented as a directed graph (A, E, R) in which A is the set of agents (nodes), E is the set of trust connections (edges), and R is an $E \rightarrow [0, 1]^2$ mapping that associates to each couple (a, b) of connected agents in E a trust score $R(a, b) = (R^+(a, b), R^-(a, b))$ in $[0, 1]^2$, in which $R^+(a, b)$ is called the trust degree of a in b and $R^-(a, b)$ is called the distrust degree of a in b .

The set of trust scores can be endowed with a bilattice structure. In particular, the trust score space [2]

$$\mathcal{BL}^{\square} = ([0, 1]^2, \leq_t, \leq_k, \neg)$$

consists of the set $[0, 1]^2$ of trust scores, a trust ordering $\leq_t$, a knowledge ordering $\leq_k$, and a negation $\neg$ defined by

$$\begin{aligned} (t_1, d_1) \leq_t (t_2, d_2) & \text{ iff } t_1 \leq t_2 \text{ and } d_1 \geq d_2 \\ (t_1, d_1) \leq_k (t_2, d_2) & \text{ iff } t_1 \leq t_2 \text{ and } d_1 \leq d_2 \\ \neg(t_1, d_1) & = (d_1, t_1) \end{aligned}$$

for all (t_1, d_1) and (t_2, d_2) in $[0, 1]^2$.

The “trust lattice” $([0, 1]^2, \leq_t)$ orders the trust scores going from complete distrust $(0, 1)$ to complete trust $(1, 0)$. The “knowledge” lattice $([0, 1]^2, \leq_k)$ evaluates the amount of available trust evidence, ranging from a “shortage of evidence”, $t + d < 1$ (incomplete information), to an “excess of evidence”, viz. $t + d > 1$ (inconsistent information). The boundary values of the $\leq_k$ ordering, $(0, 0)$ and $(1, 1)$, reflect ignorance, resp. contradiction. In this paper, trust scores (t, d) for which $t + d = 1$ are said to have perfect knowledge, while all others are called knowledge defective.

3 TRUST AGGREGATION

3.1 PROPERTIES

In [1], a number of desirable properties were introduced for a trust score aggregation operator $\Omega : ([0, 1]^2)^n \rightarrow [0, 1]^2$ ($n \geq 1$), which are recalled below:

1. Idempotence property (RQ1).

$$\Omega((t, d), \dots, (t, d)) = (t, d)$$

2. Monotonicity property (RQ2). Ω is monotonously increasing w.r.t. both $\leq_t$ and $\leq_k$, i.e., if $(t_j, d_j) \leq_t (t'_j, d'_j)$, then $(p, q) \leq_t (p', q')$; and if $(t_j, d_j) \leq_k (t'_j, d'_j)$, then $(p, q) \leq_k (p', q')$, with

$$\begin{aligned} \Omega((t_1, d_1), \dots, (t_j, d_j), \dots, (t_n, d_n)) &= (p, q) \\ \Omega((t_1, d_1), \dots, (t'_j, d'_j), \dots, (t_n, d_n)) &= (p', q') \end{aligned}$$

3. Commutativity property (RQ3).

$$\begin{aligned} \Omega((t_1, d_1), \dots, (t_n, d_n)) &= \\ \Omega((t_{\pi_1}, d_{\pi_1}), \dots, (t_{\pi_n}, d_{\pi_n})) \end{aligned}$$

where π is any permutation of $\{1, \dots, n\}$.

4. Neutral element property (RQ4). $(0, 0)$ is neutral for Ω , i.e.,

$$\begin{aligned} \Omega((t_1, d_1), \dots, (t_{n-1}, d_{n-1}), (0, 0)) &= \\ \Omega((t_1, d_1), \dots, (t_{n-1}, d_{n-1})) \end{aligned}$$

5. Opposite arguments property (RQ5). An equal number of $(1, 0)$ and $(0, 1)$ arguments yields contradiction, i.e.,

$$\Omega(\underbrace{(1, 0), \dots, (1, 0)}_{n \text{ times}}, \underbrace{(0, 1), \dots, (0, 1)}_{n \text{ times}}) = (1, 1)$$

3.2 AGGREGATION OPERATORS

The application of an (induced) OWA operator requires scalar values as arguments. As such, OWA operators are not directly applicable to aggregate trust scores. Therefore, we propose to perform trust aggregation by means of two separate OWA operators, one for trust and one for distrust.

3.2.1 Standard OWA approaches

Below, we describe a generic procedure for applying (standard) OWA operators to the trust score aggregation problem:

1. Determine n , the number of trust score arguments distinct from $(0, 0)$. Trust scores that represent complete ignorance do not take part in the aggregation process.¹
2. Construct the sequences T and D , containing the n trust values (resp., the n distrust values) of the trust score arguments.

¹If all trust scores equal $(0, 0)$, the final result is also set to $(0, 0)$ and the aggregation process terminates at this step.

3. Construct n -dimensional weight vectors W_T and W_D ; weights may or may not be dependent on the actual trust score arguments.
4. Compute the aggregated trust score as $(OWA_{W_T}(T), OWA_{W_D}(D))$.

As it is clear from the above, the actual way of aggregating the trust scores is determined by the choice of the weight vectors. One strategy is to construct W_T and W_D beforehand. For instance, the final trust (resp., distrust) value can be evaluated as the extent to which a predefined fraction (at least one, all of them, a majority, ...) of the trust score arguments exhibits trust (resp., distrust).

Example 1 (Fixed weights) In [1], given n trust score arguments to aggregate (all distinct from $(0,0)$), trust and distrust weights are computed by $(i = 1, \dots, n)$

$$W_{T_i} = \frac{2 \cdot \max(0, \lceil \frac{n}{2} \rceil - i + 1)}{\lceil \frac{n}{2} \rceil (\lceil \frac{n}{2} \rceil + 1)} \quad (3)$$

$$W_{D_i} = \frac{2 \cdot \max(0, \lceil \frac{n}{4} \rceil - i + 1)}{\lceil \frac{n}{4} \rceil (\lceil \frac{n}{4} \rceil + 1)} \quad (4)$$

The disparity between trust and distrust weights was motivated by the observation that a few distrust statements about x (in particular, a quarter of them) may suffice to reach a final conclusion of distrust, while the evaluation of trust depends on the majority of the arguments. Note that weights are decreasing, in a sense that the higher trust/distrust values have a stronger impact than the lower ones.

It can be verified, by construction and by the properties of an OWA operator, that **(RQ1)**, **(RQ3)** and **(RQ4)** always hold for this type of aggregation (in fact, regardless of the fact whether weights are fixed or not). In order for **(RQ5)** to hold, it suffices that $W_{T_i} = 0$ and $W_{D_i} = 0$ as soon as $i > \frac{n}{2}$. The proposal in Ex. 1 fulfills this condition. **(RQ2)** is a harder condition to fulfill in general, as the following example shows.

Example 2 If the trust scores to aggregate are $(1,0)$, $(0,0)$, $(0,0)$ and $(0,0)$, then the outcome by our OWA procedure is $(1,0)$ (regardless of the choice of weights vectors, because $n = 1$). If we change these trust scores to $(1,0)$, $(0.1,0)$, $(0.1,0)$ and $(0.1,0)$, the number of arguments that take part in the OWA aggregation equals 4; computing the weights as in Ex. 1, i.e., $W_T = (\frac{2}{3}, \frac{1}{3}, 0, 0)$ and $W_D = (1, 0, 0, 0)$, the final result of the aggregation equals $(0.7, 0)$. So, although $(0,0) \leq_t (0.1,0)$ and $(0,0) \leq_k (0.1,0)$, $(1,0) \not\leq_t (0.7,0)$ and $(1,0) \not\leq_k (0.7,0)$.

The reason for the failure of **(RQ2)** in this example is due to the presence (and subsequent alteration) of $(0,0)$ trust score arguments, which causes the application of the OWA operators to a different number of arguments. It can be verified, however, that if we add the restriction to **(RQ2)** that $(t_j, d_j) \neq (0,0)$ and $(t'_j, d'_j) \neq (0,0)$, the property holds, regardless of the weight vectors W_T and W_D , provided they remain fixed.

According to this analysis, fixed-weight OWA approaches perform well w.r.t. the criteria set out in Section 3.1. However, they also exhibit certain drawbacks, as the following example illustrates.

Example 3 Assume the trust scores to aggregate are $(1,0)$, $(\delta,0)$, $(\delta,0)$ and $(\delta,0)$, with δ a value close to 0. In other words, three of the trust score arguments are very close to ignorance. Intuitively, one would expect their contribution to the final result to be very small. However, using the same weighting vector as in Ex. 1, the aggregated value will be $(\frac{2}{3} + \frac{1}{3}\delta, 0) \approx (\frac{2}{3}, 0)$, which differs significantly from $(1,0)$, the result obtained if the $(\delta,0)$ values are replaced by $(0,0)$.

In fact, the above kind of problem occurs with any fixed-weight approach²; it is due to the fact that in this approach, trust scores are not discriminated w.r.t. the amount of knowledge they contain. Intuitively, one can argue that the closer a trust score (t,d) is to ignorance w.r.t. the $\leq_k$ order (i.e., $t+d \approx 0$), the lower its associated weight should be. On the other hand, it also makes sense to penalize trust scores (t,d) which are highly inconsistent (i.e., $t+d \approx 2$), to avoid that such “defective” trust scores influence the aggregation result in a disproportionate way (since both their trust and distrust values are high). In general, the “knowledge defect” of a trust score (t,d) can be expressed by the following formula:

$$kd(t,d) = 1 - |t + d - 1| \quad (5)$$

The following example illustrates a possible way to alter the weights based on the “knowledge defect” exhibited by the individual trust score arguments.

Example 4 (Knowledge-dependent weights) Given n trust score arguments (t_i, d_i) ($i = 1, \dots, n$, all trust scores distinct from $(0,0)$), we can associate with them a weight vector W^{kd} that represents each trust score’s degree of knowledge defect relative to the

²The only exception is when $W_{t_1} = W_{d_1} = 1$, i.e., only the highest trust and distrust values are taken into account.

remaining trust scores³:

$$W_i^{kd} = \frac{kd(t_i, d_i)}{\sum_{j=1}^n kd(t_j, d_j)} \quad (6)$$

We cannot use W^{kd} directly as a weight vector inside the OWA operators, since the knowledge defect weights are not associated to ordered positions, but rather to the arguments themselves. We can however use them to modify existing OWA weight vectors W^T and W^D (which can be chosen as in the fixed weight approach). The final OWA weight vectors W_T and W_D are obtained as ($i = 1, \dots, n$)

$$W_{T_i} = \frac{W_i^T W_{\pi_i}^{kd}}{\sum_{j=1}^n W_j^T W_{\pi_j}^{kd}}, W_{D_i} = \frac{W_i^D W_{\pi'_i}^{kd}}{\sum_{j=1}^n W_j^D W_{\pi'_j}^{kd}} \quad (7)$$

in which π, π' represent the permutations that map an ordered position i to the index of the trust score that appears at that position.

To illustrate the operation of this approach, we apply it to the data of Ex. 3. In this case,

$$W^{kd} = \left(\frac{1}{1+3\delta}, \frac{\delta}{1+3\delta}, \frac{\delta}{1+3\delta}, \frac{\delta}{1+3\delta} \right) \quad (8)$$

$$W_T = \left(\frac{2}{2+\delta}, \frac{\delta}{2+\delta}, 0, 0 \right) \quad (9)$$

The final aggregation result will be $\left(\frac{2+\delta^2}{2+\delta}, 0 \right) \approx (1, 0)$, which corresponds to our intuition.

The application of knowledge-dependent OWA weights does not affect the properties (RQ1) and (RQ3)–(RQ5), but (RQ2) cannot be maintained, not even in the weakened version which holds for fixed weights.

Example 5 Consider the following trust score sequences:

$$\begin{aligned} A &= \langle (1, 0), (0.9, 0.2), (0, 1) \rangle \\ B &= \langle (1, 0), (0.9, 0.1), (0, 1) \rangle \\ C &= \langle (1, 0.9), (0.9, 0.2), (0, 1) \rangle \end{aligned}$$

Constructing the weight vectors as in Ex. 4, we obtain the aggregated trust scores $\left(\frac{281}{290}, 1 \right)$ for A , $\left(\frac{29}{30}, 1 \right)$ for B and $\left(\frac{101}{110}, 1 \right)$ for C . However, while $(0.9, 0.2) \leq_t (0.9, 0.1)$, $\left(\frac{281}{290}, 1 \right) \not\leq_t \left(\frac{29}{30}, 1 \right)$ (comparing sequence A with B). Similarly, while $(1, 0) \leq_k (1, 0.9)$, $\left(\frac{281}{290}, 1 \right) \not\leq_k \left(\frac{101}{110}, 1 \right)$ (comparing sequence A with C).

³We assume that not all trust scores are equal to $(1, 1)$.

In this case, the failure of (RQ2) is due to the change of the knowledge-dependent weight vector W^{kd} . While it might be perceived as a disadvantage of this particular approach, it can also be argued that any attempt to penalize trust scores for their knowledge defects is incompatible with maintaining monotonicity. In fact, this is already evident in the fixed-weight approach: to guarantee that $(0, 0)$ can play its role as the neutral element of Ω , it needs to be handled separately.

A weakened version of trust monotonicity⁴, however, can be obtained as follows:

Weak trust monotonicity property (RQ2'). If $(t_j, d_j) \leq_t (t'_j, d'_j)$ and $t_j + d_j = t'_j + d'_j$, then $(p, q) \leq_t (p', q')$, with

$$\begin{aligned} \Omega((t_1, d_1), \dots, (t_j, d_j), \dots, (t_n, d_n)) &= (p, q) \\ \Omega((t_1, d_1), \dots, (t'_j, d'_j), \dots, (t_n, d_n)) &= (p', q') \end{aligned}$$

As a corollary of this property, trust monotonicity holds in particular when all the involved trust scores (t, d) have perfect knowledge ($t + d = 1$).

3.2.2 Induced OWA approaches

In this section, we consider an alternative approach, based on IOWA operators, to take into account the knowledge defect exhibited by trust score arguments. As stated before, in order to use the IOWA approach, we require an order inducing variable that takes values drawn from a linearly ordered space. In this section, we will consider the trust scores themselves for this purpose. Since the bilattice orderings $\leq_t$ and $\leq_k$ are only partial rather than linear, they do not qualify to construct the required linearly ordered space. However, meaningful linear orderings over trust scores do exist. In particular, if we define, for $(t_1, d_1), (t_2, d_2)$ in $[0, 1]^2$,

$$\begin{aligned} (t_1, d_1) \leq_{kd}^t (t_2, d_2) \quad \text{iff} \quad & (kd(t_1, d_1) < kd(t_2, d_2)) \vee \\ & (kd(t_1, d_1) = kd(t_2, d_2) \\ & \wedge t_1 \leq t_2) \end{aligned} \quad (10)$$

$$\begin{aligned} (t_1, d_1) \leq_{kd}^d (t_2, d_2) \quad \text{iff} \quad & (kd(t_1, d_1) < kd(t_2, d_2)) \vee \\ & (kd(t_1, d_1) = kd(t_2, d_2) \\ & \wedge d_1 \leq d_2) \end{aligned} \quad (11)$$

it can be verified that $([0, 1]^2, \leq_{kd}^t)$ and $([0, 1]^2, \leq_{kd}^d)$ are linearly ordered spaces. For both of them, the smallest element is $(0, 0)$, while $(1, 0)$ is the largest element for $\leq_{kd}^t$ and $(0, 1)$ the largest for $\leq_{kd}^d$.

The corresponding IOWA aggregation procedure is largely analogous to that for standard OWA:

⁴Note that a similar property for knowledge monotonicity holds only in the trivial case where $(t_j, d_j) = (t'_j, d'_j)$, due to the condition $t_j + d_j = t'_j + d'_j$.

1. Determine n , the number of trust score arguments distinct from $(0, 0)$. Trust scores that represent complete ignorance do not take part in the aggregation process.⁵
2. Construct the sequences I , T and D ; I contains the n trust scores, while T and D contain the corresponding n trust values (resp., the n distrust values).
3. Construct n -dimensional weight vectors W_T and W_D ; weights may or may not be dependent on the actual trust score arguments.
4. Compute the aggregated trust score as $(IOWA_{W_T}(T, I, \leq_{kd}^t), IOWA_{W_D}(D, I, \leq_{kd}^d))$.

Note that while the IOWA approach allows to take into account knowledge defects by using the trust score orders $\leq_{kd}^t$ and $\leq_{kd}^d$, it still makes sense to use knowledge-dependent weights. This becomes evident when we apply the approach to the data in Ex. 3, which gives the same outcome as in the standard OWA case. On the other hand, the monotonicity property (**RQ2**) is not guaranteed for the IOWA approach, even if fixed weights are used. The following example illustrates this.

Example 6 Consider the following trust score sequences:

$$\begin{aligned} A &= \langle (0.95, 0.05), (0.8, 0.2), (0, 0.5) \rangle \\ B &= \langle (1, 0.05), (0.8, 0.2), (0, 0.5) \rangle \end{aligned}$$

For sequence A it holds that $(0.95, 0.05) \geq_{kd}^t (0.8, 0.2) \geq_{kd}^t (0, 0.5)$ and $(0.8, 0.2) \geq_{kd}^d (0.95, 0.05) \geq_{kd}^d (0, 0.5)$, while for sequence B , $(0.8, 0.2) \geq_{kd}^t (1, 0.05) \geq_{kd}^t (0, 0.5)$ and $(0.8, 0.2) \geq_{kd}^d (1, 0.05) \geq_{kd}^d (0, 0.5)$. Constructing the weight vectors as in Ex. 1, we obtain the aggregated trust scores $(0.9, 0.8)$ for A and $(\frac{13}{15}, 0.8)$ for B . However, while $(0.95, 0.05) \leq_t (1, 0.05)$ and $(0.95, 0.05) \leq_k (1, 0.05)$, $(0.9, 0.8) \not\leq_t (\frac{13}{15}, 0.8)$ and $(0.9, 0.8) \not\leq_k (\frac{13}{15}, 0.8)$.

The IOWA approach does satisfy the weak trust monotonicity property (**RQ2'**), and this regardless of whether fixed or variable weight vectors are used.

4 CONCLUSION AND FUTURE WORK

In this paper, we have shown how ordered weighted averaging operators (OWA) can be used to solve the

⁵If all trust scores equal $(0, 0)$, the final result is also set to $(0, 0)$ and the aggregation process terminates at this step.

trust aggregation problem. In particular, we studied approaches where aggregation of trust scores is performed by applying a pair of standard or induced OWA operators to their trust and distrust components of the trust score separately.

We have also introduced two different ways to take into account that some trust scores are more useful than others in deriving a final aggregated trust score; specifically, the knowledge defect of a trust score can be incorporated both into the weight vector, as well as into the induced order used in an IOWA approach.

We also observed that taking into account knowledge defects is incompatible with maintaining the monotonicity of the trust and knowledge orders of the trust score bilattice; instead, we proposed a weaker property of trust monotonicity for trust scores that have the same level of knowledge defect, which holds for all the introduced approaches.

The choice as to which approach (OWA or IOWA, weight generating strategy, ...) is most suitable also depends on the application at hand. In our research, we are focusing in particular on recommender systems, i.e., applications which suggest items to users who might be interested in them. Trust information can help to establish more, and more accurate, recommendations, and its incorporation into existing recommender system technology is a topic of ongoing research (see e.g. [3]).

As another part of our future work, we also aim to refine the aggregation strategy to take into account certain aspects of the virtual trust network's topology. In particular, the current approach is indifferent as to the length of the paths that generated the individual trust scores, and also does not consider how many times the same agent appears on a path.

Acknowledgements

Chris Cornelis would like to thank the Research Foundation—Flanders for funding his research. Patricia Victor would like to thank the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen) for funding her research. Enrique Herrera-Viedma would like to thank the financing of andalucian excellence project TIC05299, Feder Funds in FUZZYLING project (TIN2007-61079) and PETRI project (PET 2007-0460).

References

- [1] P. Victor, C. Cornelis, M. De Cock, E. Herrera-Viedma. Aggregation of Gradual Trust and Distrust. *Proceedings of Eurofuse Workshop on Pref-*

- erence Modeling and Decision Analysis (Eurofuse 2009)*, Pág.259-264, 2009.
- [2] P. Victor, C. Cornelis, M. De Cock, P. Pinheiro Da Silva. Gradual Trust and Distrust in Recommender Systems. *Fuzzy Sets and Systems* 160(10), Pág.1367-1382, 2009.
 - [3] P. Victor, C. Cornelis, M. De Cock, A. Teredesai. Trust- and Distrust-Based Recommendations for Controversial Reviews. *IEEE Intelligent Systems*, in press, 2009.
 - [4] R. R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision making. *IEEE Transactions on Systems, Man, and Cybernetics* 18, Pág.183-190, 1988.
 - [5] R. R. Yager, D. Filev. Induced Ordered Weighted Averaging Operators. *IEEE Transactions on Systems, Man, and Cybernetics* 29(2), Pág.141-150, 1999.

A DISTANCE FUNCTION TO ASSESS THE SIMILARITY OF WORDS USING ONTOLOGIES

Montserrat Batet¹ Aida Valls¹ Karina Gibert²

¹ Universitat Rovira i Virgili, Department of Computer Science and Mathematics
Av. Països Catalans 26, 43007 Tarragona, Spain
{montserrat.batet,aida.valls}@urv.cat

² Universitat Politècnica de Catalunya, Department of Statistics and Operations Research
Campus Nord, Ed.C5, c/Jordi Girona 1-3, 08034 Barcelona, Spain
karina.gibert@upc.edu

Abstract

When comparing categorical values, traditional approaches use metrics based only on the matching of the values, obtaining a Boolean result. In this paper, it is proposed to use a measure able to compute the degree of semantic similarity between a pair of terms using an ontology as background knowledge. The presented measure - the Superconcept-based distance - have two main advantages over other approaches based on the exploitation of the hierarchical model of ontologies: on one hand, it takes into account the whole hierarchy of concepts in the ontology to assess the similarity between a pair of words; on the other hand, this paper proves that this measure fulfills the distance properties. As the paper also reviews, the usual semantic similarity measures used in the literature does not fulfill the triangle inequality, which prevents them from being used in some decision making methods.

Keywords: Semantic similarity, ontologies, linguistic terms, distance properties.

1 INTRODUCTION

Our work is focused on decision making problems that have to deal with variables that take their values in a list of linguistic terms. Differently from categorical variables that have a predefined domain of terms (i.e. modalities), we are facing the case of having a non fixed and large set of possible values. Moreover, no ordering or measurement scale in the values is defined, as traditional linguistic variables do. An example of this type of variables can be languages or hobbies.

Traditionally, the comparison between two values in categorical variables is done simply based on the equality/inequality of the words, due to the lack of proper methods for representing the meaning of the terms. Some widely used distance measures for categorical values are the Chi-Squared and the Hamming distance [16]. However, from a semantic point of view, it is possible to establish different degrees of similarity between values (i.e. Italian is more similar to French than to Chinese). Each of these terms is in fact describing a concept, thus, reasoning at a conceptual level should be done in order to calculate an approximation of the similarity.

The computation of the semantic similarity between concepts is an active trend in computational linguistics. The similarity between a pair of concepts quantifies how they are alike based on the estimation of semantic evidence observed in some knowledge source. Taxonomies and, more generally ontologies [13] are considered as a graph model in which semantic relations are modeled as links between concepts. In the literature several semantic similarity measures have been proposed. A brief review of those measures will be presented in section 2. In this paper we focus on measures based on the exploitation of the taxonomic relations in ontologies. These measures are based on the computation of the minimum path length between a pair of concepts to assess the semantic similarity. Consequently, a lot of relevant information is missed because the rest of taxonomic information is not considered.

To sort out this problem, in this paper a new measure is presented, which takes into account the common and not common ancestors (i.e. superconcepts) of the two concepts compared. It is called Superconcept-based Distance (SCD). When this comparison between terms is done in the context of decision making, it is worth to know the metrical properties of the measure, because it may have implications on the results that will be obtained. For example, for the particular case of hierarchical clustering, it is interesting to maintain

the ultrametric properties of the dendograms and the Huygens theorem of decomposition of inertia, which are directly related with the interpretability of the final results. If the comparison measure is a distance, these properties are guaranteed [3]. However, in the field of computational linguistics, the comparison measures proposed usually do not fulfill the triangle inequality property, being only similarities but not distances. In this field, this is not a handicap. As noted by Tversky [14], the triangle inequality property is not always achievable when dealing with linguistic descriptions.

The goal of this paper is to study the Superconcept-based measure and analyze its metrical properties, proving that it is a distance. Since most of the classical semantic similarity measures are not distances, this is an interesting result that may help to extend the use of this similarity assessment not only to text mining but also to other fields, specially to intelligent decision support systems.

The paper is organized as follows. Section 2 reviews the existing semantic similarity measures based on ontologies. Section 3 defines the Superconcept-based Distance. Section 4 shows the good performance results of this measure. Section 5 proves the distance properties of SCD. Discussion and future work are in section 6.

2 RELATED WORK

In the literature, we can find several different approaches to compute semantic similarity between concepts according to the techniques employed and the knowledge exploited. There are approaches in which the degree of co-occurrence between terms in a given corpus is used as an estimation of similarity [7]. These measures need a corpus as general as possible in order to estimate social-scale word usage. However, due to the lack of domain coverage of a general domain corpus and the difficulty of compiling a relevant domain corpus big enough to obtain robust statistics, they offer a limited performance.

Similarity computation can be also based on structured representations of knowledge, typically, subsumption hierarchies. The evolution of those hierarchies have given the origin to ontologies in which many types of relationships and logical descriptions can be specified to formalize knowledge [9].

Some approaches combine the knowledge provided by an ontology with the Information Content (IC) of the concepts that are being compared. IC measures the amount of information provided by a given term from its probability of appearance in a corpus. Based on this premise, Resnik [11] proposed to estimate the sim-

ilarity between two terms as the amount of information they share in common. In a taxonomy, this information is represented by the Least Common Subsumer (LCS) of both terms. So, the computation of the IC of the LCS results is used to estimate the similarity of the subsumed terms. The more specific the subsumer is (higher IC), the more similar the subsumed terms are. Several variations of this measure have been developed [5]. However, those measures are affected by the availability of the background corpus and their coverage with respect to the evaluated terms.

Other approaches consider ontologies as a graph model in which semantic interrelations are modeled as links between concepts. These measures compute concept similarity as inter-link distance in the taxonomic structure (also called Path Length) [10].

$$sim_{pL}(c_1, c_2) = \min \# \text{ of is-a edges connecting } c_1 \text{ and } c_2 \quad (1)$$

Several variations of this measure have been developed such as the one proposed by Wu and Palmer [15]. Their proposal also takes into account the depth of the concepts in the hierarchy (2). Here, N_1 and N_2 is the number of is-a links from c_1 and c_2 respectively to the LCS c , and N_3 is the number of is-a links from c to the root ρ of the ontology.

$$sim_{w\&p}(c_1, c_2) = \frac{2 * N_3}{N_1 + N_2 + 2 * N_3} \quad (2)$$

Leacock and Chodorow [6] proposed a measure that considers both the number of nodes N_p from c_1 to c_2 , and the depth D of the taxonomy in which they occur (3).

$$sim_{l\&c}(c_1, c_2) = -\log N_p / 2D \quad (3)$$

However, those measures based on Path Length only consider the minimum path between a pair of concepts, omitting the rest of the taxonomic knowledge available in the ontology, wasting a great amount of relevant knowledge.

3 SUPERCONCEPT BASED DISTANCE

The Superconcept-based Distance (SBC) has been designed to overcome the limitations of the other Path Length approaches, explained in the previous section. This measure takes into account both the amount of overlapping and non-overlapping taxonomical knowledge between concepts. To do this, the distance considers the number of shared superconcepts (upper concepts of a concept in the taxonomy) and the number of non shared superconcepts.

Considering that an ontology O as an object model composed by a set of concepts or classes C , which are

taxonomically related by the transitive is-a relation $H^c \in C \times C$, called concept hierarchy or taxonomy, and *non-taxonomically* related by named object relations $R^* \in C \times C \times \text{String}$, the Superconcept-based Distance is defined as:

Definition: Superconcept-based Distance (SCD)

$$d_{SCD}(c_i, c_j) = \sqrt{\frac{|\mathcal{A}(c_i) \cup \mathcal{A}(c_j)| - |\mathcal{A}(c_i) \cap \mathcal{A}(c_j)|}{|\mathcal{A}(c_i) \cup \mathcal{A}(c_j)|}} \quad (4)$$

where $\mathcal{A}(c_i) = \{c_j \in C | c_j \text{ is superconcept of } c_i \in H^C \vee c_i = c_j\}$.

The numerator of expression (4) calculates the number of non-common ancestors of the two concepts, c_i and c_j . That is, the amount of non-shared information. The denominator is used to consider the proportion of non-common superconcepts with respect to the total number of superconcepts. In this way, the measure is able to distinguish between cases in which the number of common superconcepts between a pair of concepts is small from those cases in which the number of common superconcepts is high. Less shared superconcepts means a greater distance. The squared root is used to smooth the result as the amount of common information is not linear in relation to similarity [7].

4 EVALUATION

The most common way of evaluating similarity measures is by using a set of word pairs whose similarity has been assessed by a group of human experts and calculate the correlation between the human ratings and the results of the computerized measures. The most commonly used benchmarks are the one proposed by Miller and Charles [8], which is a set of 30 domain-independent word pairs chosen from their high, middle and low level of synonymy from an experiment done by Rubenstein and Goodenough [12], and the one proposed by Resnik [11]. Resnik replicated the experiment of Miller and Charles in order to obtain more accurate ratings by giving a group of experts the same set of noun pairs. Resnik computed how well the rating of the subjects in his evaluation correlated with Miller and Charles ratings. The average correlation over the 10 subjects (considered by Resnik) was 0.884. This value is considered the upper bound to what one could expect from a machine computation on the same task [11]. In order to evaluate the measure presented, we used these benchmarks and WordNet[4] as background ontology (Table 1).

The results show that Leacock & Chodorow and Wu & Palmer approaches clearly outperform Path Length measure. The reason is that these measures take into account more information of the taxonomy as the

Table 1: Correlations obtained for each measure using domain independent benchmarks

Measure	Resnik	M&C	Rubenstein
Path Length	0.670	0.638	0.737
Wu&Palmer	0.804	0.795	0.801
Leacock&Chodorow	0.829	0.779	0.838
SCD	0.839	0.807	0.839

depth of the ontology (Leacock and Chodorow's proposal) or the the depth of the LCS and the relative depth between compared concepts and the LCS (Wu and Palmer). Our measure (SCD) considers the whole set of superconcepts of the pair of compared concepts. It can be seen that our proposal provides the highest correlation regarding expert's judgments. Notice that for the Resnik benchmark, with SCD measure we are approaching the upper bound (0.884) to what one could expect from a machine computation.

The distance had been also evaluated for the biomedical domain using a benchmark referring to medical disorders and SNOMED-CT terminology [1]. In this study, SCD also obtained the best performance results.

5 PROOF OF THE DISTANCE PROPERTIES

The properties that a distance measure must fulfill are the following ones:

1. Identity: $d(c_i, c_j) = 0 \iff c_i = c_j, \forall i, j$
2. Symmetry: $d(c_i, c_j) = d(c_j, c_i), \forall i, j$
3. Triangle inequality: $d(c_i, c_j) + d(c_j, c_k) \geq d(c_i, c_k), \forall i, j, k$

The proofs of the Identity and Symmetry are straightforward. The Identity is fulfilled because iff two concepts are the same, the union of all the ancestors is equal to the set of common ancestors (i.e. intersection), which leads to a 0 at the numerator of eq. 4. Symmetry also holds because the union and intersection operators are symmetric.

So, in this section we concentrate on the proof of the the triangle inequality property. First some notation is introduced to simplify the formulation of the problem.

Let:

$$A = \mathcal{A}(c_1), B = \mathcal{A}(c_2), C = \mathcal{A}(c_3)$$

Then, the triangle inequality property is expressed as:

$$\frac{|\mathcal{A} \cup B| - |\mathcal{A} \cap B|}{|\mathcal{A} \cup B|} + \frac{|\mathcal{B} \cup C| - |\mathcal{B} \cap C|}{|\mathcal{B} \cup C|} \geq \frac{|\mathcal{A} \cup C| - |\mathcal{A} \cap C|}{|\mathcal{A} \cup C|} \quad (5)$$

Proposition 1. In a taxonomic relation without multiple inheritance, this property holds:

$$A \cap B = B \cap C \text{ or } A \cap B = A \cap C \text{ or } B \cap C = A \cap C$$

Proof. Assuming that proposition 1 is not true, the following conditions must be true simultaneously:

1. $A \cap B \neq B \cap C \Rightarrow$ (1.a) $\exists c_x \in A \cap B, c_x \notin B \cap C$
or (1.b) $\exists c_x \notin A \cap B, c_x \in B \cap C$
2. $A \cap B \neq A \cap C \Rightarrow$ (2.a) $\exists c_y \in A \cap B, c_y \notin A \cap C$
or (2.b) $\exists c_y \notin A \cap B, c_y \in A \cap C$
3. $B \cap C \neq A \cap C \Rightarrow$ (3.a) $\exists c_z \in A \cap C, c_z \notin B \cap C$
or (3.b) $\exists c_z \notin A \cap C, c_z \in B \cap C$

These six possible cases are graphically represented in figure 1.

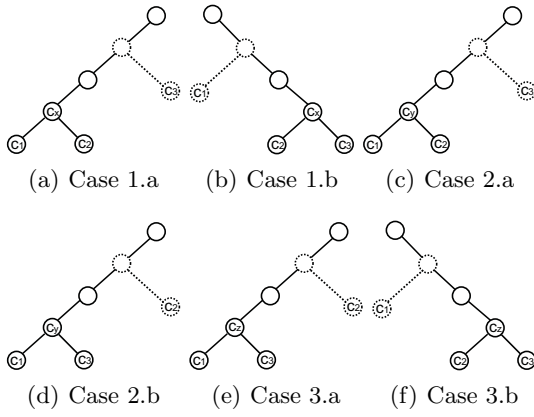


Figure 1: Intersection cases

From these figures, it is easy to see that:

If expr. 1.a = true $\Rightarrow B \cap C = A \cap C$ (Fig. 1(a))

If expr. 1.b = true $\Rightarrow A \cap B = A \cap C$ (Fig. 1(b))

If expr. 2.a = true $\Rightarrow B \cap C = A \cap C$ (Fig. 1(c))

If expr. 2.b = true $\Rightarrow A \cap B = B \cap C$ (Fig. 1(d))

If expr. 3.a = true $\Rightarrow A \cap B = B \cap C$ (Fig. 1(e))

If expr. 3.b = true $\Rightarrow A \cap B = A \cap C$ (Fig. 1(f))

□

Proof. Triangle inequality.

For simplicity, let us rename the sets of union and intersection as follows:

$$x = |A \cup B| \text{ and } p = |A \cap B| \text{ with } 1 \leq p \leq x$$

$$y = |B \cup C| \text{ and } q = |B \cap C| \text{ with } 1 \leq q \leq y$$

$$z = |A \cup C| \text{ and } r = |A \cap C| \text{ with } 1 \leq r \leq z$$

Notice that $p = 1$ iff $A \cap B = \text{root node of the ontology}$, $q = 1$ iff $B \cap C = \text{root node of the ontology}$ and $r = 1$ iff $A \cap C = \text{root node of the ontology}$. In addition, $p = x$ iff $c_1 = c_2$, $q = y$ iff $c_2 = c_3$ and $r = z$ iff $c_1 = c_3$.

So the triangle inequality can be expressed as:

$$\sqrt{\frac{x-p}{x}} + \sqrt{\frac{y-q}{y}} \geq \sqrt{\frac{z-r}{z}}$$

, that is:

$$\frac{\sqrt{(x-p)yz}}{\sqrt{xyz}} + \frac{\sqrt{(y-q)xz}}{\sqrt{xyz}} \geq \frac{\sqrt{(z-r)xy}}{\sqrt{xyz}}$$

This is equivalent to demonstrate:

$$\sqrt{xyz - pyz} + \sqrt{xyz - xqz} \geq \sqrt{xyz - xyr} \quad (6)$$

Let us prove eq.6 in the three cases of Proposition 1.

1. If $A \cap B = B \cap C \Rightarrow p = q$ and $p \leq r$

$$\sqrt{xyz - pyz} + \sqrt{xyz - xpz} \geq \sqrt{xyz - xyr}$$

$$(\sqrt{xyz - pyz} + \sqrt{xyz - xpz})^2 \geq (\sqrt{xyz - xyr})^2$$

$$xyz + xyr - pz(x+y) + 2\sqrt{xyz - pyz}\sqrt{xyz - xpz} \geq 0$$

In this case we can rewrite x, y and z as:

$$x = a + b + p$$

$$y = b + c + p$$

$$z = a - (r - p) + c - (r - p) + r$$

$$= a + 2p + c - r$$

,where $a = |A| - p$, $b = |B| - p$, and $c = |C| - p$. So, a , b , c are the number of superconcepts of A , B , C , respectively, that not belong to $A \cap B$. Then,

$$\begin{aligned} & a^2b + 4abp + 2abc - abr + a^2c + 4acp + ac^2 \\ & - acr + a^2p + 3ap^2 - apr + b^2a + 2b^2p + b^2c \\ & - b^2r + 4bcp + bc^2 - bcr + 4bp^2 - 2bpr + 3cp^2 \\ & + c^2p - cpr + 2p^3 - p^2r + abr + acr + apr \\ & + b^2r + bcr + 2bpr + cpr + p^2r - a^2p - 2abp \\ & - 4ap^2 - 2acp - 4bp^2 - 4p^3 - 4cp^2 - 2bcp \\ & - c^2p + apr + 2bpr + 2p^2r + cpr \\ & + 2\sqrt{xyz - pyz}\sqrt{xyz - xpz} \geq 0 \end{aligned}$$

Having that $p \leq r$ (the number of concepts of $A \cap B$ is less or equal than the number of concepts of $A \cap C$) and $p \leq x$ (the number of concepts of $A \cap B$ is less or equal than the number of concepts of $A \cup B$) and $p \leq y$ (the number of concepts of

$B \cap C$ is less than number of concepts of $B \cup C$), we can determine the positivity of the pairs:

$$\begin{aligned} & a^2b + 2abp + 2abc + a^2c + 2acp + ac^2 \\ & + b^2a + 2b^2p + b^2c + 2bcp + bc^2 \\ & + \underbrace{2p^2r - 2p^3}_{\geq 0} + \underbrace{apr - ap^2}_{\geq 0} + \underbrace{cpr - cp^2}_{\geq 0} + 2bpr \\ & + 2\sqrt{\underbrace{xyz - pyz}_{\geq 0}}\sqrt{\underbrace{xyz - xpz}_{\geq 0}} \geq 0 \end{aligned}$$

So, triangle inequality holds when $A \cap B = B \cap C$.

2. If $A \cap B = A \cap C \Rightarrow p = r$ and $p \leq q$

$$\begin{aligned} & \sqrt{xyz - pyz} + \sqrt{xyz - xqz} \geq \sqrt{xyz - xyp} \\ & xyz + yp(x - z) - xqz + 2\sqrt{xyz - pyz}\sqrt{xyz - xqz} \geq 0 \end{aligned}$$

In this case we can rewrite x, y and z as:

$$\begin{aligned} x &= a + b + p \\ y &= b - (q - p) + c - (q - p) + q \\ &= b + 2p + c - q \\ z &= a + c + p \end{aligned}$$

,where $a = |A| - p$, $b = |B| - p$, and $c = |C| - p$. Then,

$$\begin{aligned} & a^2b + 2a^2p + a^2c - a^2q + 2abc + 4acp \\ & + ac^2 - acq + 4abp + 4ap^2 - 2apq + ab^2 \\ & - abq + b^2c + 4bcp + bc^2 - bcq + b^2p + 3bp^2 \\ & - bpq + 3cp^2 + c^2p - cpq + 2p^3 - p^2q + b^2p \\ & + 2bp^2 - 2cp^2 - c^2p - bpq + cpq - a^2q - acq \\ & - 2apq - abq - bcq - bpq - cpq - p^2q \\ & + 2\sqrt{xyz - pyz}\sqrt{xyz - xqz} \geq 0 \end{aligned}$$

In this case, the number of superconcepts of c_2 and c_3 are greater or equal than $|B \cap C|$, so we have that $q \leq c + p$ and $q \leq b + p$. In addition, we have that $p \leq x$ and $q \leq y$. So:

$$\begin{aligned} & \underbrace{a^2p + a^2b - a^2q}_{\geq 0} + \underbrace{a^2p + a^2c - a^2q}_{\geq 0} \\ & + \underbrace{acp + ac^2 - acq}_{\geq 0} + \underbrace{acp + abc - acq}_{\geq 0} \\ & + \underbrace{2ap^2 + 2acp - 2apq}_{\geq 0} + \underbrace{2ap^2 + 2abp - 2apq}_{\geq 0} \\ & + \underbrace{abp + ab^2 - abq}_{\geq 0} + \underbrace{abp + abc - abq}_{\geq 0} \\ & + \underbrace{bcp + b^2c - bcq}_{\geq 0} + \underbrace{bcp + bc^2 - bcq}_{\geq 0} \\ & + \underbrace{2bp^2 + 2b^2p - 2bpq}_{\geq 0} + \underbrace{bp^2 + bcp - bpq}_{\geq 0} \\ & + \underbrace{2p^3 + 2bp^2 - 2p^2q}_{\geq 0} + \underbrace{cp^2 + bcp^2 - cpq}_{\geq 0} \end{aligned}$$

$$+ 2\sqrt{\underbrace{xyz - pyz}_{\geq 0}}\sqrt{\underbrace{xyz - xqz}_{\geq 0}} \geq 0$$

So, the triangle inequality is demonstrated when $A \cap B = A \cap C$.

3. If $B \cap C = A \cap C \Rightarrow q = r$ and $q \leq p$

$$\begin{aligned} & \sqrt{xyz - pyz} + \sqrt{xyz - xqz} \geq \sqrt{xyz - xyp} \\ & xyz + xq(y - z) - pyz + 2\sqrt{xyz - pyz}\sqrt{xyz - xqz} \geq 0 \end{aligned}$$

In this case we can rewrite x, y and z as:

$$\begin{aligned} x &= a - (p - q) + b - (p - q) + p = \\ &= a + b + 2q - p \\ y &= b + c + q \\ z &= a + c + q \end{aligned}$$

,where $a = |A| - p$, $b = |B| - p$, and $c = |C| - p$ and $q \leq p$. And having that $p \leq a + q$, $p \leq b + q$, $p \leq x$ and $q \leq y$:

$$\begin{aligned} & \underbrace{a^2b + abq - abp}_{\geq 0} + \underbrace{ab^2 + abq - abp}_{\geq 0} \\ & + \underbrace{abc + bcq - bcq}_{\geq 0} + \underbrace{b^2c + bcq - bcp}_{\geq 0} \\ & + \underbrace{2b^2q + 2bq^2 - 2bqp}_{\geq 0} + \underbrace{abq + bq^2 - bqp}_{\geq 0} \\ & + \underbrace{a^2c + acq - acp}_{\geq 0} + \underbrace{abc + acq - acp}_{\geq 0} \\ & + \underbrace{ac^2 + c^2q - c^2p}_{\geq 0} + \underbrace{bc^2 + c^2q - c^2p}_{\geq 0} \\ & + \underbrace{2acq + 2cq^2 - 2cqp}_{\geq 0} + \underbrace{2bcq^2 + 2cq^2 - 2cqp}_{\geq 0} \\ & + \underbrace{abq + aq^2 - aqp}_{\geq 0} + \underbrace{2bq^2 + 2q^3 - 2q^2p}_{\geq 0} \\ & + 2\sqrt{\underbrace{xyz - pyz}_{\geq 0}}\sqrt{\underbrace{xyz - xqz}_{\geq 0}} \geq 0 \end{aligned}$$

Finally, the triangle inequality is demonstrated when $B \cap C = A \cap C$. $\square$

6 DISCUSSION AND FUTURE WORK

This paper proposes the use of semantic-based similarity measures for the comparison on linguistic terms in decision making. In particular, the measures based on the exploitation of the taxonomic relations in an ontology have been reviewed. The main drawbacks of

these measures are that they only exploit the minimum path length between a pair of concepts of an ontology, wasting the rest of taxonomic relations, and that they do not hold the triangle inequality property.

In this paper, the SCD measure that takes into account the ratio between the shared and non-shared taxonomically related concepts of the compared pair of concepts is analyzed. As shown in the evaluation, our proposal outperforms previous attempts, exploiting the taxonomical structure of WordNet.

The paper proves that SCD measure fulfils the metric properties, which means that is suitable to be used in some applications, such as in hierarchical clustering.

In fact, an approximation using semantic measures based on ontologies to make clustering was studied in [2]. In that work, it was demonstrated that the clusters obtained using these measures are more accurate and interpretable.

As future work, we plan to evaluate the SCD measure with some domain ontologies and using other domain dependent benchmarks, which it will be interesting in order to evaluate the dependency of the results in relation to the ontology coverage. We also plan to extend the use of this semantic-based comparison of terms to other decision making methods.

Acknowledgements

This work has been partially funded by the Spanish research projects DAMASK (TIN2009-11005) and ARES (CSD2007-00004) and the Universitat Rovira i Virgili (2009 AIRE-04). Montserrat Batet is also supported by a research grant provided by the Universitat Rovira i Virgili. The authors want to specially thank the help of Dr. Juan A. Rodriguez.

References

- [1] M. Batet, D. Snchez, A. Valls, and K. Gibert. Ontology-based semantic similarity in the biomedical domain. In *12th Conference on Artificial Intelligence in Medicine (AIME'09). WS. Intelligent Data Analysis in Biomedicine and Pharmacology*, Italy, 2009.
- [2] M. Batet, A. Valls, and K. Gibert. Improving classical clustering with ontologies. In *Proceedings of the 4th World conference of the IASC*, number 137–146, Japan, 2008.
- [3] W. R. Dillon and M. Goldstein. *Multivariate Analysis: Methods and Applications*. Wiley, 1984.
- [4] C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press. More information: <http://www.cogsci.princeton.edu/~wn/>, Cambridge, Massachusetts, 1998.
- [5] J. Jiang and D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. of the International Conference on Research in Computational Linguistics*, pages 19–33, Sep 1997.
- [6] C. Leacock and M. Chodorow. *Combining local context and WordNet similarity for word sense identification*, chapter WordNet: An electronic lexical database, pages 265–283. MIT Press, 1998.
- [7] B. Lemaire and G. Denhire. Effects of high-order co-occurrences on word semantic similarities. *Current Psychology Letters - Behaviour, Brain and Cognition*, 18(1), 2006.
- [8] G.A. Miller and W.G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.
- [9] T. Pedersen, S. Pakhomov, S. Patwardhan, and C. Chute. Measures of semantic similarity and relatedness in the biomedical domain. *Journal of Biomedical Informatics*, 40:288–299, 2007.
- [10] R. Rada, H. Mili, E. Bichnell, and M. Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):17–30, 1989.
- [11] P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proc. of the 14th International Joint Conference on Artificial Intelligence (IJCAI 95)*, pages 448–453, Montreal, Canada, 1995.
- [12] H. Rubenstein and J. Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, 1965.
- [13] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. *IEEE Transactions on Data and Knowledge Engineering*, 25(1-2)(1-2):161–197, 1998.
- [14] A. Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.
- [15] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *Proc. of the 32nd annual Meeting of the Association for Computational Linguistics*, pages 133–138, New Mexico, USA, 1994.
- [16] R. Xu and D. Wunsch II. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645–678, May 2005.

PROBABILISTIC DECISION MAKING WITH THE OWA OPERATOR AND THE 2-TUPLE LINGUISTIC APPROACH

José M. Merigó¹

Montserrat Casanovas¹

Macarena Espinilla²

Luis Martínez²

¹ Department of Business Administration, University of Barcelona, {jmerigo, mcasanovas}@ub.edu

² Department of Computer Science, University of Jaén, {martin, mestevéz}@ujaen.es

Abstract

We present a new probabilistic decision making model by using the 2-tuple linguistic representation approach. We introduce the 2-tuple linguistic probabilistic ordered weighted averaging (2T-LPOWA) operator. We study some of its particular cases including the 2-tuple linguistic probabilistic aggregation and the 2-tuple linguistic arithmetic OWA operator. We show an application of the new approach in a decision making problem about the selection of investments.

Keywords: Decision making, Probability, OWA operator, 2-tuple linguistic approach.

1 INTRODUCTION

Decision making problems are very common in a lot of disciplines [3-11,15-19]. A very useful tool to accomplish aggregation process in decision making is the ordered weighted averaging (OWA) operator [15]. Such an aggregation operator provides a method for representing the attitudinal attitude of the decision maker in the aggregation process. Since its appearance, the OWA operator has been studied and applied by many authors [1,3-19].

Recently, Merigó [9] has suggested a new model that unifies the probability with the OWA operator in the same formulation and considering the degree of importance that each concept has in the aggregation. It was called the probabilistic ordered weighted averaging (POWA) operator. Note that in the literature there are other aggregation operators analyzing the use of probabilities and the OWA in the same formulation such as the immediate probability. Note that it is also possible to extend other approaches that use OWAs and weighted

averages (WAs) in the same formulation to the probabilistic framework [12,14].

When using the POWA operator, it is assumed that the available information is numerical. However, this may not be the real situation in decision making problems. Sometimes, the available information is vague or imprecise and it is not possible to analyze it with numerical values and this uncertainty does not have a probability character because they are related to imprecision and vagueness of meaning. Then, a better approach may be the use of linguistic assessments [20]. In the literature, we find a wide range of methods for dealing with linguistic information such as [2,4-6,8,10,13,20]. In this paper, we will use the linguistic 2-tuple representation model [5-6,8,10,13] in order to accomplish processes of computing with words without loss of information.

The aim of this paper is to present a new model for decision making when the available information can be represented with probabilities, OWA operators and linguistic variables. Thus, we are able to deal with decision making problems under risk and under uncertain environments in the same formulation. Moreover, we can assess the uncertain information with linguistic variables obtaining a more complete representation of the decision problem. For doing so, we introduce the 2-tuple linguistic probabilistic ordered weighted averaging (2T-LPOWA) operator. It is a new aggregation operator that includes a wide range of particular cases such as the simple linguistic probabilistic aggregation, the 2-tuple OWA, the 2-tuple linguistic probabilistic arithmetic mean and a lot of other cases. We study some of its main properties and provide an illustrative example in a decision making problem about investment selection.

This paper is organized as follows. In Section 2, we briefly review some basic concepts. Section 3 presents the 2T-LPOWA operator and Section 4 gives an illustrative example. In Section 5 we end the paper summarizing the conclusions.

2 PRELIMINARIES

2.1. 2-TUPLE LINGUISTIC APPROACH

In [5], Herrera and Martínez developed a fuzzy linguistic representation model, which represents the linguistic information with a pair of values called 2-tuple, (s, α) , where s is a linguistic label and α is a numerical value that represents the value of the symbolic translation. With this model, it is possible to accomplish CW processes without loss of information, solving one of the main limitations of the previous linguistic computational models [2,4,20].

Definition 1. Let β be the result of an aggregation of the indexes of a set of labels assessed in the linguistic label set $S = \{s_0, s_1, \dots, s_g\}$, i.e., the result of a symbolic aggregation operation. $\beta \in [0, g]$, being $g + 1$ the cardinality of S . Let $i = \text{round}(\beta)$ and $\alpha = \beta - i$ be two values, such that, $i \in [0, g]$ and $\alpha \in [-0.5, 0.5)$, then α is called a symbolic translation.

Note that the 2-tuple (s_i, α) that expresses the equivalent information to β is obtained with the following function:

$$\Delta : [0, g] \rightarrow S \times [-0.5, 0.5),$$

$$\Delta(\beta) = \begin{cases} s_i & i = \text{round}(\beta), \\ \alpha = \beta - i & \alpha \in [-0.5, 0.5). \end{cases} \quad (1)$$

where round is the usual round operation, s_i has the closest index label to β and α is the value of the symbolic translation.

Proposition 1. Let $S = \{s_0, \dots, s_g\}$ be a linguistic term set and (s_i, α_i) be a linguistic 2-tuple. There is always a Δ^{-1} function, such that, from a 2-tuple it returns its equivalent numerical value $\beta \in [0, g]$.

Proof. It is trivial, we consider the following function:

$$\Delta^{-1} : S \times [-0.5, 0.5) \rightarrow [0, g]$$

$$\Delta^{-1}(s_i, \alpha) = i + \alpha = \beta$$

For further information on the 2-tuple linguistic representation model, see [5-6,8,10,13].

2.2. THE OWA OPERATOR

The OWA operator was introduced by Yager in [15] and it provides a parameterized family of aggregation

operators that include the arithmetic mean, the maximum and the minimum. It can be defined as follows.

Definition 2. An OWA operator of dimension n is a mapping $OWA: R^n \rightarrow R$ that has an associated weighting vector W of dimension n such that the sum of the weights is 1 and $w_j \in [0, 1]$, then:

$$OWA(a_1, a_2, \dots, a_n) = \sum_{j=1}^n w_j b_j \quad (2)$$

where b_j is the j th largest of the a_i .

The OWA operator is commutative, monotonic, bounded and idempotent. For further information about the OWA and its applications, see, [1,3-19].

2.3. THE PROBABILISTIC OWA OPERATOR

The probabilistic ordered weighted averaging (POWA) operator is an aggregation operator that unifies the probability and the OWA operator in the same formulation considering the degree that each concept has in the analysis [8-9]. With this approach, we can either under estimate or over estimate the probabilities according to the attitudinal attitude of the decision maker. It can be defined as follows.

Definition 3. A POWA operator of dimension n is a mapping $POWA: R^n \rightarrow R$ that has associated two weighting vector W and V of dimension n such that w_j and $v_i \in [0, 1]$, and $\sum_{j=1}^n w_j = 1$ and $\sum_{i=1}^n v_i = 1$, according to the following formula:

$$POWA(a_1, \dots, a_n) = \sum_{j=1}^n \hat{v}_j b_j \quad (3)$$

where b_j is the j th largest of the a_i , each argument a_i has an associated weight (probability) v_i with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, $\hat{v}_j = \delta w_j + (1 - \delta)v_j$ with $\delta \in [0, 1]$ and v_j is the weight (probability) v_i ordered according to b_j , that is, according to the j th largest of the a_i .

Note that it is possible to obtain a wide range of particular types of POWA operators [8-9]. Especially, when $\delta = 0$, we get the probabilistic approach, and if $\delta = 1$, we get the OWA operator. Other interesting cases are found when $w_j = 1/n$, for all a_i , because then, we get the arithmetic probability (AP). And if $v_i = 1/n$, for all a_i , we get the arithmetic OWA operator. Note that inside the arithmetic OWA we find the arithmetic maximum and the arithmetic minimum, and so on.

3 LINGUISTIC POWA OPERATOR

The 2-tuple linguistic probabilistic ordered weighted averaging (2T-LPOWA) operator is an extension of the POWA operator for situations where the available information is uncertain but can be assessed with the 2-tuple linguistic representation model. It can also be seen as a unification between linguistic decision making problems under uncertainty (with linguistic OWA operators) and under risk (with probabilities with linguistic information). This approach seems to be complete, at least as an initial real unification between OWA operators and probabilities under a linguistic environment.

However, note that some previous models already considered the possibility of using OWA operators and probabilities in the same formulation. The main model is the concept of immediate probability explained in Section 2 [3,7-8,17-19].

Definition 4. Let S be the set of the 2-tuples. An IP-LOWA operator of dimension n is a mapping IP-LOWA: $S^n \rightarrow S$ that has associated two weighting vector W and V of dimension n such that w_j and $v_i \in [0, 1]$, and $\sum_{j=1}^n w_j = 1$ and $\sum_{i=1}^n v_i = 1$, according to the following formula:

$$IP-LOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) = \Delta(\sum_{j=1}^n \hat{v}_j \Delta^{-1}(s_j, \alpha_j)) \quad (4)$$

where $\Delta^{-1}(s_j, \alpha_j)$ is the j th largest of the $\Delta^{-1}(s_i, \alpha_i)$ values, each $\Delta^{-1}(s_i, \alpha_i)$ has associated a probability v_i with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, $\hat{v}_j = (w_j v_j / \sum_{j=1}^n w_j v_j)$ and v_j is the probability v_i ordered according to $\Delta^{-1}(s_j, \alpha_j)$, that is, according to the j th largest of the $\Delta^{-1}(s_i, \alpha_i)$ values.

Although it seems to be a good approach it is not so complete than the POWA because it can unify OWAs and probabilities in the same model but it can not take in consideration the degree of importance of each case in the aggregation process. Note that the IP-OWA (or IP-LOWA) could be formulated in other ways in order to obtain the immediate weights. For example, we could use $\hat{v}_j = ((w_j + v_j) / \sum_{j=1}^n (w_j + v_j))$, and the results would be also similar and consistent because they would accomplish the main aggregation properties,

$\hat{v}_j = ((\alpha w_j + (1 - \alpha) v_j) / \sum_{j=1}^n (\alpha w_j + (1 - \alpha) v_j))$, and so on.

Other methods that could be considered are the hybrid averaging (HA) operator [14] and the weighted OWA (WOWA) operator [12]. These methods are focused on the weighted average (WA) but it is easy to extend them to probabilities because sometimes the WA is used as a subjective probability. Note that in this case, we could be talking about the 2-tuple WOWA or the 2-tuple HA or its probabilistic version. As said before, these and other approaches are useful for some particular situations but they does not seem to be so complete than the POWA because they can unify OWAs with probabilities (or with WAs) but they can not unify them giving different degrees of importance to each case. Note that in future research we will also prove that these models can be seen as a special case of a general POWA operator (or its respective model with WAs) that uses quasi-arithmetic means. Obviously, it is possible to develop more complex models of the IP-OWA, the HA and the WOWA that takes into account the degree of importance of the OWAs and the probabilities (or WAs) in the model but they seem to be artificial and not a natural unification as it will be shown below.

In the following, we are going to analyze the 2T-LPOWA (or LPOWA) operator. Note that we use the 2-tuple linguistic representation model but it would be possible to use a lot of other linguistic approaches [8]. It can be defined as follows.

Definition 5. Let S be the set of the 2-tuples. A 2T-LPOWA operator of dimension n is a mapping 2T-LPOWA: $S^n \rightarrow S$ that has associated two weighting vector W and V of dimension n such that w_j and $v_i \in [0, 1]$, and $\sum_{j=1}^n w_j = 1$ and $\sum_{i=1}^n v_i = 1$, according to the following formula:

$$2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) = \Delta(\sum_{j=1}^n \hat{v}_j \Delta^{-1}(s_j, \alpha_j)) \quad (5)$$

where $\Delta^{-1}(s_j, \alpha_j)$ is the j th largest of the $\Delta^{-1}(s_i, \alpha_i)$ values, each $\Delta^{-1}(s_i, \alpha_i)$ has associated a probability v_i with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, $\hat{v}_j = \delta w_j + (1 - \delta) v_j$ with $\delta \in [0, 1]$ and v_j is the probability v_i ordered according to $\Delta^{-1}(s_j, \alpha_j)$, that is, according to the j th largest of the $\Delta^{-1}(s_i, \alpha_i)$ values.

Note that it is also possible to formulate the 2T-LPOWA operator separating the part that strictly affects the OWA operator and the part that affects the probabilities. This representation is useful to see both models in the same formulation but it does not seem to be as a unique equation that unifies both models.

Remark 1. Let S be the set of the 2-tuples. A 2T-LPOWA operator of dimension n is a mapping 2T-PLOWA: $S^n \rightarrow S$ that has an associated weighting vector W , with $\sum_{j=1}^n w_j = 1$ and $w_j \in [0, 1]$ and a probabilistic vector V , with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, such that:

$$2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) = \Delta \left(\delta \sum_{j=1}^n w_j \Delta^{-1}(s_j, \alpha_j) + (1 - \delta) \sum_{i=1}^n v_i \Delta^{-1}(s_i, \alpha_i) \right) \quad (6)$$

where $\Delta^{-1}(s_j, \alpha_j)$ is the j th largest of the $\Delta^{-1}(s_i, \alpha_i)$ values and $\delta \in [0, 1]$.

Note that we can distinguish between the descending 2T-LPOWA (2T-LDPOWA) and the ascending 2T-LPOWA (2T-LAPOWA) operator by using $w_j = w_{n-j+1}^*$, where w_j is the j th weight of the 2T-LDPOWA and w_{n-j+1}^* the j th weight of the 2T-LAPOWA operator.

If B is a vector corresponding to the ordered arguments $\Delta^{-1}(s_j, \alpha_j)$, we shall call this the ordered argument vector and W^T is the transpose of the weighting vector, then, the 2T-LPOWA operator can be expressed as:

$$2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) = \Delta(\hat{V}^T B) \quad (7)$$

Remark 2. Note that in some situations we may find that the weighting vector is not normalized, i.e., $\hat{V} = \sum_{j=1}^n \hat{v}_j \neq 1$, then, the 2T-LPOWA operator can be reformulated as follows:

$$2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) = \Delta \left(\frac{1}{\hat{V}} \sum_{j=1}^n \hat{v}_j \beta_j^* \right) \quad (8)$$

The 2T-LPOWA is monotonic, bounded and idempotent. It is not commutative because the probabilistic aggregations (with WAs) are not commutative.

- It is monotonic because if $(s_i, \alpha_i) \geq (s_i^*, \chi_i)$, for all i , then, $2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) \geq 2T-LPOWA((s_1^*, \chi_1), \dots, (s_n^*, \chi_n))$.

- It is bounded because the 2T-LPOWA aggregation is delimited by the linguistic minimum and the linguistic maximum. That is, $\text{Min}\{(s_i, \alpha_i)\} \leq 2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) \leq \text{Max}\{(s_i, \alpha_i)\}$.
- It is idempotent because if $(s_i, \alpha_i) = (s, \alpha)$, for all (s_i, α_i) , then, $2T-LPOWA((s_1, \alpha_1), \dots, (s_n, \alpha_n)) = (s, \alpha)$.

Another interesting issue to analyze are the measures for characterizing the weighting vector W . Following a similar methodology as it has been developed for the OWA operator [8,15,19] we can formulate the attitudinal character, the entropy of dispersion, the divergence of W and the balance operator. Note that these measures affect the weighting vector W but not the probabilities because they are given as some kind of objective information. For example, the attitudinal character can be formulated as follows.

$$\alpha(W) = \sum_{j=1}^n w_j \left(\frac{n-j}{n-1} \right) \quad (9)$$

Another interesting issue to analyze is the different particular cases of the 2T-LPOWA operator that can be obtained by studying the weighting vector W and the parameter δ . For example, we can consider the following cases.

- If $\delta = 0$, we get the 2-tuple linguistic probabilistic aggregation.
- If $\delta = 1$, we get the 2-tuple linguistic OWA operator.
- The 2-tuple linguistic arithmetic probabilistic aggregation (if $w_j = 1/n$, for all j).
- The 2-tuple linguistic arithmetic OWA operator (if $v_i = 1/n$, for all i).
- The 2-tuple linguistic average (if $v_i = 1/n$, for all i , and $w_j = 1/n$, for all j).
- The 2-tuple linguistic maximum probabilistic aggregation ($w_1 = 1$ and $w_j = 0$, for all $j \neq 1$).
- The 2-tuple linguistic minimum probabilistic aggregation ($w_n = 1$ and $w_j = 0$, for all $j \neq n$).
- The 2-tuple linguistic Hurwicz probabilistic criteria ($w_l = \alpha$, $w_n = 1 - \alpha$ and $w_j = 0$, for all $j \neq 1, n$).
- The step-2T-LPOWA ($w_k = 1$ and $w_j = 0$, for all $j \neq k$).
- The olympic-2T-LPOWA operator ($w_l = w_n = 0$, and $w_j = 1/(n-2)$ for all others).
- The general olympic-2T-LPOWA operator ($w_j = 0$ for $j = 1, 2, \dots, k, n, n-1, \dots, n-k+1$; and for all others $w_{j^*} = 1/(n-2k)$, where $k < n/2$).

- The general probabilistic olympic-2T-LOWA operator ($\hat{v}_j = 0$ for $j = 1, 2, \dots, k, n, n-1, \dots, n-k+1$; and for all others $\hat{v}_{j^*} = 1/(n-2k)$, where $k < n/2$).
- The S-2T-LPOWA ($w_l = (1/n)(1 - (\alpha + \beta) + \alpha$, $w_n = (1/n)(1 - (\alpha + \beta) + \beta$, and $w_j = (1/n)(1 - (\alpha + \beta)$ for $j = 2$ to $n-1$ where $\alpha, \beta \in [0, 1]$ and $\alpha + \beta \leq 1$).
- The centered-2T-LPOWA (if it is symmetric, strongly decaying from the center to the maximum and the minimum, and inclusive).
- Etc.

Note that other families of 2T-LPOWA operators may be used following a similar methodology as it has been developed for the OWA operator and its extensions [1,7-19]. Moreover, we could extend this analysis to other types of linguistic approaches [2,4,13,20].

4 ILLUSTRATIVE EXAMPLE

The 2T-LPOWA operator is applicable in a wide range of situations where it is possible to use probabilities, linguistic information and OWA operators. Therefore, we see that the applicability is incredibly broad because all the previous models and theories that use the probability can be extended by using the 2T-LPOWA operator. The reason is that all the problems with probabilities deal with uncertainty. Therefore, in a lot of situations the information is not clear and needs to be assessed with linguistic variables. Moreover by using the 2T-LPOWA operator we can underestimate or overestimate the results according to a degree of orness that we want to have in the aggregation.

Summarizing some of the main fields where it is possible to apply the 2T-LPOWA operator, we can mention: Statistics (especially in probability theory), Mathematics, Economics, Decision theory, Engineering, Physics, etc.

In this paper, we focus on an application in decision making about selection of investments. The main reason for using the 2T-LPOWA operator is that we are able to assess the decision making problem considering linguistic information, probabilities and the attitudinal character of the decision maker. Thus, we get a more complete representation of the decision problem.

We analyze a company that operates in Europe and North America that wants to invest some money in a new market. They consider five alternatives.

- A_1 = Invest in the Asian market.

- A_2 = Invest in the South American market.
- A_3 = Invest in the African market.
- A_4 = Invest in all three markets.
- A_5 = Do not invest money in any market.

In order to evaluate these investments, the investor has brought together a group of experts. This group considers that the key factor is the economic situation of the world economy for the next period. They consider 5 possible states of nature that could happen in the future:

- N_1 = Very bad economic situation.
- N_2 = Bad economic situation.
- N_3 = Regular economic situation.
- N_4 = Good economic situation.
- N_5 = Very good economic situation.

The results of the available investments, depending on the state of nature N_i and the alternative A_k that the decision maker chooses, are shown in Table 1. Note that we assume a set of seven linguistic terms $S = \{s_1 = \text{None}, s_2 = \text{Very Low}, s_3 = \text{Low}, s_4 = \text{Medium}, s_5 = \text{High}, s_6 = \text{Very High}, s_7 = \text{Perfect}\}$.

Table 1: Available investment alternatives

	N_1	N_2	N_3	N_4	N_5
A_1	$(s_3, 0)$	$(s_5, 0)$	$(s_6, 0)$	$(s_2, 0)$	$(s_4, 0)$
A_2	$(s_7, 0)$	$(s_6, 0)$	$(s_2, 0)$	$(s_2, 0)$	$(s_2, 0)$
A_3	$(s_4, 0)$	$(s_2, 0)$	$(s_4, 0)$	$(s_4, 0)$	$(s_5, 0)$
A_4	$(s_6, 0)$	$(s_4, 0)$	$(s_4, 0)$	$(s_5, 0)$	$(s_2, 0)$
A_5	$(s_2, 0)$	$(s_3, 0)$	$(s_7, 0)$	$(s_4, 0)$	$(s_2, 0)$

In this problem, the experts assume the following weighting vector: $W = (0.1, 0.2, 0.2, 0.2, 0.3)$. They assume that the probability that each state of nature will happen is: $P = (0.3, 0.3, 0.2, 0.1, 0.1)$. Note that the OWA operator has an importance of 30% and the probabilistic information an importance of 70%. With this information, we can aggregate the expected results in order to make a decision. In Table 2, we present different results obtained by using different types of 2T-LPOWA operators.

Table 2: Aggregated results

	2T-Prob.	2TOWA	2T-PAM	POWA
A_1	$(S_4, 0.2)$	$(S_4, -0.4)$	$(S_4, 0.14)$	$(S_4, 0.02)$
A_2	$(S_5, -0.3)$	$(S_3, 0.3)$	$(S_4, 0.43)$	$(S_4, 0.28)$
A_3	$(S_3, 0.5)$	$(S_3, 0.5)$	$(S_4, -0.41)$	$(S_3, 0.5)$
A_4	$(S_4, 0.5)$	$(S_4, -0.2)$	$(S_4, 0.41)$	$(S_4, 0.29)$
A_5	$(S_3, 0.5)$	$(S_3, 0.1)$	$(S_4, -0.47)$	$(S_3, 0.38)$

If we establish an ordering of the investments, a typical situation if we want to consider more than one alternative, then, we get the results shown in Table 3. Note that the first alternative in each ordering is the optimal choice.

Table 3: Ordering of the investments

	Ordering
2-tuple probabilistic	$A_2 \setminus A_4 \setminus A_1 \setminus A_3 = A_5$
2T-OWA	$A_4 \setminus A_1 \setminus A_3 \setminus A_2 \setminus A_5$
2T-PAM	$A_2 \setminus A_4 \setminus A_1 \setminus A_3 \setminus A_5$
2T-POWA	$A_4 \setminus A_2 \setminus A_1 \setminus A_3 \setminus A_5$

As we can see, depending on the aggregation operator used, the ordering of the investments may be different. Sometimes, it is clear which the best alternative is because one alternative is clearly better than the others. But sometimes, it is not clear because the alternatives give similar results and depending on the state of nature that happens in the future the optimal alternative may be different. Therefore, depending on the particular type of 2T-LPOWA operator used, the optimal choice may change because the 2T-LPOWA provides a parameterized family of aggregation operators between the maximum and the minimum. Thus, when looking to the maximum and similar results, we may find an optimal alternative that is different than looking to the minimum.

5 CONCLUSIONS

We have presented the 2T-LPOWA operator. It is a new aggregation operator that unifies the probability and the OWA operator in the same formulation and considering the degree of importance that each concept has in the aggregation. Moreover, it also deals with uncertain environments where the available information can be assessed with linguistic variables. Thus we get a more complete representation of the uncertain information and we are able to compute with words in the aggregation process. We have analyzed some of its main particular cases such as the 2-tuple linguistic probabilistic aggregation, the 2-tuple linguistic OWA operator, the 2-tuple linguistic arithmetic mean, the 2-tuple linguistic arithmetic OWA operator and the 2-tuple linguistic probabilistic arithmetic mean.

We have also studied the applicability of the new approach and we have seen that it is very broad because all the studies that use the probability can be revised and extended with this new model. We have focussed on a decision making problem about the selection of investments. We have seen that depending on the particular type of 2T-LPOWA operator used, the results may lead to different decisions.

In future research we expect to develop further developments of this approach by using other types of linguistic information and adding new characteristics and generalizations such as the use of order-inducing variables, distance measures and quasi-arithmetic means.

We will also consider other business applications and we will extend it to group decision making.

Acknowledgements

We would like to thank the anonymous reviewers for useful comments that have improved the quality of the paper.

References

- [1] G. Beliakov, A. Pradera and T. Calvo, *Aggregation Functions: A Guide for Practitioners*. Berlin: Springer-Verlag, 2007.
- [2] H. P. Bonissone, "A Fuzzy Sets Based Linguistic Approach: Theory and Applications", in: M.M. Gupta and E. Sanchez, Eds., *Approximate Reasoning in Decision Analysis*, North-Holland, pp. 329-339, 1982.
- [3] K.J. Engemann, D.P. Filev and R.R. Yager, Modelling decision making using immediate probabilities. *International Journal of General Systems*, 24:281-294, 1996.
- [4] F. Herrera, E. Herrera-Viedma and J.L. Verdegay, "A Sequential Selection Process in Group Decision Making with a Linguistic Assessment Approach", *Information Sciences*, 85, 1995, pp. 223-239.
- [5] F. Herrera and L. Martínez, "A 2-tuple Fuzzy Linguistic Representation Model for Computing with Words", *IEEE Transactions on Fuzzy Systems*, 8, 2000, pp. 746-752.
- [6] F. Herrera and L. Martínez, "The 2-tuple Linguistic Computational Model. Advantages of its linguistic description, accuracy and consistency", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9, 2001, pp. 33-48.
- [7] J.M. Merigó, Using immediate probabilities in fuzzy decision making. In: *Proceedings of the 22nd ASEPELT Conference*, Barcelona, pp. 1650-1664, 2008.
- [8] J.M. Merigó, *New extensions to the OWA operators and its application in decision making* (In Spanish). PhD Thesis, Department of Business Administration, University of Barcelona, 2008.
- [9] J.M. Merigó, Probabilistic decision making with the OWA operator and its application in investment management. In: *Proceedings of the IFSA-EUSFLAT Conference*, Lisbon, Portugal, pp. 1364-1369, 2009.
- [10] J.M. Merigó, M. Casanovas, L. Martínez, A decision making model based on Dempster-Shafer theory and linguistic hybrid aggregation operators. In:

- Proceedings of the 8th HIS Conference*, Barcelona, pp. 80-85, 2008.
- [11] J.M. Merigó and A.M. Gil-Lafuente, The induced generalized OWA operator. *Information Sciences*, 179:729-741, 2009.
 - [12] V. Torra, The weighted OWA operator. *International Journal of Intelligent Systems*, 12:153-166, 1997.
 - [13] J.H. Wang and J. Hao, "A new version of 2-tuple fuzzy linguistic representation model for computing with words", *IEEE Transactions on Fuzzy Systems*, 14:435-445, 2006.
 - [14] Z.S. Xu and Q.L. Da, An overview of operators for aggregating information. *International Journal of Intelligent Systems*, 18:953-968, 2003
 - [15] R.R. Yager, On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man and Cybernetics B*, 18:183-190, 1988.
 - [16] R.R. Yager, Families of OWA operators. *Fuzzy Sets and Systems*, 59:125-148, 1993.
 - [17] R.R. Yager, Including decision attitude in probabilistic decision making. *International Journal of Approximate Reasoning*, 21:1-21, 1999.
 - [18] R.R. Yager, K.J. Engemann, D.P. Filev, On the concept of immediate probabilities, *International Journal of Intelligent Systems*, 10:373-397, 1995.
 - [19] R.R. Yager and J. Kacprzyk, *The Ordered Weighted Averaging Operators: Theory and Applications*. Norwell: Kluwer Academic Publishers, 1997.
 - [20] L.A. Zadeh, "The Concept of a Linguistic Variable and its application to Approximate Reasoning. Part 1", *Information Sciences*, 8:199-249, 1975; "Part 2", *Information Sciences*, 8:301-357, 1975; "Part 3", *Information Sciences*, 9:43-80, 1975.

A MOBILE LINGUISTIC DECISION SUPPORT SYSTEM TO HELP USERS IN THEIR E-COMMERCE ACTIVITIES

I.J. Pérez¹ S. Alonso² F.J. Cabrerizo³ E. Herrera-Viedma¹

¹ Dept. of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain
{ijperez,viedma}@decsai.ugr.es

² Dept. of Software Engineering, University of Granada, 18071 Granada, Spain
zerjioi@ugr.es

³ Dept. of Software Engineering and Computer Systems, Distance Learning University of Spain (UNED), 28040 - Madrid, Spain
cabrerizo@issi.uned.es

Abstract

With the recent spreading of social networks and digital communities, the Web has a new business model for providing free content-sharing services. Using the power of the masses and the collective intelligence, Web 2.0 sites have created enormous e-commerce opportunities for marketing and advertising. Thus, there is still a necessity of developing tools to help users to reach good decisions. We propose to manage this collective intelligence with a Mobile Decision Support System (MDSS). It allows that users with the same user profile that the potential customer can act as a set of shop assistants that advises him/her according with their own experiences. Then, the MDSS obtains a collective advice that helps customers to make decisions related with their e-commerce activities.

Keywords: Group decision making, e-commerce, decision support system, linguistic approach, web 2.0, social network, collective intelligence.

1 INTRODUCTION

The global build of Internet connectivity and growing availability of computing and communication devices have made the World Wide Web a new virtual borderless continent. New Web technologies have allowed the creation of many different services where users from all over the world can join, interact and produce new contents and resources. One of the most recent trends, the so called *Web 2.0* [14], represents a paradigm shift in how people use the web which comprises a set of

different web development and design techniques. It allows the easy communication, information sharing, interoperability and collaboration in this new virtual environment. Web 2.0 Communities, that can take different forms as Internet forums, groups of blogs, social network services and so on, provide a platform in which users can collectively contribute to a Web presence and generate massive content behind their virtual collaboration [12].

Due to the above new Web features, we can think about great potentials and challenges for the future of some internet business like electronic commerce (e-commerce). E-commerce is the buying and selling of goods and services on the Internet, and it is now spreading into all walks of life. Even, users can view, select and pay for online services in a mobile framework [1, 12]. New challenges about e-commerce involves new technologies, services and business models. Social shopping sites emerged as the latest developments to leverage the power of social networking with online shopping. Users on social shopping sites can post product recommendations, create wish lists, post photos, make purchases, and form social shopping communities [12].

We propose to incorporate new services (web 2.0) and technologies (mobile devices) in a decision support system (DSS) for advising customers in their e-commerce experiences on-time. The central goal of DSSs is to process and provide suitable information in order to support individuals or organizations in their decision making tasks like to decide where travel in holidays or shopping elections [4]. Usually, people bring their mobile devices with them anywhere. So, they have the possibility to use some mobile services wherever they go. Studies of mobile commerce suggest that there is a general customer interest in the services that it provides, like shopping and banking applications [3, 11, 15]. Its adoption, however, has been slower than expected. Mobile phones impose very different constraints than desktop computers. It has been ar-

gued that this stems from complexity of the transactions, perceived lack of security, lack of user friendly mobile portals, etc. [7, 15].

Due to the big amount of different items and online advertisements, the first problem for the customer is to rank the different alternatives in order to buy the best one taking into account his/her connected partners' advice. This problem can be seen as a group decision making (GDM) problem because it includes all the necessary requirements for this kind of problems.

GDM models are used to obtain the best solution(s) for a problem according to the information provided by some decision makers (expert). Usually, each expert may approach the decision process from a different angle, but they have a common interest in reaching an agreement on taking the best decision. Concretely, in a GDM problem we have a set of different alternatives to solve the problem and a set of experts which are usually required to provide their preferences about the alternatives by means of a particular preference format [9]. In the case of e-commerce advising, the alternatives are the items or services that can be bought by the customer, and the set of experts of the problem are the members connected with the customer's social network.

To deal with the above situations, we present a mobile DSS that could be incorporated as a tool into a social network to help customers in their e-commerce activities. To advise customers in their electronic shopping, the MDSS shows to the customer the ranking extracted by aggregating the collective intelligence of the virtual community. In such a way, our system allows that the members connected with the customer help him/her to choose the best good or service of the stock, according to the customer's needs. To represent the preferences provided by the social network members we use a *fuzzy linguistic modelling* [18]. Thus, to compute the quality assessments we use computing with words tools based on linguistic aggregation operators [10] and an estimation algorithm to deal with incomplete information [2].

To do so, the paper is set as follows: in section 2 we present our preliminaries, that is, some of the most important characteristics of Web 2.0 Communities and the basic concepts that we use in our paper like GDM problems and computing with words. Section 3 deals with the incorporation of the MDSS as a mobile web 2.0 service. Finally, in section 4 we point out our conclusions.

2 PRELIMINARIES

2.1 New E-Commerce Opportunities Based on Web 2.0

Emerging web and mobile technologies provide the foundation for new e-commerce opportunities and applications. They also generate the push for new enterprise system architectures in order to deploy new Web technologies and transform businesses into e-commerce driven operations. This new framework allows people from all over the globe to meet other individuals which share some of their interests. Thus, virtual communities can be created in order to collaborate, communicate, share information or resources and so on. The Web has become a media supported by advertisers that, in turn, sell their products to users via e-commerce. In digital cyberspace, e-commerce is moving into a new border, mixing physical and digital goods trading. Companies have even created their own digital currencies for online exchange of goods and services [1, 12, 13].

E-commerce is an area of study that has practical applications and a prominent future. Although the Internet bubble has generated some negative sentiment about the industry, many previously unsuccessful e-commerce ventures and business models have now reemerged and become feasible and profitable. With ubiquitous broadband connectivity and powerful personal devices (including PCs, mobile phones, and media players), mobile commerce has become a growing part of our daily lives. More important, endless opportunities still exist to deploy new mobile commerce products and services that are simply impossible without automated Web-based and mobile support [1, 12, 13].

With most online shopping, people have numerous choices when looking for a specific item in which they are interested. Many shoppers use search engines but often get more information than what they really want. In this paper, we want to identify an unexplored market and to create new products and services using mobile tools and collective intelligence to help customers. To do so, we propose a linguistic mobile decision support system that acts like a recommender system where the recommendations are given by other users of the social networking web site who have the same user profile that the customer. This application could improve the customers' satisfaction in their e-commerce activities.

2.2 Group Decision Making Models

A decision making process, consisting in deriving the best option from a feasible set, is present in just about

every conceivable human task. It is obvious that the comparison of different actions according to their desirability in decision problems, in many cases, cannot be done by using a single criterion or an unique person. Thus, we interpret the decision process in the framework of group decision making (GDM).

There have been several efforts in the specialized literature to create different models to correctly address and solve GDM situations. Some of them make use of fuzzy theory as it is a good tool to model and deal with vague or imprecise opinions. Many of those models are usually focused on solving GDM situations in which a particular issue or difficulty is present. For example, there have been models that allow to use linguistic assessments instead of numerical ones, thus making it easier for the experts to express their preferences about the alternatives [9]. Other models allow experts to use multiple preference structures [5] and other different approaches deal with incomplete information situations if experts are not able to provide all their preferences when solving a GDM problem [2].

In a GDM problem we have a finite set of feasible alternatives. $X = \{x_1, x_2, \dots, x_n\}$, ($n \geq 2$) and the best alternative from X has to be identified according to the information given by a set of experts, $E = \{e_1, e_2, \dots, e_m\}$, ($m \geq 2$).

Usual resolution methods for GDM problems are composed by two different processes [9] *i) consensus process and ii) Selection process*. (see Figure 1):

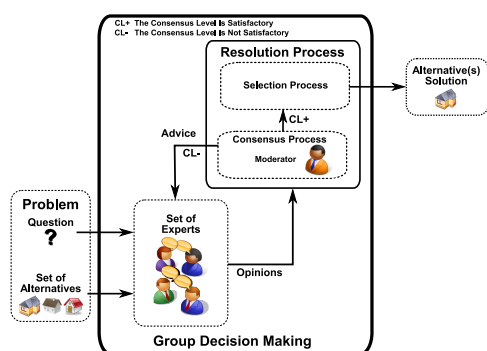


Figure 1: Resolution process of a GDM

In this paper we center our attention only in the selection process implementation, where the system has to obtain the solution ranking of alternatives from the opinions on the alternatives given by the experts.

2.3 Use of Incomplete Linguistic Information in GDM Problems

There are some situations in which the information may be not quantified due to its nature, and thus, it may be stated only in linguistic terms. Linguistic de-

cision analysis is applied for solving decision making problems under linguistic information. Its application in the development of the theory and method in decision analysis is very beneficial because it introduces a more flexible framework which allows us to represent the information in a more direct and adequate way when we are unable to express it precisely. So, this method has been widely used in decision field [2].

The ordinal fuzzy linguistic modelling [10] is a useful kind of fuzzy linguistic approach proposed as an alternative tool to the traditional fuzzy linguistic modelling [18]. This tool simplifies the computing with words process as well as some linguistic aspects of problems. It is defined by considering a finite and totally ordered label set $S = \{s_i\}$, $i \in \{0, \dots, g\}$ in the usual sense, i.e., $s_i \geq s_j$ if $i \geq j$, and with odd cardinality (usually 7 or 9 labels).

Other different issue arises when each expert has his/her own experience concerning the problem being studied and they could have some difficulties in giving all their preferences. This may be due to an expert not possessing a precise or sufficient level of knowledge of the problem, or because that expert is unable to discriminate the degree to which some options are better than others. In such situations, experts are forced to provide incomplete fuzzy linguistic preference relations (FLPRs) [2, 16]. Therefore, it is of great importance to provide tools to deal with this lack of knowledge in experts' opinions.

We assume that each social network member e_h provides his/her preferences by means of a FLPR P^h which, depending on the situation, could be incomplete. So, before to start the selection process, we have to fill the gaps of all the missing values by using an additive consistency based estimation process developed in [2].

On the other hand, an useful linguistic aggregation operator is the Linguistic Ordered Weighted Averaging (LOWA) operator which has been extensively used in the literature by its good axiomatic properties [10]. We shall use it in our MDSS.

3 A MOBILE LINGUISTIC DECISION SUPPORT SYSTEM TO HELP USERS IN THEIR E-COMMERCE ACTIVITIES

To develop the MDSS as a Web 2.0 service, the system asks customer his/her current needs (the tool offers a personalized service). Taking into account these needs, together with the community's collective knowledge, the system shows to the customer the ranking through his own mobile device. Therefore, the cus-

tomers receive a social support to choose the best item for buying. The advice is represented by means of linguistic rankings of shopping alternatives obtained from the individual preference relations provided by the social network members applying a selection process.

To do so, we have chosen a classical “Client/Server” architecture, where the client is a mobile device. What is more, as we want to develop the MDSS like a web service, our implementation has thin client features such as no additional technology investment required, and no risk of client-side software becoming obsolete. There is no need for additional servers, and no unique client-side software or upgrade costs. Its main drawback is that all content cannot be easily delivered to all browsers.

In what follows, we describe the client interface and the server structure in detail.

3.1 Client Interface

The client software shows a specific interface to the users depending on the kind of user. At first, the user has to introduce his/her password in the system, then, the system decides the kind of interface to show. There are two different interfaces: *Customer interface* and *Member Interface*.

1. *Customer interface*: This interface is designed for the customers that need help with their m-commerce experiences.
 - *Select alternatives*: To introduce the preferred items to buy (see Figure 2 a).
 - *Needs survey*: To choose the importance degree of the item’s features (see Figure 2 b).
 - *Output*: At the end of the decision process, the device will show the set of solution alternatives as an ordered set of alternatives marking the most relevant ones (see Figure 3 a).
2. *Member Interface*: This interface is designed for the social network members that can help the potential customers.
 - *Problem description*: The device shows to the members a brief description about the problem and the discussion subset of alternatives.
 - *Insertion of preferences*: The device has a specific interface to insert the linguistic preferences using a set of labels. To introduce the preferences on the interface, the user has to use the keys of the device (see Figure 3 b).



Figure 2: a) Selection of alternatives. b) Needs survey



Figure 3: b) Displayed ranking advice. b) Insertion of individual advices

3.2 Server Structure

The web server is the main side of the MDSS. It implements the estimation process, the selection process and the database that stores not only the problem data, but also problem parameters and some information generated during the decision process. The communication with the client to receive/send information from/to the experts is supported by mobile Internet (M-Internet) technologies.

Once the customer has submitted his/her preferred items and his/her current needs, the members connected with him/her have to give their opinions, taking into account the customer’s needs, about the alternatives that the customer has selected (see Figure 2). When all the experts have given their opinions using incomplete FPLRs as element of preferences’ representation, the system starts to compute the collective advice.

3.2.1 Estimation Process

In [2] we develop an additive consistency based estimation process of missing values to deal with incomplete FLPRs defined in a 2-tuple linguistic context. In this paper we adapt it to deal with incomplete FLPRs defined in an ordinal linguistic context.

To deal with incomplete FLPRs we need to define the following sets [16]:

$$\begin{aligned} A &= \{(i, j) \mid i, j \in \{1, \dots, n\} \wedge i \neq j\} \\ MV^h &= \{(i, j) \in A \mid p_{ij}^h \text{ is unknown}\} \\ EV^h &= A \setminus MV^h \end{aligned} \quad (1)$$

where MV^h is the set of pairs of alternatives whose preference degrees are not given by expert e_h and EV^h is the set of pairs of alternatives whose preference degrees are given by the expert e_h . We do not take into account the preference value of one alternative over itself as this is always assumed to be equal to $s_{g/2}$.

The subset of missing values that could be estimated in step t of the process, called EMV_t^h (*estimated missing values*), is defined in [2].

When $EMV_{maxIter}^h = \emptyset$, with $maxIter > 0$, the procedure will stop as there will not be any more missing values to be estimated. Furthermore, if $\bigcup_{l=0}^{maxIter} EMV_l^h = MV^h$, then all missing values are estimated, and, consequently, the procedure is said to be successful in the completion of the incomplete FLPR.

In iteration t , to estimate a particular value p_{ik}^h with $(i, k) \in EMV_t^h$, we use the function $estimate\_p(h, i, k)$ [2].

Then, the complete iterative estimation procedure is the following:

ITERATIVE ESTIMATION PROCEDURE

0. $EMV_0^h = \emptyset$
1. $t = 1$
2. while $EMV_t^h \neq \emptyset$ {
3. for every $(i, k) \in EMV_t^h$ {
4. estimate- $p(h, i, k)$
5. }
6. $t++$
7. }

3.2.2 Selection Process

The selection process implemented computes all the individual preferences to obtain a collective advice. It has two different phases: Aggregation and Exploitation [8]:

1. Aggregation:

The aggregation phase defines a collective preference relation, $P^c = (p_{ij}^c)$, obtained by means of the aggregation of all individual linguistic preference relations $\{P^1, P^2, \dots, P^m\}$. It indicates the global preference between every pair of items according to the majority of experts' opinions. The aggregation is carried out by means of a LOWA operator [6, 10].

A LOWA operator of dimension n is a function $\phi : S^n \rightarrow S$ that has a weighting vector associated with it, $W = (w_1, \dots, w_n)$, with $w_i \in [0, 1]$, $\sum_{i=1}^n w_i = 1$, and it is defined according to the following expression:

$$\phi_Q(p_1, \dots, p_n) = \sum_{i=1}^n w_i \cdot p_{\sigma(i)}, \quad p_i \in S.$$

being $\sigma : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$ a permutation defined on linguistic values, such that $p_{\sigma(i)} \geq p_{\sigma(i+1)}, \forall i = 1, \dots, n-1$, that is, $p_{\sigma(i)}$ is the i -highest linguistic value in the set $\{p_1, \dots, p_n\}$.

A natural question in the definition of OWA operators is how to obtain W . In [17] it was defined an expression to obtain W that allows to represent the concept of fuzzy majority by means of a fuzzy linguistic non-decreasing quantifier Q :

$$w_i = Q(i/n) - Q((i-1)/n), \quad i = 1, \dots, n. \quad (2)$$

Therefore, in our model the collective FLPR is obtained as follows:

$$p_{ij}^c = \phi_Q(p_{ij}^1, \dots, p_{ij}^m) \quad (3)$$

We can use the linguistic quantifier *most of* defined as $Q(r) = r^{1/2}$, that generates a weighting vector of n values. As an example with three experts, $W = (0.58, 0.24, 0.18)$. To obtain each collective ordinal linguistic preference value p_{ij}^c we have to do that follows:

- $p_{ij}^1 = M, \quad p_{ij}^2 = L, \quad p_{ij}^3 = VL \Rightarrow \sigma(1) = 1, \quad \sigma(2) = 3, \quad \sigma(3) = 2$
- $p_{ij}^c = \Gamma(w_1 \cdot \Gamma^{-1}(p_{ij}^1) + w_2 \cdot \Gamma^{-1}(p_{ij}^3) + w_3 \cdot \Gamma^{-1}(p_{ij}^2)) = \Gamma(0.58 \cdot 3 + 0.24 \cdot 1 + 0.18 \cdot 2) = \Gamma(2.34) = L$

2. Exploitation:

This phase transforms the global information about the alternatives into a global ranking of them, from which the set of solution alternatives is obtained. The global ranking is obtained applying two choice degrees of alternatives on the collective preference relation QGDD and QGNDD. These degrees can be studied in more detail in [2].

When the system has computed all these values, the customer receives them in his mobile device in form of advice (see Figure 3 a):

4 CONCLUSIONS

We have presented a MDSS tool based on GDM models as a Web 2.0 service related with e-commerce and collective intelligence that is represented with incomplete FLPRs. This tool uses the advantages of M-Internet communication technologies to advise the customer in their m-commerce experiences and improves the customer satisfaction with the decision of purchase in anytime and anywhere. We have used smartphones as device to send the experts' preferences but the structure of the prototype is designed to use any mobile device as PDAs.

Acknowledgements

The word **Acknowledgements** This paper has been developed with FEDER funds, PETRI project (PET2007-0460) and FUZZYLING project (TIN2007-61079).

References

- [1] JM. Aguado, IJ. Martinez, "From social web to mobile 2.0: The 2.0 paradigm in the mediatization process of mobile communications", *Profesional de la informacion*, 18, 2, 2009, 155–161.
- [2] S. Alonso, F.J. Cabrerizo, F. Chiclana, F. Herrera, and E. Herrera-Viedma. Group decision-making with incomplete fuzzy linguistic preference relations. *International Journal of Intelligent Systems*, 24(2):201–222, 2009.
- [3] B. Anckar and D. D'Incau, "Value creation in Mobile commerce: Findings from a consumer survey", *JITTA Journal of Information Technology Theory and Application*, 4, 1, (2002), 43–64.
- [4] F. Burstein, J. Cowie, A. Zaslavsky, and J. San Pedro. *Handbook on Decision Support Systems*. Springer, (2008).
- [5] E. Herrera-Viedma, F. Herrera, and F. Chiclana. A consensus model for multiperson decision making with different preference structures. *IEEE Tran. on Systems, Man & Cybernetics, p. A*, 32(3):394–402, 2002.
- [6] M. Delgado, J.L. Verdegay, and M.A. Vila. On aggregation operations of linguistic labels. *International Journal of Intelligent Systems*, 8:351–370, 1993.
- [7] Frolick, M.N. and Chen, L.d., "Assessing m-commerce opportunities", *Information Systems Management*, 21, 2, (2004), 53–61.
- [8] F. Herrera, E. Herrera-Viedma, and J.L. Verdegay. A sequential selection process in group decision making with a linguistic assessment approach. *Information Sciences*, 85(4):223–239, 1995.
- [9] F. Herrera, E. Herrera-Viedma, and J.L. Verdegay. A model of consensus in group decision making under linguistic assessments. *Fuzzy Sets and Systems*, 78(1):73–87, 1996.
- [10] F. Herrera, E. Herrera-Viedma, and J.L. Verdegay. Direct approach processes in group decision making using linguistic owa operators. *Fuzzy Sets and Systems*, 79:175–190, (1996).
- [11] Herzberg, A., "Payments and Banking with Mobile Personal Devices", *Communications of the ACM*, 46, 5, (2003), 53–58.
- [12] K.J. Lin, "E-commerce technology - Back to a prominent future", *IEEE Internet Computing* 12, 1, (2008), 60–65.
- [13] K.J. Lin, "Building web 2.0", *Computer*, online, (2007), 101–102.
- [14] T. Oreilly. What is web 2.0: Design patterns and business models for the next generation of software. *Communications & Strategies*, 1:17, 2007.
- [15] Siau, K. and Shen, Z., "Building Customer Trust in Mobile Commerce", *Communications of the ACM*, 46, 4, (2003), 91–94.
- [16] E. Herrera-Viedma, S. Alonso, F. Chiclana, and F. Herrera. A consensus model for group decision making with incomplete fuzzy preference relations. *IEEE Transactions on Fuzzy Systems*, 15(5):863–877, 2007.
- [17] R.R. Yager. On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190, 1988.
- [18] L.A. Zadeh. The concept of a linguistic variable and its applications to approximate reasoning. *Information Sciences*, 8:199–249, 1975.

LICITACIÓN DE PROYECTOS PÚBLICOS CON VALORACIONES LINGÜÍSTICAS

Edurne Falcó Díaz de Cerio¹ José Luis García Lapresta²

¹ PRESAD Research Group, Universidad de Valladolid, edurne@eco.uva.es
SEED Research Group, Universidad Pública de Navarra

² PRESAD Research Group, Universidad de Valladolid, lapresta@eco.uva.es

Resumen

Este trabajo presenta una propuesta para evitar paradojas del ranking en licitaciones de proyectos públicos, en donde los criterios son evaluados independientemente para cada criterio. En particular, a cada proyecto se le asigna un término lingüístico de una escala cualitativa en cada uno de los criterios evaluados, según sus méritos, y se agregan las valoraciones obtenidas mediante un proceso puramente ordinal basado en la metodología *Majority Judgement* introducida recientemente por Balinski y Laraki. Con ello se obtiene una ordenación de los proyectos presentados a concurso.

Palabras Clave: licitaciones públicas, valoraciones lingüísticas, Majority Judgement, mediana.

1 INTRODUCCIÓN

La licitación de proyectos públicos es una de las formas más utilizadas en la concesión de contratos con las Administraciones Públicas. Si bien es la más usual, eso no implica que esté falta de problemas. La valoración que se lleva a cabo de los proyectos presentados lleva en muchas ocasiones a que la puntuación de un proyecto se vea influenciada por otros, originando sistemas de asignación poco robustos y faltos de transparencia.

Que haya anomalías en la adjudicación de proyectos públicos no solo afecta a la equidad y a la justicia de las decisiones públicas, sino que tiene gran relevancia económica, debido a que existen grandes sumas de dinero involucradas en tales concesiones. A todo esto hay que añadir que los problemas causados por adjudicaciones anómalas de proyectos públicos han acabado

en numerosas denuncias y en algunas de ellas han tenido que intervenir el Tribunal de Justicia de las Comunidades Europeas y otros organismos.

Chen [6] muestra varias paradojas cuando la puntuación para algún criterio se determina mediante comparación entre los proyectos presentados por los licitantes. Uno de los criterios más afectados por este tipo de paradojas es el precio, al que comúnmente se aplican fórmulas para su valoración.

Ejemplo 1. Consideremos tres proyectos *A*, *B* y *C* presentados a licitación pública y dos criterios: *precio* y *calidad*. La puntuación del criterio *precio* es valorada de la siguiente manera:

$$\text{Precio} = 50 \cdot \frac{L}{P}$$

siendo *L* el precio más bajo de todos los proyectos presentados y *P* el precio del proyecto licitante cuya puntuación se está calculando.

La Tabla 1 muestra una situación concreta donde la puntuación de la *calidad* viene dada directamente y la del *precio* se calcula por la expresión anterior. Puede observarse que los proyectos *A* y *C* estarían empatados en la posición ganadora con 75 puntos.

Tabla 1: Ejemplo de paradoja del ranking

	Pt. calidad	Precio	Pt. precio	Total
<i>A</i>	25	40	50	75
<i>B</i>	32	50	40	72
<i>C</i>	50	80	25	75

Supongamos que el licitante del proyecto *C* proclama su derecho a haber ganado debido a que el proyecto *A* no cumple alguno de los requisitos exigidos. Si por esta razón el proyecto *A* es excluido y se utilizan las puntuaciones contenidas en la Tabla 1, ciertamente *C* sería el nuevo ganador. Ahora bien, si se realizara una nueva calificación teniendo en cuenta tan solo los

proyectos restantes, B y C , los resultados variarían, ya que el proyecto A era el que menor precio poseía y la puntuación de los otros dos proyectos se vería alterada. La Tabla 2 recoge las nuevas puntuaciones.

Tabla 2: Ejemplo de paradoja del ranking al eliminar el proyecto A

	Pt. calidad	Precio	Pt. precio	Total
B	32	50	50	82
C	50	80	31.25	81.25

El ganador en este caso sería el proyecto B , que era el peor clasificado en la situación anterior. Esto es posible cuando las puntuaciones son relativas, es decir, cuando la puntuación para algún criterio se determina comparando los diferentes proyectos entre sí.

Las paradojas del ranking están asociadas a lo que se denomina en el contexto de la *Elección Social* como el incumplimiento del principio de *independencia de alternativas irrelevantes* (Arrow [1]). Este principio postula que el ranking relativo entre dos alternativas ha de depender únicamente de las valoraciones o preferencias sobre dichas alternativas y no debe verse afectado por otras alternativas, que han de ser irrelevantes en tal comparación¹.

Una forma de impedir la aparición de las paradojas del ranking en la licitación de proyectos públicos consiste en evaluar cada criterio de forma independiente para cada proyecto, sin compararlos entre sí. En este trabajo se procederá de esta forma y, además, se considerará que las valoraciones de los criterios se emiten mediante términos lingüísticos de una escala fijada previamente, con objeto de tener en cuenta la habitual vaguedad existente en los criterios a evaluar (véase el Ejemplo 5, tomado de un caso real de licitación [13]).

En una propuesta reciente, Balinski y Laraki [3, 4] propugnan un nuevo método de votación denominado *Majority Judgement* con el que pretenden evitar este marco “insatisfactorio” de Arrow (véase nota 1). Con este método desaparece el problema subyacente en la licitación de proyectos públicos, el incumplimiento del principio de *independencia de alternativas irrelevantes*, por lo que inicialmente se presenta como un procedimiento adecuado para la solución de las paradojas del ranking. El punto principal del método consiste en tomar el grado del votante mediano como valoración colectiva, ya que de esta forma no puede se

¹En el mismo contexto de *Elección Social*, Arrow [1] demostró con precisión matemática, en su célebre *teorema de imposibilidad*, que si unimos el principio de *independencia de alternativas irrelevantes* a otros tres principios aceptables, no existirá ninguna función de bienestar social que pueda cumplir todos ellos.

considerado ni demasiado alto ni demasiado bajo por una mayoría absoluta de agentes. Conviene destacar que este método fue puesto en práctica con éxito en un experimento en tres distritos electorales durante las elecciones presidenciales de Francia en el año 2007 [5].

El trabajo está organizado como sigue. En la sección 2 se presenta la metodología de *Majority Judgement*. En la sección 3 se incluye nuestra propuesta de aplicación de *Majority Judgement* para la licitación de proyectos públicos y el ajuste de *Majority Judgement* presentado por García-Lapresta y Martínez-Panero en [8]. En la sección 4 se presentan algunas conclusiones y el trabajo futuro.

2 MAJORITY JUDGEMENT

Balinski y Laraki introducen en [3, 4] un nuevo sistema de votación para elecciones políticas denominado *Majority Judgement* (**MJ**), el cual considera que cada votante emite un juicio sobre cada candidato mediante un término lingüístico de una escala ordenada común, previamente fijada. Una vez conocidas las opiniones de los votantes, se asigna como valoración colectiva la mediana menor de las valoraciones individuales. Para desempatar, los autores proponen un procedimiento secuencial en el que se tienen en cuenta el número de opiniones por encima y por debajo de la valoración colectiva.

El modelo general de **MJ** consta de un conjunto de agentes o votantes $V = \{1, \dots, m\}$ ($m \geq 3$) que muestra su opinión sobre un conjunto de alternativas $X = \{x_1, \dots, x_n\}$ ($n \geq 3$) a través de una escala finita estrictamente ordenada $L = \{l_1, \dots, l_g\}$, con $l_1 < \dots < l_g$ y $g \geq 2$. Los términos de L pueden ser etiquetas lingüísticas como *muy mal*, *mal*, *regular*, *bien*, *muy bien* y *excelente*. La *granularidad* de L es el cardinal de L , $\#L = g$.

Un *perfil* es una matriz $m \times n$, (a_j^i) , donde $a_j^i \in L$ es la asignación que el agente i otorga a la alternativa x_j . El conjunto de perfiles se denota mediante $\mathcal{P}$.

Teniendo en cuenta la propuesta de Balinski y Laraki [3], un *método de clasificación* puede definirse como una función $F : \mathcal{P} \rightarrow L^n$ que asigna un término $l(x_j) \in L$ a cada alternativa $x_j \in X$ para cada uno de los perfiles y satisface *neutralidad*, *anonimato*, *unanimidad*, *monotonía* e *independencia de alternativas irrelevantes*.

Balinski y Laraki [3] proponen que $l(x_j) \in L$ debe satisfacer:

$$\#\{i \in V \mid a_j^i \geq l(x_j)\} \geq \frac{m}{2}$$

y

$$\#\{i \in V \mid a_j^i \leq l(x_j)\} \geq \frac{m}{2}$$

para cualquier $x_j \in X$. Si $l(x_j)$ cumple estas dos condiciones, los autores sostienen que el grado social no puede ser rechazado por una mayoría absoluta de votantes por ser o bien demasiado alto o bien demasiado bajo.

Es fácil comprobar que siempre existe al menos un término $l(x_j) \in L$ para cada $x_j \in X$ que cumple estas condiciones, si bien no es necesariamente único. Denotamos con $L(x_j)$ al conjunto de elementos que cumplen las condiciones mencionadas. Balinski y Laraki [4] consideran $l(x_j)$ como el mínimo de $L(x_j)$, es decir, la mediana menor de las valoraciones individuales.

Debido a que la granularidad de L no suele ser muy elevada, es fácil que varias alternativas obtengan el mismo grado final. Para deshacer los empates, seguimos el método secuencial usado por Balinski y Laraki en [4] (véase García-Lapresta y Martínez-Panero [9]).

En primer lugar introducimos los conjuntos

$$N^+(x_j) = \#\{i \in V \mid a_j^i > l(x_j)\}$$

$$N^-(x_j) = \#\{i \in V \mid a_j^i < l(x_j)\}$$

y un índice $d(x_j) \in \{-1, 0, 1\}$ para cada alternativa $x_j \in X$ de este modo:

$$d(x_j) = \begin{cases} -1, & \text{si } N^+(x_j) < N^-(x_j), \\ 0, & \text{si } N^+(x_j) = N^-(x_j), \\ 1, & \text{si } N^+(x_j) > N^-(x_j). \end{cases}$$

Teniendo en cuenta las asignaciones colectivas y los índices previos, podemos definir un orden débil² $\succeq$ en X de la siguiente manera: $x_j \succeq x_k$ si y solo si se cumple alguna de las siguientes condiciones:

1. $l(x_j) > l(x_k)$.
2. $l(x_j) = l(x_k)$ y $d(x_j) > d(x_k)$.
3. $l(x_j) = l(x_k)$, $d(x_j) = d(x_k) = 1$ y $N^+(x_j) > N^+(x_k)$.
4. $l(x_j) = l(x_k)$, $d(x_j) = d(x_k) = 1$, $N^+(x_j) = N^+(x_k)$ y $N^-(x_j) \leq N^-(x_k)$.
5. $l(x_j) = l(x_k)$, $d(x_j) = d(x_k) = 0$ y $m - N^+(x_j) - N^-(x_j) \geq m - N^+(x_k) - N^-(x_k)$.

²Un orden débil (o preorden completo) es una relación binaria completa y transitiva.

$$6. \quad l(x_j) = l(x_k), \quad d(x_j) = d(x_k) = -1 \quad \text{y} \\ N^-(x_j) < N^-(x_k).$$

$$7. \quad l(x_j) = l(x_k), \quad d(x_j) = d(x_k) = -1, \\ N^-(x_j) = N^-(x_k) \quad \text{y} \quad N^+(x_j) \geq N^+(x_k).$$

Definimos también:

$$x_j \succ x_k \Leftrightarrow \text{no } x_k \succeq x_j \\ x_j \sim x_k \Leftrightarrow (x_j \succeq x_k \text{ y } x_k \succeq x_j).$$

Para mostrar cómo funciona el procedimiento de decisión recién descrito, presentamos a continuación un ejemplo ilustrativo. Se tendrá en cuenta que, por anonimato, todo perfil queda determinado por

$$N_h(x_j) = \#\{i \in V \mid a_j^i = l_h\}$$

para $h = 1, \dots, g$ y $j = 1, \dots, n$.

Ejemplo 2. Consideremos $X = \{x_1, x_2, x_3, x_4\}$, $V = \{1, \dots, 30\}$, $L = \{l_1, l_2, l_3, l_4\}$ y el perfil incluido en la Tabla 3.

Tabla 3: Esquema del perfil del Ejemplo 2

	$N_1(x_j)$	$N_2(x_j)$	$N_3(x_j)$	$N_4(x_j)$
x_1	5	16	6	3
x_2	4	6	14	6
x_3	3	6	11	10
x_4	6	7	12	5

Si, como hemos comentado previamente, tomamos la mediana para cada una de las alternativas, obtenemos $l(x_1) = l_2$ y $l(x_2) = l(x_3) = l(x_4) = l_3$, por lo que habrá que utilizar el procedimiento de desempate descrito anteriormente entre las alternativas x_2 , x_3 y x_4 . Como $d(x_2) = -1$, $d(x_3) = 1$ y $d(x_4) = -1$, la alternativa ganadora es x_3 . Para determinar la ordenación final debemos seguir con el procedimiento de desempate. Dado que $N^-(x_2) = 10 < 13 = N^-(x_4)$, se obtiene $x_2 \succ x_4$, por lo que la ordenación final resultante será: $x_3 \succ x_2 \succ x_4 \succ x_1$.

3 APLICACIÓN DE MAJORITY JUDGEMENT A LA LICITACIÓN DE PROYECTOS PÚBLICOS

Para aplicar el procedimiento de **MJ** a la licitación de proyectos públicos se adaptará la terminología de **MJ** a nuestro contexto. Esto resulta inmediato si se tiene en cuenta la propuesta de Arrow y Raynaud [2], donde los términos utilizados en la *Elección Social* pueden ser trasladados a los del *Análisis Multicriterio*

y viceversa. Así, $V = \{1, \dots, m\}$ será el conjunto de criterios a valorar y $X = \{x_1, \dots, x_n\}$ el conjunto de proyectos presentados por los licitadores.

3.1 Criterios equiponderados

En primer lugar, se considerará que cada uno de los criterios tiene la misma importancia en la valoración final del proyecto o, lo que es lo mismo, los criterios son equiponderados; además se supondrá que hay un único especialista valorando cada uno de ellos.

Ejemplo 3. Consideremos tres proyectos x_1, x_2 y x_3 valorados en cuatro criterios diferentes: *precio*, *calidad*, *garantía* y *limpieza* con la misma ponderación. La escala cualitativa está formada por 6 términos lingüísticos: $L = \{\text{Muy Mal (MM), Mal (M), Pasable (P), Bien (B), Muy Bien (MB), Excelente (E)}\}$. El perfil de las valoraciones individuales se incluye en la Tabla 4.

Tabla 4: Perfil del Ejemplo 3

	Precio	Calidad	Garantía	Limpieza
x_1	MM	B	B	E
x_2	B	MM	MM	MM
x_3	E	MB	P	P

Si aplicamos **MJ**, obtenemos $l(x_1) = B$, $l(x_2) = MM$ y $l(x_3) = P$, con la ordenación $x_1 \succ x_3 \succ x_2$.

3.2 Criterios con ponderaciones distintas

En la práctica, los pliegos³ de las licitaciones públicas suelen otorgar pesos variables a los criterios, dándose mayor importancia a unos criterios que a otros. Para reflejar los pesos en una escala cualitativa, se propone replicar las valoraciones de cada criterio las veces que sea necesario hasta que tales replications reflejen la ponderación requerida. Este proceso se ilustra a continuación mediante un ejemplo sencillo.

Ejemplo 4. Consideremos ahora el Ejemplo 3, pero con ponderaciones distintas para los cuatro criterios: *precio*, *calidad*, *garantía* y *limpieza* con pesos 40%, 30%, 20% y 10%, respectivamente.

Para aplicar las ponderaciones a los criterios, el primer paso consiste en encontrar la relación entre los pesos de los criterios; en este caso $\text{precio} = 4 \cdot \text{limpieza}$, $\text{calidad} = 3 \cdot \text{limpieza}$ y $\text{garantía} = 2 \cdot \text{limpieza}$. Una vez hallada la relación entre los pesos, replicamos $a_j^i \in L$ un número de veces equivalente a la relación encontrada para el peso del criterio i . Para el proyecto x_1

³Un *pliego* es un memorial donde se exponen las condiciones o cláusulas que se proponen en una concesión.

replicaremos 4 veces MM en *precio*, 3 veces B en *calidad*, 2 veces B en *garantía* y se mantiene E en *limpieza*, y así con todos los proyectos. La Tabla 5 contiene el nuevo perfil con las replications correspondientes a los pesos requeridos, con las valoraciones ordenadas de forma creciente.

Tabla 5: Perfil ordenado con replicación según peso

x_1	MM	MM	MM	MM	B	B	B	B	B	E
x_2	MM	MM	MM	MM	MM	MM	B	B	B	B
x_3	P	P	P	MB	MB	MB	E	E	E	E

Una vez realizado este proceso, al aplicar **MJ** se obtiene $x_3 \succ x_1 \succ x_2$, dado que $l(x_1) = B$, $l(x_2) = MM$ y $l(x_3) = MB$.

Para obtener la relación entre las ponderaciones para los diferentes criterios, si todos los porcentajes son números enteros bastaría hallar el máximo común divisor entre ellos y a continuación dividir los porcentajes por el máximo común divisor.

Ejemplo 5. Para ilustrar el caso recién mencionado, consideramos ahora un contrato real de gestión del servicio público de teleasistencia domiciliaria ([13]) donde los criterios a valorar y sus respectivos pesos son los siguientes:

- Precio ofertado (30%).
- Número de ambulancias (25%).
- Recursos y equipamiento para la atención exclusiva del servicio de la provincia (20%).
- Sistemas para el control del servicio (15%).
- Plan de revisión y mantenimiento de las instalaciones (5%).
- Campaña de divulgación, información y publicidad del servicio (5%).

Para determinar el número mínimo de replications, se halla el máximo común divisor de todos los porcentajes, que en este caso es 5, y el porcentaje de cada criterio se divide por 5: $\frac{30}{5} = 6$, $\frac{25}{5} = 5$, $\frac{20}{5} = 4$, $\frac{15}{5} = 3$ y $\frac{5}{5} = 1$, y se replica cada una de las etiquetas por las cantidades halladas.

En el caso de que los criterios estuvieran ponderados con porcentajes no enteros, bastaría multiplicar por la potencia de 10 correspondiente y, una vez que todos los pesos fueran números enteros, se aplicaría el procedimiento anterior.

3.3 ADJUSTED MAJORITY JUDGEMENT

El procedimiento de **MJ** fue desarrollado para su aplicación en votaciones electorales, donde el número de votantes es muy numeroso; sin embargo, si se utiliza este sistema en electorados pequeños (comités) es probable que $\#L(x_j) > 1$ y, por tanto, la valoración que se utiliza, $l(x_j) = \min L(x_j)$ no sea una valoración colectiva apropiada (García-Lapresta y Martínez-Panero [8]). En el caso de las licitaciones públicas nos encontramos ante el mismo problema, ya que el número de criterios, que equivalen a los votantes en el modelo general, son reducidos.

Ejemplo 6. En la Tabla 6 se incluye un perfil de tres alternativas y seis criterios equiponderados, donde las tres alternativas están igualadas en cuatro de los seis criterios (Excelente en los dos primeros criterios y Muy Mal en los dos últimos), por lo que la ordenación estará determinada por los criterios 3 y 4.

Tabla 6: Perfil del Ejemplo 6

	Criterios					
	1	2	3	4	5	6
x_1	E	E	E	M	MM	MM
x_2	E	E	P	P	MM	MM
x_3	E	E	MB	MM	MM	MM

Aplicando **MJ** obtenemos $l(x_1) = M$, $l(x_2) = P$ y $l(x_3) = MM$, con la ordenación $x_2 \succ x_1 \succ x_3$, la cual no parece satisfactoria puesto que el proyecto x_1 podría considerarse superior a x_2 , ya que posee un Excelente por encima de la mediana, mientras que x_2 tan solo un Pasable.

Para solucionar este y otros problemas presentes en **MJ**, García-Lapresta y Martínez-Panero [8] desarrollan un método denominado *Adjusted Majority Judgement* (**AMJ**) por el cual, en vez de tomar la mediana menor, se considera una valoración colectiva menos sesgada hacia la mediana de las valoraciones individuales. La propuesta consiste en que $l(x_j)$ sea:

- La mediana de $L(x_j)$, si $\#L(x_j)$ es impar
- La mediana de $L(x_j) \setminus \{\max L(x_j)\}$, si $\#L(x_j)$ es par.

El proceso de desempate es el mismo que se seguía en el modelo general.

Al aplicar **AMJ** en el Ejemplo 6, se tiene $l(x_1) = B$ y $l(x_2) = l(x_3) = P$. Mediante el proceso de desempate, se obtiene $d(x_2) = d(x_3) = 0$ y $m - N^+(x_2) -$

$N^-(x_2) = 2 > 0 = m - N^+(x_3) - N^-(x_3)$. Así, el nuevo ranking es $x_1 \succ x_2 \succ x_3$, que proporciona una ordenación más adecuada de los proyectos que la generada por **MJ**.

4 CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha considerado que cada uno de los proyectos públicos presentados a licitación es evaluado por el centro decisor de forma cualitativa en relación a diversos criterios. El procedimiento de agregación utilizado inicialmente está basado en el sistema de votación *Majority Judgement*, recientemente introducido por Balinski y Laraki [3, 4]. Gracias a él se han podido evitar las paradojas del ranking habitualmente presentes en la licitación de proyectos públicos. No obstante, para resolver algunas situaciones indeseables que se dan cuando *Majority Judgement* se utiliza para agregar un número pequeño de juicios cualitativos, se ha considerado también el ajuste propuesto por García-Lapresta y Martínez-Panero en [8].

En primer lugar, por simplicidad, se ha considerado el caso en el que todos los criterios tengan el mismo peso en la valoración de los proyectos. Dado que esta situación es inusual, se ha planteado también el caso en el que los criterios tengan ponderaciones diversas, según la importancia que el organismo desee conceder a cada uno de ellos. Nuestra propuesta consiste en replicar la valoración cualitativa de cada proyecto tantas veces como sea necesario hasta lograr que las proporciones de las replicas sean equivalentes a los pesos asignados a los criterios.

Conviene destacar que la propuesta realizada es puramente ordinal y descansa en la mediana como operador de agregación de términos lingüísticos. A pesar de las buenas propiedades de la mediana como agregador de valoraciones lingüísticas (véase Balinski y Laraki [3]), la mediana no es suficiente para ordenar adecuadamente las alternativas y, en consecuencia, se hace necesario disponer de un procedimiento secuencial de desempate que utilice información adicional (principalmente, número de valoraciones existentes por encima y por debajo de la mediana, tal como apuntan Balinski y Laraki en [4]).

Se ha de tener en cuenta que el planteamiento de este trabajo solo considera valoraciones cualitativas, lo cual es apropiado para juzgar criterios en donde haya vaguedad o incertidumbre. Sin embargo, en la actualidad los proyectos públicos suelen valorarse exclusivamente de forma cuantitativa, independiente de la naturaleza de los criterios. Así, se valoran numéricamente tanto criterios precisos como im-

precisos. Muy posiblemente el uso de este enfoque se deba fundamentalmente a la falta de experiencia de los organismos convocantes de las licitaciones en manejar información cualitativa.

Un próximo objetivo consiste en permitir valoraciones cualitativas y cuantitativas en función de la naturaleza de cada criterio. En este sentido, resulta de interés el tratamiento de Herrera *et al.* [12] sobre la agregación de información heterogénea. Para ello también se cuenta con García-Lapresta y Martínez-Panero [9], en donde se introduce una clase de sistemas de votación que generaliza *Majority Judgement* mediante el uso de operadores OWA centrados (Yager [14]) y 2-tuplas lingüísticas (Herrera y Martínez [11]) con el fin de evitar la excesiva dependencia de la mediana como valor central y su falta de compensación con otras valoraciones colindantes. En Herrera *et al.* [10] puede verse una amplia y reciente revisión de diversos métodos de tratamiento de información lingüística para la toma de decisiones.

Otro aspecto a tener en cuenta es permitir la existencia de varios especialistas que valoren los proyectos, posiblemente con distintos pesos y con diferentes escalas lingüísticas según los criterios. En este sentido resulta de interés el trabajo de de Andrés *et al.* [7].

Agradecimientos

Los autores agradecen a Miguel Martínez Panero sus comentarios, así como la financiación del Ministerio de Ciencia e Innovación (proyectos ECO2009-07332 y ECO2009-12836), de la Consejería de Educación de la Junta de Castilla y León (proyecto VA092A08) y de FEDER.

Referencias

- [1] K.J. Arrow. *Social Choice and Individual Values*. 2nd ed., John Wiley and Sons, Nueva York, 1963.
- [2] K.J. Arrow, H. Raynaud. *Social Choice and Multicriterion Decision-Making*. MIT Press, Cambridge, 1986.
- [3] M. Balinski, R. Laraki. "A theory of measuring, electing and rating", *Proceedings of the National Academy of Sciences of the United States of America* 104, pp. 8720-8725 (2007).
- [4] M. Balinski, R. Laraki. *The Majority Judgement*. <http://ceco.polytechnique.fr/judgement-majoritaire.html>, 2007.
- [5] M. Balinski, R. Laraki. "Election by Majority Judgement: Experimental evidence". *Ecole Polytechnique – Centre National de la Recherche Scientifique*. Cahier 2007-28, 2007.
- [6] T.H. Chen. "An economic approach to public procurement". *Journal of Public Procurement* 8 (3), pp. 407-430, 2008.
- [7] R. de Andrés, J.L. García-Lapresta, L. Martínez. "A multi-granular linguistic model for management decision-making in performance appraisal". *Soft Computing* 14, pp. 21-34, 2010.
- [8] J.L. García-Lapresta, M. Martínez-Panero. "Adjusting Majority Judgement to small electorates", *9th International Meeting of the Society for Social Choice and Welfare*. Montreal (Canada), 2008.
- [9] J.L. García-Lapresta, M. Martínez-Panero. "Linguistic-based voting through centered OWA operators". *Fuzzy Optimization and Decision Making*, en prensa, doi: 10.1007/s10700-009-9067-0.
- [10] F. Herrera, S. Alonso, F. Chiclana, E. Herrera-Viedma. "Computing with words in decision making: foundations, trends and prospects". *Fuzzy Optimization and Decision Making*, en prensa, doi: 10.1007/s10700-009-9065-2.
- [11] F. Herrera, L. Martínez. "A 2-tuple fuzzy linguistic representation model for computing with words". *IEEE Transactions on Fuzzy Systems* 8, pp. 746-752, 2000.
- [12] F. Herrera, L. Martínez, P.J. Sánchez. "Managing non-homogeneous information in group decision making". *European Journal of Operational Research* 166, pp. 115-132, 2005.
- [13] "Pliego de cláusulas administrativas particulares y prescripciones técnicas particulares para contratar mediante procedimiento abierto, tramitación ordinaria y concesión la gestión del servicio público de teleasistencia domiciliaria en la provincia de Palencia", 2008.
- [14] R.R. Yager. "Centered OWA operators". *Soft Computing* 11, pp. 631-639, 2007.

DECISIONES MAYORITARIAS CON UMBRAL DE APOYO. ANÁLISIS DE LA COHERENCIA COLECTIVA

José Luis García Lapresta¹ Bonifacio Llamazares² Patrizia Pérez Asurmendi³

¹ PRESAD Research Group, Universidad de Valladolid, lapresta@eco.uva.es

² PRESAD Research Group, Universidad de Valladolid, boni@eco.uva.es

³ PRESAD Research Group, SEED Research Group, Universidad Rey Juan Carlos, patrizia.perez@urjc.es

Resumen

En este trabajo se analiza la conexión entre la racionalidad individual y colectiva en el ámbito de las mayorías con umbral de apoyo. Estas reglas de decisión permiten a los votantes graduar sus preferencias entre las alternativas, y declaran ganadora una alternativa sobre otra si la intensidad colectiva de preferencia es superior en un umbral previamente fijado. Los resultados proporcionados determinan los umbrales de apoyo que impiden la aparición de inconsistencias para tres casos de transitividad en las preferencias individuales.

Palabras Clave: Preferencias recíprocas, Mayoría simple, Mayorías por diferencia de votos, Mayorías con umbral de apoyo, Paradoja del voto, Aciclicidad.

1 INTRODUCCIÓN

Dadas dos alternativas, la regla de votación de la *mayoría simple* declara ganadora a una alternativa con el único requisito de que tenga más votos que la otra. Así, puede ocurrir que una alternativa resulte vencedora con muy poco apoyo si hay gran abstención. Con objeto de evitar este aspecto indeseable, han sido consideradas en la literatura otro tipo de mayorías que requieren mayor apoyo para declarar vencedora a una alternativa (véanse, por ejemplo, Fishburn [2, chapter 6], Ferejohn y Grether [1], Saari [6, pp. 122-123], García-Lapresta y Llamazares [3] y Llamazares [5]).

En García-Lapresta y Llamazares [3] se introducen las *mayorías por diferencia de votos*, las cuales declaran ganadora a una alternativa cuando el número de votos conseguido es mayor que el de la otra más un um-

bral previamente fijado. Dichas mayorías oscilan entre la mayoría simple y la unanimidad y, en consecuencia, pueden utilizarse en diferentes contextos, con tan solo ajustar el umbral al nivel deseado. La caracterización axiomática de este tipo de mayorías dada por Llamazares [5] en el ámbito de la Teoría de la Elección Social refuerza claramente su interés, debido a sus buenas propiedades.

Cuando se utilizan las reglas de votación por mayoría simple, absoluta, cualificadas, por diferencia de votos, etc. fuera de su escenario habitual de dos alternativas, y se amplía a tres o más, surge el problema de la posible aparición de inconsistencias. El ejemplo más conocido de inconsistencia es el de la *paradoja del voto*, denunciado por Condorcet en 1781. Dicha paradoja aparece, por ejemplo, cuando tres votantes ordenan tres alternativas de la siguiente forma: $x \succ y \succ z$, $y \succ z \succ x$, $z \succ x \succ y$. Entonces, al aplicar la reglas de votación de la mayoría simple o de la mayoría absoluta surge un ciclo: x vence a y , y vence a z , pero z también vence a x .

En García-Lapresta y Llamazares [4] se extienden las mayorías por diferencia de votos al escenario en el que los votantes declaran la intensidad de sus preferencias entre las alternativas. En este caso el umbral de apoyo pasa a ser un número real entre 0 y el número de votantes. Así, a la flexibilidad que supone permitir a los votantes graduar sus preferencias se une la posibilidad de fijar umbrales de apoyo en un espectro continuo. La caracterización axiomática de este tipo de mayorías dada en [4] pone de manifiesto sus buenas propiedades desde la perspectiva de la Teoría de la Elección Social. De nuevo, al extender el uso de este tipo de mayorías a más de dos alternativas surge el problema de las inconsistencias en las preferencias colectivas. A esto hay que añadir que las mayorías con umbral de apoyo pueden generar inconsistencias cuando hay únicamente dos votantes, algo que no sucede en las mayorías simple, absoluta, cualificadas ni por diferencia de votos.

Las mayorías con umbral de apoyo definen una relación de preferencia colectiva ordinaria a partir de las intensidades de preferencia individuales. El objetivo de este trabajo consiste en determinar el grado de consistencia en las preferencias colectivas en función de la consistencia de las preferencias individuales. En el marco de las intensidades de preferencia individuales se han considerado tres modelos representativos de racionalidad basados en la transitividad. Para la consistencia colectiva se han tenido en cuenta únicamente dos modelos de racionalidad: la triple-aciclicidad, que evita la aparición de ciclos entre tres alternativas, y la aciclicidad, que impide cualquier tipo de ciclos.

El trabajo se organiza como sigue. La sección 2 está dedicada a introducir los conceptos básicos, la sección 3 contiene los resultados para el caso en el que solo hay dos individuos, la sección 4 analiza el caso en el que hay tres individuos y, por último, la sección 5 muestra el trabajo futuro a realizar.

2 PRELIMINARES

Consideraremos m agentes que muestran la intensidad de sus preferencias sobre un conjunto de alternativas $X = \{x_1, \dots, x_n\}$ mediante relaciones de preferencia recíprocas $R^p : X \times X \rightarrow [0, 1]$, $p = 1, \dots, m$; es decir, $r_{ij}^p + r_{ji}^p = 1$ para cualesquiera $i, j \in \{1, \dots, n\}$, donde $r_{ij}^p = R^p(x_i, x_j)$. Con $\mathcal{R}(X)$ se denota el conjunto de relaciones de preferencia recíprocas sobre X .

Un *perfil* es un vector $(R_1, \dots, R_m)$ que contiene las relaciones recíprocas individuales. El conjunto de perfiles vendrá denotado por $\mathcal{R}^m(X)$.

Una *relación de preferencia ordinaria* sobre X es una relación binaria asimétrica P sobre X : si $x_i P x_j$, entonces no ocurre $x_j P x_i$. Con $\mathcal{P}(X)$ denotaremos el conjunto de relaciones de preferencia ordinarias sobre X . $P \in \mathcal{P}(X)$ es *acíclica* si para cualesquiera $x_{i_1}, \dots, x_{i_r} \in X$ se verifica que si ocurre $x_{i_1} P x_{i_2}, \dots, x_{i_{r-1}} P x_{i_r}$, entonces no ocurre $x_{i_r} P x_{i_1}$. $P \in \mathcal{P}(X)$ es *triple-acíclica* si para cualesquiera $x_i, x_j, x_l \in X$ se verifica que si ocurre $x_i P x_j$ y $x_j P x_l$, entonces no ocurre $x_l P x_i$. Obviamente, toda relación de preferencia acíclica es triple-acíclica. Por otra parte, para $n = 3$ los conceptos de aciclicidad y triple-aciclicidad coinciden.

2.1 Consistencia en preferencias recíprocas

Definición 1. Un operador monótono es una función continua $g : [0.5, 1]^2 \rightarrow [0.5, 1]$ que satisface las siguientes propiedades:

1. *Monotonía.* $g(a, b) \geq g(a', b')$ para cualesquiera $(a, b), (a', b') \in [0.5, 1]^2$ tales que $a \geq a'$ y $b \geq b'$.

2. *Simetría.* $g(a, b) = g(b, a)$ para cualesquiera $a, b \in [0.5, 1]$.

Definición 2. Sean $R \in \mathcal{R}(X)$ y g un operador monótono. R es g -transitiva si para cualesquiera $i, j, l \in \{1, \dots, n\}$ se verifica

$$r_{ij} > 0.5 \text{ y } r_{jl} > 0.5 \Rightarrow r_{il} > 0.5 \text{ y } r_{il} \geq g(r_{ij}, r_{jl}).$$

Denotaremos por T_g al conjunto de relaciones de preferencia recíprocas g -transitivas; es decir,

$$T_g = \{R \in \mathcal{R}(X) \mid R \text{ es } g\text{-transitiva}\}.$$

Es importante destacar que si f y g son dos operadores monótonos, R es g -transitiva y $g(a, b) \geq f(a, b)$ para todo $(a, b) \in [0.5, 1]^2$, entonces R es f -transitiva; es decir, $T_g \subseteq T_f$.

En este trabajo se considerarán tres operadores monótonos: los correspondientes a la función constante 0.5, al mínimo y al máximo:

1. R es *0.5-transitiva* si R es g -transitiva, con $g(a, b) = 0.5$ para todo $(a, b) \in [0.5, 1]^2$:

$$r_{ij} > 0.5 \text{ y } r_{jl} > 0.5 \Rightarrow r_{il} > 0.5$$

para cualesquiera $i, j, l \in \{1, \dots, n\}$.

2. R es *min-transitiva* si R es g -transitiva, con $g(a, b) = \min\{a, b\}$ para todo $(a, b) \in [0.5, 1]^2$:

$$r_{ij} > 0.5 \text{ y } r_{jl} > 0.5 \Rightarrow r_{il} \geq \min\{r_{ij}, r_{jl}\}$$

para cualesquiera $i, j, l \in \{1, \dots, n\}$.

3. R es *max-transitiva* si R es g -transitiva, con $g(a, b) = \max\{a, b\}$ para todo $(a, b) \in [0.5, 1]^2$:

$$r_{ij} > 0.5 \text{ y } r_{jl} > 0.5 \Rightarrow r_{il} \geq \max\{r_{ij}, r_{jl}\}$$

para cualesquiera $i, j, l \in \{1, \dots, n\}$.

Los conjuntos de relaciones de preferencia recíprocas 0.5-transitivas, min-transitivas y max-transitivas se denotarán por $T_{0.5}$, $T_{\min}$ y $T_{\max}$, respectivamente.

Puesto que $\max\{a, b\} \geq \min\{a, b\} \geq 0.5$ para cualesquiera $a, b \in [0.5, 1]$, se tiene $T_{\max} \subseteq T_{\min} \subseteq T_{0.5}$.

Además, las relaciones de preferencia recíprocas min-transitivas (y, por tanto, las max-transitivas) cumplen la siguiente propiedad.

Lema 1. Si $R \in T_{\min}$, entonces $r_{ij} + r_{jl} + r_{li} \leq 2$ para cualesquiera $i, j, l \in \{1, \dots, n\}$.

DEMOSTRACIÓN

Supongamos, por reducción al absurdo, que existen $i, j, l \in \{1, \dots, n\}$ tales que $r_{ij} + r_{jl} + r_{li} > 2$. De aquí se deduce que dos de los tres sumandos son mayores que 0.5 (si lo fueran los tres la relación no sería min-transitiva). Supongamos $r_{ij}, r_{jl} > 0.5$. Entonces $r_{il} \geq \min\{r_{ij}, r_{jl}\}$ o, equivalentemente, $r_{li} \leq \max\{r_{ji}, r_{lj}\} = \max\{1 - r_{ij}, 1 - r_{jl}\}$. Por tanto,

$$r_{ij} + r_{jl} + r_{li} \leq r_{ij} + r_{jl} + \max\{1 - r_{ij}, 1 - r_{jl}\} \leq 2,$$

lo cual contradice $r_{ij} + r_{jl} + r_{li} > 2$. Los otros dos casos posibles, $r_{ij}, r_{li} > 0.5$ y $r_{jl}, r_{li} > 0.5$, también llevan a contradicción cuando se realiza un análisis similar al efectuado para el caso $r_{ij}, r_{jl} > 0.5$. ■

2.2 Mayorías con umbral de apoyo

Las mayorías con umbral de apoyo fueron introducidas y caracterizadas axiomáticamente en el ámbito de la Teoría de la Elección Social por García-Lapresta y Llamazares [4]. Constituyen una extensión de las mayorías por diferencia de votos (García-Lapresta y Llamazares [3], Llamazares [5]) al permitir que los votantes puedan mostrar la intensidad de sus preferencias entre las alternativas y que los umbrales de apoyo varíen en un espectro continuo.

Definición 3. Sean $k \in [0, m)$ y $D \subseteq \mathcal{R}^m(X)$. La mayoría con umbral de apoyo k , F_k , es la aplicación $F_k : D \rightarrow \mathcal{P}(X)$ definida por $F_k(R^1, \dots, R^m) = P_k$, donde

$$x_i P_k x_j \Leftrightarrow \sum_{p=1}^m r_{ij}^p > \sum_{p=1}^m r_{ji}^p + k.$$

Resulta inmediato comprobar que la mayoría F_k puede definirse mediante la media aritmética de las intensidades de preferencia individuales:

$$x_i P_k x_j \Leftrightarrow \frac{1}{m} \sum_{p=1}^m r_{ij}^p > 0.5 + \frac{k}{2m}.$$

La relación de indiferencia asociada a P_k viene definida por

$$x_i I_k x_j \Leftrightarrow \left| \sum_{p=1}^m r_{ij}^p - \sum_{p=1}^m r_{ji}^p \right| \leq k,$$

o, equivalentemente,

$$x_i I_k x_j \Leftrightarrow \left| \frac{1}{m} \sum_{p=1}^m r_{ij}^p - 0.5 \right| \leq \frac{k}{2m}.$$

3 Aciclicidad con dos individuos

En la mayoría simple, cuando solo intervienen dos votantes cuyas preferencias son transitivas, las preferencias colectivas son también transitivas y, por tanto, no aparece la paradoja del voto. Sin embargo, en las mayorías con umbral de apoyo sí puede darse este fenómeno indeseable. Sin duda, es un precio a pagar por disponer de mayor flexibilidad.

En el primer resultado se proporcionan condiciones necesarias para que las relaciones de preferencia colectivas asociadas a las mayorías con umbral de apoyo sean acíclicas cuando las relaciones de preferencia recíprocas de los dos individuos son 0.5-transitivas, min-transitivas y max-transitivas.

Proposición 1. Si $m = 2$ y $F_k : D \rightarrow \mathcal{P}(X)$ es tal que P_k es acíclica para todo $(R^1, R^2) \in D \subseteq \mathcal{R}^2(X)$, entonces

1. $k \in [1, 2)$ para $D = T_{0.5}^2$.
2. $k \in [\frac{2}{3}, 2)$ para $D = T_{\min}^2$.
3. $k \in [0.5, 2)$ para $D = T_{\max}^2$.

DEMOSTRACIÓN

1. Demostraremos que si $k < 1$, entonces P_k no es acíclica. Consideremos dos individuos con las siguientes relaciones de preferencia recíprocas:

$$R^1 = \begin{pmatrix} 0.5 & 1 & \frac{3-k}{4} & \dots \\ 0 & 0.5 & 1 & \dots \\ \frac{1+k}{4} & 0 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

$$R^2 = \begin{pmatrix} 0.5 & \frac{1+k}{4} & 0 & \dots \\ \frac{3-k}{4} & 0.5 & 1 & \dots \\ 1 & 0 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

donde los elementos de las matrices no dados explícitamente valen 0.5. Resulta inmediato comprobar que $R^1, R^2 \in T_{0.5}$. Puesto que

$$x_i P_k x_j \Leftrightarrow r_{ij}^1 + r_{ij}^2 > 1 + \frac{k}{2},$$

se tiene $x_1 P_k x_2$, $x_2 P_k x_3$ y $x_3 P_k x_1$. Por tanto, P_k no es acíclica.

2. Demostraremos que si $k < \frac{2}{3}$, entonces P_k no es acíclica. Consideremos dos individuos con las siguientes relaciones de preferencia recíprocas:

$$R^1 = \begin{pmatrix} 0.5 & 1 & \frac{2}{3} & \dots \\ 0 & 0.5 & \frac{2}{3} & \dots \\ \frac{1}{3} & \frac{1}{3} & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

$$R^2 = \begin{pmatrix} 0.5 & \frac{1}{3} & 0 & \cdots \\ \frac{2}{3} & 0.5 & \frac{2}{3} & \cdots \\ 1 & \frac{1}{3} & 0.5 & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{pmatrix},$$

donde los elementos de las matrices no dados explícitamente valen 0.5. Resulta inmediato comprobar que $R^1, R^2 \in T_{\min}$. Puesto que

$$x_i P_k x_j \Leftrightarrow r_{ij}^1 + r_{ij}^2 > 1 + \frac{k}{2},$$

se tiene $x_1 P_k x_2$, $x_2 P_k x_3$ y $x_3 P_k x_1$. Por tanto, P_k no es acíclica.

3. Demostraremos que si $k < 0.5$, entonces P_k no es acíclica. Consideremos dos individuos con las siguientes relaciones de preferencia recíprocas:

$$R^1 = \begin{pmatrix} 0.5 & 0.75 & 0.75 & \cdots \\ 0.25 & 0.5 & 0.75 & \cdots \\ 0.25 & 0.25 & 0.5 & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{pmatrix},$$

$$R^2 = \begin{pmatrix} 0.5 & 0.5 & 0 & \cdots \\ 0.5 & 0.5 & 0.5 & \cdots \\ 1 & 0.5 & 0.5 & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{pmatrix},$$

donde los elementos de las matrices no dados explícitamente valen 0.5. Resulta inmediato comprobar que $R^1, R^2 \in T_{\max}$. Puesto que

$$x_i P_k x_j \Leftrightarrow r_{ij}^1 + r_{ij}^2 > 1 + \frac{k}{2},$$

se tiene $x_1 P_k x_2$, $x_2 P_k x_3$ y $x_3 P_k x_1$. Por tanto, P_k no es acíclica. ■

Las condiciones necesarias de la proposición anterior son también suficientes si se sustituye aciclicidad por triple-aciclicidad, tal como se demuestra en los siguientes resultados.

Proposición 2. Para $m = 2$ y $F_k : T_{0.5}^2 \rightarrow \mathcal{P}(X)$, si $k \in [1, 2)$, entonces P_k es triple-acíclica para todo $(R^1, R^2) \in T_{0.5}^2$.

DEMOSTRACIÓN

Supongamos, por reducción al absurdo, que P_k no es triple-acíclica. Entonces, existen $R^1, R^2 \in T_{0.5}$, $i, j, l \in \{1, \dots, n\}$ tales que $x_i P_k x_j$, $x_j P_k x_l$ y $x_l P_k x_i$. Por tanto,

$$r_{ij}^1 + r_{ij}^2 > 1 + \frac{k}{2} \geq 1.5,$$

$$r_{jl}^1 + r_{jl}^2 > 1 + \frac{k}{2} \geq 1.5,$$

$$r_{li}^1 + r_{li}^2 > 1 + \frac{k}{2} \geq 1.5.$$

De las condiciones anteriores se deduce $r_{ij}^1 > 0.5$, $r_{jl}^1 > 0.5$ y $r_{li}^1 > 0.5$. Como $R^1 \in T_{0.5}$, se tiene $r_{il}^1 > 0.5$, lo cual contradice $r_{li}^1 > 0.5$. ■

Proposición 3. Para $m = 2$ y $F_k : T_{\min}^2 \rightarrow \mathcal{P}(X)$, si $k \in [\frac{2}{3}, 2)$, entonces P_k es triple-acíclica para todo $(R^1, R^2) \in T_{\min}^2$.

DEMOSTRACIÓN

Supongamos, por reducción al absurdo, que P_k no es triple-acíclica. Entonces, existen $R^1, R^2 \in T_{\min}$, $i, j, l \in \{1, \dots, n\}$ tales que $x_i P_k x_j$, $x_j P_k x_l$ y $x_l P_k x_i$; es decir,

$$r_{ij}^1 + r_{ij}^2 > 1 + \frac{k}{2} \geq \frac{4}{3},$$

$$r_{jl}^1 + r_{jl}^2 > 1 + \frac{k}{2} \geq \frac{4}{3},$$

$$r_{li}^1 + r_{li}^2 > 1 + \frac{k}{2} \geq \frac{4}{3}.$$

Por tanto, $r_{ij}^1 + r_{ij}^2 + r_{jl}^1 + r_{jl}^2 + r_{li}^1 + r_{li}^2 > 4$, lo cual es absurdo pues $R^1, R^2 \in T_{\min}$ y, por el Lema 1, $r_{ij}^1 + r_{jl}^1 + r_{li}^1 + r_{ij}^2 + r_{jl}^2 + r_{li}^2 \leq 2 + 2 = 4$. ■

Proposición 4. Para $m = 2$ y $F_k : T_{\max}^2 \rightarrow \mathcal{P}(X)$, si $k \in [0.5, 2)$, entonces P_k es triple-acíclica para todo $(R^1, R^2) \in T_{\max}^2$.

DEMOSTRACIÓN

Supongamos, por reducción al absurdo, que P_k no es triple-acíclica. Entonces, existen $R^1, R^2 \in T_{\max}$, $i, j, l \in \{1, \dots, n\}$ tales que $x_i P_k x_j$, $x_j P_k x_l$ y $x_l P_k x_i$. Por tanto,

$$r_{ij}^1 + r_{ij}^2 > 1 + \frac{k}{2} \geq 1.25,$$

$$r_{jl}^1 + r_{jl}^2 > 1 + \frac{k}{2} \geq 1.25,$$

$$r_{li}^1 + r_{li}^2 > 1 + \frac{k}{2} \geq 1.25.$$

Es importante destacar que la última condición es equivalente a $r_{il}^1 + r_{il}^2 < 0.75$. Vamos a distinguir tres casos en función del número de individuos a los que se puede aplicar la max-transitividad sobre los pares de alternativas (x_i, x_j) y (x_j, x_l) ; es decir, dependiendo del cardinal del siguiente conjunto:

$$\mathcal{P} = \left\{ p \in \{1, 2\} \mid \min\{r_{ij}^p, r_{jl}^p\} > 0.5 \right\}.$$

1. Si $\#\mathcal{P}=2$, entonces $r_{il}^p \geq \max\{r_{ij}^p, r_{jl}^p\}$ para todo $p \in \{1, 2\}$. Por tanto, $r_{il}^1 + r_{il}^2 \geq r_{ij}^1 + r_{ij}^2 > 1.25$, lo cual contradice $r_{il}^1 + r_{il}^2 < 0.75$.

2. Si $\#P=1$, podemos suponer, sin pérdida de generalidad, que $P = \{1\}$. Por la max-transitividad se cumple $r_{il}^1 \geq \max\{r_{ij}^1, r_{jl}^1\} > 0.5$. Como $r_{il}^1 + r_{il}^2 < 0.75$, se tiene $r_{il}^2 < 0.25$ y $\max\{r_{ij}^1, r_{jl}^1\} < 0.75 - r_{il}^2$. Distinguimos dos casos:

- Si $r_{ij}^2 \leq 0.5$, entonces $0.75 - r_{il}^2 > r_{ij}^1 > 1.25 - r_{ij}^2$; es decir, $r_{ij}^2 - r_{il}^2 > 0.5$, lo cual es absurdo pues $r_{ij}^2, r_{il}^2 \in [0, 0.5]$.
- Si $r_{ij}^2 > 0.5$, entonces, dado que $P = \{1\}$, se tiene $r_{jl}^2 \leq 0.5$. En este caso, $0.75 - r_{il}^2 > r_{jl}^1 > 1.25 - r_{jl}^2$; es decir, $r_{jl}^2 - r_{il}^2 > 0.5$, lo cual es absurdo pues $r_{jl}^2, r_{il}^2 \in [0, 0.5]$.

3. Si $\#P=0$, como $r_{ij}^1 + r_{ij}^2 > 1.25$ y $r_{jl}^1 + r_{jl}^2 > 1.25$, podemos suponer, sin pérdida de generalidad, que $r_{ij}^1 > 0.5$, $r_{jl}^1 \leq 0.5$, $r_{ij}^2 \leq 0.5$ y $r_{jl}^2 > 0.5$. Dado que $r_{li}^1 + r_{li}^2 > 1.25$, distinguimos tres casos:

- Si $r_{li}^1 > 0.5$ y $r_{li}^2 > 0.5$, entonces, como $R^1, R^2 \in T_{\max}$, se tiene $r_{lj}^1 \geq \max\{r_{li}^1, r_{ij}^1\}$ y $r_{ji}^2 \geq \max\{r_{jl}^2, r_{li}^2\}$. La primera condición es equivalente a $r_{jl}^1 \leq \min\{r_{ji}^1, r_{il}^1\}$. Por tanto,

$$r_{ji}^1 + r_{ji}^2 \geq r_{jl}^1 + r_{jl}^2 > 1.25,$$

lo cual es absurdo, pues $r_{ij}^1 + r_{ij}^2 > 1.25$.

- Si $r_{li}^1 > 0.5$ y $r_{li}^2 \leq 0.5$, entonces, como $R^1 \in T_{\max}$, se tiene $r_{lj}^1 \geq \max\{r_{li}^1, r_{ij}^1\}$, lo cual es equivalente a $r_{jl}^1 \leq \min\{r_{ji}^1, r_{il}^1\}$. Por tanto,

$$\begin{aligned} r_{jl}^2 - r_{il}^2 &\geq (r_{jl}^1 + r_{jl}^2) - (r_{il}^1 + r_{il}^2) \\ &> 1.25 - 0.75 = 0.5, \end{aligned}$$

lo cual es absurdo pues $r_{ij}^2, r_{il}^2 \in [0.5, 1]$.

- Si $r_{li}^1 \leq 0.5$, entonces, dado que $r_{li}^1 + r_{li}^2 > 1.25$, se tiene $r_{li}^2 > 0.75$. Como $R^2 \in T_{\max}$, entonces $r_{ji}^2 \geq \max\{r_{jl}^2, r_{li}^2\}$, lo cual es equivalente a $r_{il}^2 \leq \min\{r_{jl}^2, r_{li}^2\}$. Por tanto,

$$\begin{aligned} r_{ij}^1 - r_{il}^1 &\geq (r_{ij}^1 + r_{ij}^2) - (r_{il}^1 + r_{il}^2) \\ &> 1.25 - 0.75 = 0.5, \end{aligned}$$

lo cual es absurdo pues $r_{ij}^1, r_{il}^1 \in [0.5, 1]$. ■

4 Aciclicidad con tres individuos

Al igual que en la sección anterior, en primer lugar se proporcionan condiciones necesarias para que las relaciones de preferencia colectivas asociadas a las mayorías con umbral de apoyo sean acíclicas.

Proposición 5. Si $m = 3$ y $F_k : D \longrightarrow \mathcal{P}(X)$ es tal que P_k es acíclica para todo $(R^1, R^2, R^3) \in D \subseteq R^3(X)$, entonces

1. $k \in [2, 3)$ para $D = T_{0.5}^3$.

2. $k \in [1, 3)$ para $D = T_{\min}^3$ o $D = T_{\max}^3$.

DEMOSTRACIÓN

1. Demostraremos que si $k < 2$, entonces P_k no es acíclica. Consideremos tres individuos con las siguientes relaciones de preferencia recíprocas:

$$R^1 = \begin{pmatrix} 0.5 & 1 & \frac{5-k}{6} & \dots \\ 0 & 0.5 & 1 & \dots \\ \frac{1+k}{6} & 0 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

$$R^2 = \begin{pmatrix} 0.5 & 1 & 0 & \dots \\ 0 & 0.5 & \frac{1+k}{6} & \dots \\ 1 & \frac{5-k}{6} & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

$$R^3 = \begin{pmatrix} 0.5 & \frac{1+k}{6} & 0 & \dots \\ \frac{5-k}{6} & 0.5 & 1 & \dots \\ 1 & 0 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

donde los elementos de las matrices no dados explícitamente valen 0.5. Resulta inmediato comprobar $R^1, R^2, R^3 \in T_{0.5}$. Puesto que

$$x_i P_k x_j \Leftrightarrow r_{ij}^1 + r_{ij}^2 + r_{ij}^3 > 1.5 + \frac{k}{2},$$

se tiene $x_1 P_k x_2$, $x_2 P_k x_3$ y $x_3 P_k x_1$. Por tanto, P_k no es acíclica.

2. Demostraremos que si $k < 1$, entonces P_k no es acíclica. Consideremos tres individuos con las siguientes relaciones de preferencia recíprocas:

$$R^1 = \begin{pmatrix} 0.5 & 1 & 1 & \dots \\ 0 & 0.5 & 1 & \dots \\ 0 & 0 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

$$R^2 = \begin{pmatrix} 0.5 & 1 & 0 & \dots \\ 0 & 0.5 & 0 & \dots \\ 1 & 1 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

$$R^3 = \begin{pmatrix} 0.5 & 0 & 0 & \dots \\ 1 & 0.5 & 1 & \dots \\ 1 & 0 & 0.5 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix},$$

donde los elementos de las matrices no dados explícitamente valen 0.5. Resulta inmediato comprobar $R^1, R^2, R^3 \in T_{\max} \subseteq T_{\min}$. Puesto que

$$x_i P_k x_j \Leftrightarrow r_{ij}^1 + r_{ij}^2 + r_{ij}^3 > 1.5 + \frac{k}{2},$$

se tiene $x_1 P_k x_2$, $x_2 P_k x_3$ y $x_3 P_k x_1$. Por tanto, P_k no es acíclica. ■

Las condiciones necesarias de la proposición anterior son también suficientes si se sustituye aciclicidad por triple-aciclicidad, tal como se demuestra en los siguientes resultados.

Proposición 6. Para $m = 3$ y $F_k : T_{0.5}^3 \rightarrow \mathcal{P}(X)$, si $k \in [2, 3)$, entonces P_k es triple-acíclica para todo $(R^1, R^2, R^3) \in T_{0.5}^3$.

DEMOSTRACIÓN

Supongamos, por reducción al absurdo, que P_k no es triple-acíclica. Entonces, existen $R^1, R^2, R^3 \in T_{0.5}$, $i, j, l \in \{1, \dots, n\}$ tales que $x_i P_k x_j$, $x_j P_k x_l$ y $x_l P_k x_i$. Por tanto,

$$r_{ij}^1 + r_{ij}^2 + r_{ij}^3 > 1.5 + \frac{k}{2} \geq 2.5,$$

$$r_{jl}^1 + r_{jl}^2 + r_{jl}^3 > 1.5 + \frac{k}{2} \geq 2.5,$$

$$r_{li}^1 + r_{li}^2 + r_{li}^3 > 1.5 + \frac{k}{2} \geq 2.5.$$

De las condiciones anteriores se deduce $r_{ij}^1 > 0.5$, $r_{jl}^1 > 0.5$ y $r_{li}^1 > 0.5$. Como $R^1 \in T_{0.5}$, se tiene $r_{il}^1 > 0.5$, lo cual contradice $r_{li}^1 > 0.5$. ■

Proposición 7. Para $m = 3$ y $F_k : T_{\min}^3 \rightarrow \mathcal{P}(X)$ o $F_k : T_{\max}^3 \rightarrow \mathcal{P}(X)$, si $k \in [1, 3)$, entonces P_k es triple-acíclica para todo $(R^1, R^2, R^3) \in T_{\min}^3$.

DEMOSTRACIÓN

Sea $F_k : T_{\min}^3 \rightarrow \mathcal{P}(X)$ (el caso $F_k : T_{\max}^3 \rightarrow \mathcal{P}(X)$ se demuestra de forma análoga). Supongamos, por reducción al absurdo, que P_k no es triple-acíclica. Entonces, existen $R^1, R^2, R^3 \in T_{\min}$, $i, j, l \in \{1, \dots, n\}$ tales que $x_i P_k x_j$, $x_j P_k x_l$ y $x_l P_k x_i$; es decir,

$$r_{ij}^1 + r_{ij}^2 + r_{ij}^3 > 1.5 + \frac{k}{2} \geq 2,$$

$$r_{jl}^1 + r_{jl}^2 + r_{jl}^3 > 1.5 + \frac{k}{2} \geq 2,$$

$$r_{li}^1 + r_{li}^2 + r_{li}^3 > 1.5 + \frac{k}{2} \geq 2.$$

Por tanto,

$$\sum_{p=1}^3 r_{ij}^p + \sum_{p=1}^3 r_{jl}^p + \sum_{p=1}^3 r_{li}^p > 6,$$

lo cual es absurdo pues $R^1, R^2, R^3 \in T_{\min}$ y, por el Lema 1,

$$\sum_{p=1}^3 (r_{ij}^p + r_{jl}^p + r_{li}^p) \leq 2 + 2 + 2 = 6. \quad \blacksquare$$

A la vista de los resultados obtenidos, cuanto mayor sea el nivel de coherencia en las preferencias individuales, más bajo es el umbral que garantiza la coherencia en la preferencia colectiva.

5 Trabajo futuro

En el presente trabajo se ha considerado únicamente la presencia de dos o tres votantes. Queda pendiente, por tanto, la tarea de analizar el problema cuando hay un número mayor de individuos. Por otra parte, solo se han analizado dos condiciones básicas de racionalidad colectiva: la triple-aciclicidad y la aciclicidad. En el futuro se analizará cuándo se puede garantizar la transitividad de las preferencias colectivas al aplicar las mayorías con umbral de apoyo.

Agradecimientos

Los autores agradecen la financiación del Ministerio de Ciencia e Innovación (proyectos ECO2009-07332 y ECO2009-12836), de la Consejería de Educación de la Junta de Castilla y León (proyecto VA092A08) y de FEDER.

Referencias

- [1] J.A. Ferejohn, D.M. Grether. "On a class of rational social decisions procedures". *Journal of Economic Theory* 8, pp. 471-482, 1974.
- [2] P.C. Fishburn. *The Theory of Social Choice*. Princeton University Press, Princeton, 1973.
- [3] J.L. García-Lapresta, B. Llamazares. "Majority decisions based on difference of votes". *Journal of Mathematical Economics* 35, pp. 463-481, 2001.
- [4] J.L. García-Lapresta, B. Llamazares. "Preference intensities and majority decisions based on difference of support between alternatives". *Group Decision and Negotiation*, en prensa, doi 10.1007/s10726-009-9156-z.
- [5] B. Llamazares. "The forgotten decision rules: Majority rules based on difference of votes". *Mathematical Social Sciences* 51, pp. 311-326, 2006.
- [6] D.G. Saari. "Consistency of decision processes". *Annals of Operations Research* 23, pp. 103-137, 1990.

TOMA DE DECISIÓN LINGÜÍSTICA. MODELOS COMPUTACIONALES SIMBÓLICOS Y SU APLICACIÓN EN EL TRATAMIENTO DE PERCEPCIONES

R.M. Rodríguez¹ L. Martínez¹ F. Herrera²

¹Departamento de Informática, Universidad de Jaén, 23071 (Jaén) {rmrodrig,martin}@ujaen.es

²Departamento de CCIA, Universidad de Granada, 18071 (Granada) {herrera}@decsai.ugr.es

Resumen

La Toma de Decisiones aparece en distintas áreas y contextos del mundo real. El interés de esta contribución se centra en contextos vagos y con incertidumbre, donde la información puede ser imprecisa o basada en percepciones. En estos casos la información lingüística modela de forma flexible este conocimiento, aunque implica procesos de computación con palabras. Existen distintos modelos computacionales para operar lingüísticamente. Nuestro objetivo es hacer un estudio de los mismos en su aplicación en problemas de Toma de Decisión Lingüística y sus implicaciones en computación con percepciones.

Palabras Clave: Toma de decisión, información lingüística, computación con palabras, computación con percepciones.

1 INTRODUCCIÓN

La Toma de Decisión es un proceso habitual en muchas actividades del mundo real como, la ingeniería, las finanzas, la psicología, etc. Esta heterogeneidad implica que dichos problemas puedan definirse en distintos contextos. Aquí nos centraremos en problemas de decisión definidos bajo incertidumbre, donde los humanos toman sus decisiones con conocimiento incierto o basado en sus propias percepciones. En estos casos el Enfoque Lingüístico Difuso [27] modela dicho conocimiento mediante información lingüística obteniéndose resultados satisfactorios en estos problemas de Toma de Decisión que se denominan Toma de Decisión Lingüística (TDL). El éxito del Enfoque Lingüístico Difuso en TDL, se debe, a que los conjuntos difusos son adecuados para tratar la incertidumbre y por la cercanía de la información lingüística al modelo cognitivo humano.

En la literatura existen distintos modelos para realizar

procesos de computación con palabras necesarios en los modelos de resolución de problemas de TDL. Los dos modelos inicialmente utilizados son:

- El modelo basado en el principio de extensión [4, 18, 19, 20] opera sobre los números difusos que definen la semántica de los términos lingüísticos.
- El modelo simbólico que considera una escala ordinal de etiquetas. Este modelo es planteado bajo el uso de los operadores max-min [2, 23], y bajo el uso de la combinación convexa sobre los índices de las etiquetas [5].

Estos modelos computacionales presentan algunas limitaciones de compresión y precisión que son discutidas en [8]. Para superar estas limitaciones se han propuesto nuevos modelos computacionales basados en la escala ordinal de términos lingüísticos y el manejo de los índices de las etiquetas.

En esta contribución nos centraremos en estudiar algunas de estas propuestas que proponen una mejora del modelo simbólico, tales como, el modelo lingüístico 2-Tupla [7], el modelo lingüístico Virtual [22] y el modelo lingüístico 2-Tupla proporcional [21]. El objetivo que perseguimos es analizar y comparar el papel de dichos modelos computacionales en la TDL, y estudiar los procesos de computación con palabras para el tratamiento de percepciones en problemas basados en análisis de decisión.

Para ello este trabajo se estructura como sigue: en la Sección 2, se introducen preliminares necesarios. En la Sección 3, revisamos los diferentes modelos de computación con palabras. En la Sección 4 realizamos un análisis comparativo de los mismos en un problema de TDL. En la Sección 5 analizamos sus implicaciones en la computación con percepciones y finalmente concluimos en la Sección 6.

2 PRELIMINARES

En esta sección revisamos brevemente la Aproximación Lingüística Difusa y sus modelos computacionales

clásicos. También presentamos un problema de TDL y su esquema de resolución.

2.1 APROXIMACIÓN LINGÜÍSTICA DIFUSA

Normalmente trabajamos en entornos cuantitativos donde la información se expresa por medio de valores numéricos. Sin embargo, muchos aspectos del mundo real no pueden ser valorados de esta forma, aunque si mediante percepciones o con conocimiento impreciso. En este caso, un mejor enfoque podría ser la utilización de valoraciones lingüísticas en vez de numéricas. El enfoque lingüístico difuso representa dicho conocimiento mediante variables lingüísticas [27].

Definición 1 Una variable lingüística se caracteriza por una quintupla $(H, T(H), U, G, M)$, donde H es el nombre de la variable; $T(H)$ es el conjunto de términos de H , donde cada valor es una variable difusa notada como X y que varía a lo largo del universo del discurso U , el cuál está asociado con una variable base u ; G es una regla sintáctica para generar los nombres de los valores de H ; y M es una regla semántica para asociar significado $M(X)$, a cada elemento de H , el cuál es un conjunto difuso de U .

Por tanto, hay que elegir adecuadamente los descriptores lingüísticos para el conjunto de términos lingüísticos y su semántica. Para realizar esta tarea, existen diferentes posibilidades. Una de ellas consiste en proporcionar directamente el conjunto de términos, considerando todos los términos distribuidos en una escala la cual tiene un orden definido [24]. Por ejemplo, un conjunto de 7 términos podría ser:

$$S = \{s_0 : nada, s_1 : muy poco, s_2 : poco, s_3 : medio, s_4 : alto, s_5 : muy alto, s_6 : total\}$$

En estos casos se requiere que existan los operadores:

1. Negación $Neg(s_i) = s_j$ tal que $j = g - i$, siendo $(g + 1)$ la cardinalidad de S .)
2. Maximización: $\max(s_i, s_j) = s_i$ si $s_i \geq s_j$
3. Minimización: $\min(s_i, s_j) = s_i$ si $s_i \leq s_j$

La semántica de los términos se da mediante números difusos. Una forma de caracterizar un número difuso es utilizar una representación paramétrica de sus funciones de pertenencia [1]. Por ejemplo, podemos asignar la semántica mostrada en la Fig. 1, al conjunto de siete términos con números difusos triangulares.

2.1.1 Modelos Computacionales Clásicos

El uso de variables lingüísticas implica procesos de computación con palabras, tales como, agregación, comparación, etc. Para realizar estas operaciones la aproximación lingüística difusa proporciona dos modelos computacionales:

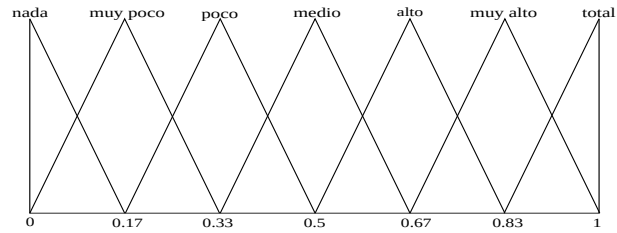


Figura 1: Conjunto de 7 términos lingüísticos

1. **Modelo Basado en el Principio de Extensión:** Opera con términos lingüísticos, $s_i \in S$, mediante operaciones asociadas a sus funciones de pertenencia basadas en el Principio de Extensión. Los resultados obtenidos con los operadores difusos son números difusos, $\bar{s}$, que normalmente no coinciden con ningún término lingüístico del conjunto inicial, $\bar{s} \notin S$. Por lo que para expresarlo lingüísticamente hay que hacer un proceso de aproximación.
2. **Modelo Simbólico:** Este modelo usa la estructura ordenada del conjunto de términos lingüísticos, $S = \{s_0, s_1, \dots, s_g\}$ donde $s_i < s_j$ si $i < j$, para operar. Usando operadores min-max o la combinación convexa donde los resultados son valores numéricos, $\gamma \in [0, g]$, que serán aproximados mediante una función de aproximación $app(\gamma) \in \{0, \dots, g\}$ que obtendrá un valor numérico que indica el índice del término lingüístico asociado, $s_{app(\gamma)} \in S$.

Ambos modelos producen pérdida de información debido a sus procesos de aproximación, y de ahí la falta de precisión en los resultados [8].

2.2 TOMA DE DECISIÓN LINGÜÍSTICA

Un problema básico de TDL consiste en elegir entre un conjunto finito de alternativas, $X = \{x_1, \dots, x_n\}$ sobre el que un conjunto de expertos, $P = \{p_1, \dots, p_m\}$ expresa sus valoraciones, $x_i^j, i = 1 \dots n, j = 1 \dots m$, en un conjunto de términos lingüísticos, $S = \{s_0, \dots, s_g\}$, para seleccionar la mejor alternativa al problema. Los valores, $x_i^j \in S$, son conjuntos difusos definidos por una función de pertenencia en $[0, 1]$ (ver Fig. 1).

Un esquema de resolución de un problema de Toma de Decisión puede verse en la Fig. 2. Dicho esquema consta de dos fases: la *fase de agregación*, para obtener valoraciones colectivas de cada alternativa y una *fase de explotación*, para obtener el conjunto solución de alternativas al problema.

Tong y Bonissone presentaron en [20] una técnica para TDL en la que se asumía que el uso de etiquetas lingüísticas con semántica difusa era adecuada para tratar la incertidumbre propia en estos problemas.

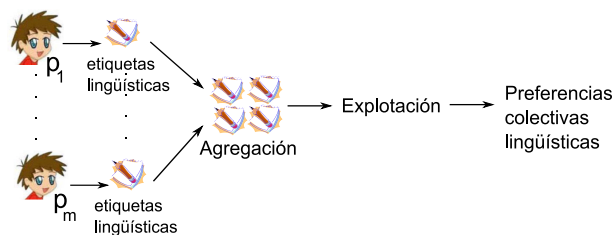


Figura 2: Esquema de Toma de Decisión

También se asumía que los resultados para tomar una decisión, debían ser cuantificables en el lenguaje natural. Diferentes autores [12, 14, 17, 20, 25, 26] asumen que no parece adecuado tomar decisiones basadas en información numérica cuando se ha desarrollado un modelo lingüístico cercano al modelo cognitivo humano. De esta forma, se introduce la idea de una retraslación de la información para ser proporcionada en su salida de forma lingüística [26].

El modelo de resolución propuesto en la Fig. 2 adaptado a un esquema lingüístico sería (ver Fig. 3):

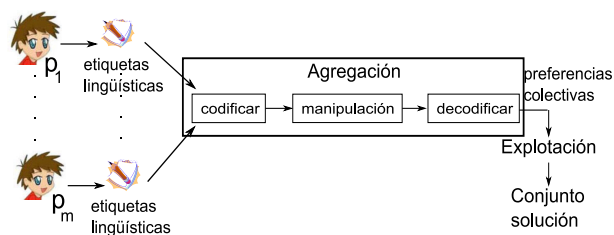


Figura 3: Esquema de Toma de Decisión Lingüístico

3 MODELOS COMPUTACIONALES SIMBÓLICOS BASADOS EN ÍNDICES

La necesidad de tratar con información lingüística en la TDL ha llevado a desarrollar modelos para mejorar los procesos de computación con palabras. Aquí vamos a revisar algunos enfoques simbólicos recientes que pretenden mejorar la precisión de los resultados de la computación con palabras proporcionada por el modelo simbólico revisado en la sección 2.1, tales como, el modelo Lingüístico 2-Tupla [7], el modelo Lingüístico Virtual [22] y el modelo Lingüístico 2-Tupla proporcional [21]. Estos enfoques han sido usados en problemas tales como, Toma de Decisión [22], Evaluación [3], Sistemas de Recomendación [13], etc.

Revisaremos estos modelos analizando la representación de la información lingüística y sus modelos computacionales.

3.1 MODELO LINGÜÍSTICO 2-TUPLA

Este modelo fue presentado en [7] para mejorar la precisión en procesos de computación con palabras.

a) Modelo de Representación:

La información lingüística se representa mediante un par de valores (s_i, α) , siendo s_i un término lingüístico y α un número que representa la traslación simbólica.

Definición 2 La traslación simbólica de un término lingüístico $s_i \in S = \{s_0, \dots, s_g\}$ es un valor numérico definido en $[-0.5, 0.5)$ que representa la “diferencia de información” entre una cantidad de información $\beta \in [0, g]$ obtenida de una operación simbólica y el índice del término lingüístico más cercano.

Este modelo lingüístico define un conjunto de funciones para realizar transformaciones entre valores numéricos y 2-tupla.

Definición 3 Sea $S = \{s_0, \dots, s_g\}$ un conjunto de términos lingüísticos y $\beta \in [0, g]$ un valor obtenido de una operación simbólica, la 2-tupla que expresa la información equivalente a β se obtiene:

$$\Delta : [0, g] \longrightarrow S \times [-0.5, 0.5)$$

$$\Delta(\beta) = (s_i, \alpha), \quad \text{con} \quad \begin{cases} i = \text{round}(\beta), \\ \alpha = \beta - i, \end{cases} \quad (1)$$

Debemos tener en cuenta que Δ es biyectiva [7, 8] y $\Delta^{-1} : S \times [-0.5, 0.5) \longrightarrow [0, g]$ es definida por $\Delta^{-1}(s_i, \alpha) = i + \alpha$.

b) Modelo Computacional:

Junto a este modelo de representación, un modelo computacional basado en las funciones Δ y Δ^{-1} fue definido en [7], con las siguientes operaciones:

1. Comparación de 2-tuplas

Se realiza según un orden lexicográfico. Sea (s_k, α_1) y (s_l, α_2) dos 2-tuplas:

- si $k < l$ entonces $(s_k, \alpha_1) < (s_l, \alpha_2)$
- si $k = l$ entonces
 1. si $\alpha_1 = \alpha_2$ entonces $(s_k, \alpha_1), (s_l, \alpha_2)$ representa la misma información
 2. si $\alpha_1 < \alpha_2$ entonces $(s_k, \alpha_1) < (s_l, \alpha_2)$
 3. si $\alpha_1 > \alpha_2$ entonces $(s_k, \alpha_1) > (s_l, \alpha_2)$

2. Operador de negación de una 2-tupla

Este operador fue definido como:

$$\text{Neg}((s_i, \alpha)) = \Delta(g - (\Delta^{-1}(s_i, \alpha))) \quad (2)$$

donde $g + 1$ es la cardinalidad de S , $S = \{s_0, \dots, s_g\}$.

3. Agregación de 2-tuplas

En [7] fueron definidos distintos operadores de agregación para 2-tuplas lingüísticas.

3.2 MODELO LINGÜÍSTICO VIRTUAL

Este modelo fue propuesto por Xu en [22] para evitar la pérdida de información e incrementar los operadores en procesos de computación con palabras.

a) Modelo de Representación:

En este modelo, Xu extendió el conjunto de términos discreto $S = \{s_0, \dots, s_g\}$ a un conjunto de términos continuo $\bar{S} = \{s_\alpha | s_0 < s_\alpha \leq s_g, \alpha \in [0, g]\}$, donde, si $s_\alpha \in S$, s_α es llamado término lingüístico *original*, en caso contrario, $s_\alpha \in \bar{S}$ es un término lingüístico *virtual*, el cual no tiene asignado ninguna *semántica ni sintaxis*.

Los expertos utilizan términos lingüísticos originales para valorar las variables lingüísticas y los términos lingüísticos virtuales aparecen al operar.

b) Modelo Computacional:

Para llevar a cabo procesos de computación con palabras, Xu introdujo las siguientes operaciones:

Sea $s_\alpha, s_\beta \in \bar{S}$, dos términos lingüísticos y $\mu, \mu_1, \mu_2 \in [0, 1]$.

1. $(s_\alpha)^\mu = s_{\alpha\mu}$
2. $(s_\alpha)^{\mu_1} \otimes (s_\alpha)^{\mu_2} = (s_\alpha)^{\mu_1 + \mu_2}$
3. $(s_\alpha \otimes s_\beta)^\mu = (s_\alpha)^\mu \otimes (s_\beta)^\mu$
4. $s_\alpha \otimes s_\beta = s_\beta \otimes s_\alpha = s_{\alpha\beta}$
5. $s_\alpha \oplus s_\beta = s_{\alpha+\beta}$
6. $s_\alpha \oplus s_\beta = s_\beta \oplus s_\alpha$
7. $\mu s_\alpha = s_{\mu\alpha}$
8. $(\mu_1 + \mu_2)s_\alpha = \mu_1 s_\alpha \oplus \mu_2 s_\alpha$
9. $\mu(s_\alpha \oplus s_\beta) = \mu s_\alpha \oplus \mu s_\beta$

Muchas de estas operaciones no tienen una base cualitativa, por lo que los resultados que obtienen no pueden ser representados ni interpretados en el dominio de expresión inicial de los expertos.

3.3 MODELO LINGÜÍSTICO 2-TUPLA PROPORCIONAL

Este modelo presentado por Wang y Hao en [21] es una generalización y extensión del modelo lingüístico 2-tupla [7].

a) Modelo de Representación:

Este modelo representa la información lingüística mediante una 2-Tupla proporcional, p.ej. $(0.2A, 0.8B)$ sería la clasificación de un curso completo de un estudiante que obtuvo 20%A y 80%B. Los autores remarcan que si B fuese usada como la calificación aproximada, entonces se perdería información. Este enfoque

de 2-Tupla proporcional, está basado en el concepto de *proporción simbólica* [21].

Definición 4 Sea $S = \{s_0, s_1, \dots, s_g\}$ un conjunto de términos ordinales, $I = [0, 1]$ y

$$IS \equiv I \times S = \{(\alpha, s_i) : \alpha \in [0, 1] \text{ y } i = 0, 1, \dots, g\} \quad (3)$$

donde S es el conjunto ordenado de $g + 1$ términos ordinales $\{s_0, \dots, s_g\}$. Dado un par (s_i, s_{i+1}) de dos términos ordinales sucesivos de S , cualquiera dos elementos de $(\alpha, s_i), (\beta, s_{i+1})$ de IS es denominado par de proporción simbólica y α, β son denominados pares de proporciones simbólicas del par (s_i, s_{i+1}) siendo $\alpha + \beta = 1$. Un par de proporción simbólica $(\alpha, s_i), (1 - \alpha, s_{i+1})$ es representado por $(\alpha s_i, (1 - \alpha)s_{i+1})$ y el conjunto de todos los pares de proporciones simbólicas es representado mediante $\bar{S}$, tal que, $\bar{S} = \{(\alpha s_i, (1 - \alpha)s_{i+1}) : \alpha \in [0, 1], i = 0, 1, \dots, g - 1\}$.

$\bar{S}$ es denominado el *conjunto ordinal 2-Tupla proporcional* generado por S y los miembros de $\bar{S}$, *2-Tupla proporcional ordinal*, el cual es usado para representar la información ordinal en computación con palabras.

De forma similar a [7], Wang y Hao definieron funciones para facilitar las operaciones con este tipo de representación.

Definición 5 Sea $S = \{s_0, s_1, \dots, s_g\}$ un conjunto de términos ordinales y $\bar{S}$ el conjunto 2-Tupla proporcional ordinal generado por S . La función $\pi : \bar{S} \rightarrow [0, g]$ fue definida como sigue:

$$\pi((\alpha s_i, (1 - \alpha)s_{i+1})) = i + (1 - \alpha), \quad (4)$$

donde $i = \{0, 1, \dots, g - 1\}, \alpha \in [0, 1]$ y π es la posición de la función índice de la 2-tupla ordinal.

Cabe destacar que la función π es biyectiva, $\bar{S} \rightarrow [0, g]$ y su inversa $\pi^{-1} : [0, g] \rightarrow \bar{S}$ es definida como sigue:

$$\pi^{-1}(x) = ((1 - \beta)s_i, \beta s_{i+1}) \quad (5)$$

donde $i = E(x)$, E es la parte entera de la función, $\beta = x - i$.

b) Modelo Computacional:

Para operar con información lingüística, Wang y Hao definieron los siguientes operadores:

1. Comparación de 2-Tupla proporcional

Para cualquier $(\alpha s_i, (1 - \alpha)s_{i+1}), (\beta s_j, (1 - \beta)s_{j+1}) \in \bar{S}$, se define $(\alpha s_i, (1 - \alpha)s_{i+1}) < (\beta s_j, (1 - \beta)s_{j+1}) \Leftrightarrow \alpha i + (1 - \alpha)(i + 1) < \beta j + (1 - \beta)(j + 1) \Leftrightarrow i + (1 - \alpha) < j + (1 - \beta)$.

De esta forma, para cualquier dos 2-tupla proporcionales $(\alpha s_i, (1 - \alpha)s_{i+1})$ y $(\beta s_j, (1 - \beta)s_{j+1})$:

- si $i < j$, entonces

Tabla 1: Resolución del problema de TDL con distintos modelos simbólicos

	$\bar{x}_1^e$	$\bar{x}_2^e$	$\bar{x}_3^e$	$\bar{x}_4^e$
2-Tupla	(medio, .25)	(medio, -.5)	(poco, .25)	(medio, -.25)
Ling. Virtual	(s3.25)	(s2.5)	(s2.25)	(s2.75)
2-T Prop.	(0.75medio, 0.25alto)	(0.5poco, 0.5medio)	(0.75poco, 0.25medio)	(0.25poco, 0.75medio)

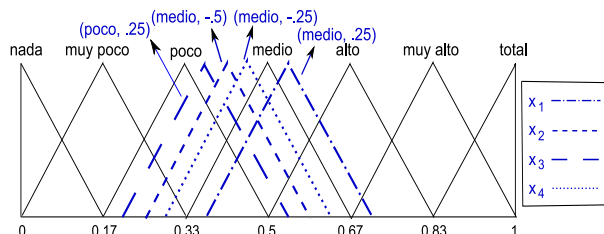


Figura 4: Resultados del modelo 2-Tupla para el problema de TDL

1. $(\alpha s_i, (1-\alpha)s_{i+1}), (\beta s_j, (1-\beta)s_{j+1})$ son iguales cuando $i = j - 1$ y $\alpha = 0, \beta = 1$
2. $(\alpha s_i, (1-\alpha)s_{i+1}) < (\beta s_j, (1-\beta)s_{j+1})$ en otro caso,
 - si $i = j$, entonces
 1. si $\alpha = \beta$ entonces $(\alpha s_i, (1-\alpha)s_{i+1}), (\beta s_j, (1-\beta)s_{j+1})$ representan la misma información
 2. si $\alpha < \beta$ entonces $(\alpha s_i, (1-\alpha)s_{i+1}) < (\beta s_j, (1-\beta)s_{j+1})$
 3. si $\alpha > \beta$ entonces $(\alpha s_i, (1-\alpha)s_{i+1}) > (\beta s_j, (1-\beta)s_{j+1})$
2. Operador de negación de una 2-Tupla proporcional

$Neg((\alpha s_i, (1-\alpha)s_{i+1})) = ((1-\alpha)s_{g-i-1}, \alpha s_{g-i}),$ (6) siendo $g+1$ la cardinalidad de S .

3. Agregación de 2-Tupla proporcional

Wang y Hao definieron varios operadores de agregación para manejar procesos de computación con palabras. Las definiciones de estos operadores de agregación están basados en valores característicos de las etiquetas lingüísticas [21].

4 ANÁLISIS COMPARATIVO

En esta sección resolvemos un problema de TDL con los diferentes modelos presentados en la sección 3 y hacemos un estudio comparativo de los mismos.

4.1 PROBLEMA DE TOMA DE DECISIÓN

Supongamos que una pequeña empresa quiere renovar los ordenadores a sus empleados de ventas, $P = \{p_1, p_2, p_3, p_4\}$, que no son expertos en informática. Para ello les solicita su opinión sobre cuál de las distintas alternativas que tiene se adapta mejor a sus necesidades. Las alternativas son las siguientes:

x_1	x_2	x_3	x_4
PC	Portatil	Netbook	Imac

Dado que los empleados no son expertos en informática, sus preferencias están fuertemente marcadas por sus percepciones e incluyen cierta incertidumbre debido a su falta de conocimiento. Por lo que, expresarán dichas preferencias lingüísticamente en el conjunto $S = \{s_0 : nada, s_1 : muy poco, s_2 : poco, s_3 : medio, s_4 : alto, s_5 : muy alto, s_6 : total\}$, cuya semántica fue mostrada en la Fig. 1. Cada empleado proporciona un vector de preferencias:

	μ_{ij}	alternativas			
		x_1	x_2	x_3	x_4
expertos	p_1	poco	medio	medio	poco
	p_2	medio	poco	muy poco	alto
	p_3	alto	muy poco	medio	medio
	p_4	alto	alto	poco	poco

Utilizando el modelo de resolución de la Fig. 3 y los distintos modelos simbólicos revisados en la sección 3, obtenemos los resultados de la Tabla 1. En la Fig. 4 se muestra la representación semántica de los resultados del modelo 2-Tupla, mientras que para el modelo virtual y 2-Tupla proporcional no es posible realizar figuras análogas porque no tienen una representación semántica.

La mejor alternativa en sendos modelos es $x_1 = PC$. A continuación analizamos los distintos resultados obtenidos.

4.2 ESTUDIO COMPARATIVO

Este análisis estudia si las propuestas simbólicas revisadas y aplicadas a un problema de TDL se adecúan a la técnica expuesta en la sección 2.2. Para ello se analiza la representación de la información, su operativa

Tabla 2: Análisis comparativo de modelos simbólicos

	2-Tupla	Lingüístico Virtual	2-Tupla proporcional
Representación	Difusa	No difusa	No difusa
Modelo Computacional	Lingüístico	No lingüístico	Lingüístico
Precisión	Etiquetas equidistantes	Siempre, no semántica	Misma amplitud
Comprensión	Fácil de entender	Útil en ordenaciones	Comprensible

computacional, precisión y comprensión de los resultados obtenidos (ver Tabla 2). A partir de dicha tabla que se basa en los resultados obtenidos en el problema de TDL de la sección 4.1 podemos indicar:

- En el modelo lingüístico Virtual [22] las decisiones no se toman basándose en valoraciones lingüísticas difusas como se asumía en [12, 20, 25] y reflejaba el esquema de la Fig. 3, ya que sus valores lingüísticos no utilizan ni semántica, ni sintáxis, tal y como define el Enfoque Lingüístico Difuso; siendo su interpretación numérica. La 2-Tupla proporcional [21] tampoco mantiene una representación difusa, ya que utiliza las proporciones de dos etiquetas lingüísticas consecutivas para representar el resultado. Sin embargo, el modelo 2-Tupla [7] sí que mantiene una representación difusa de la información lingüística tal y como se muestra en la Fig. 4.
- El modelo computacional del modelo Virtual es preciso en cualquier conjunto de etiquetas, ya que no utiliza semántica alguna. Además de poder utilizar valores fuera del universo del discurso de la variable lingüística. Los otros modelos sólo pueden obtener valores en el universo del discurso de la variable y garantizan precisión cuando el conjunto de etiquetas es simétrico y uniformemente distribuido en el caso de la 2-Tupla y cuando el soporte de todas las etiquetas es el mismo en la 2-Tupla proporcional [7, 21].
- Si analizamos los resultados obtenidos por los 3 modelos, observamos que el modelo Virtual obtiene valores de difícil interpretación más allá de una mera ordenación, por tanto, no son adecuados para representar incertidumbre ni cercanos al modelo cognitivo de los seres humanos, ya que no son propiamente lingüísticos. Mientras que los modelos 2-Tupla y 2-Tupla proporcional ofrecen resultados cualitativos fáciles de entender, aunque el último tiene un poco más de complejidad, ya que utiliza cuatro valores para expresar una única valoración.

Del análisis anterior vemos que el modelo 2-Tupla es el único modelo basado en el Enfoque Lingüístico Difuso, ya que mantiene una sintáxis y semántica difusa al representar y operar con términos lingüísticos. Lo cual

lo hace adecuado para el tratamiento de la incertidumbre y es cercano al modelo cognitivo de los humanos. Además, como veremos en la siguiente sección lo hace útil y apropiado para el tratamiento de percepciones subjetivas en aquellos problemas donde aparecen.

5 COMPUTACIÓN CON PERCEPCIONES: USO DE LA 2-TUPLA LINGÜÍSTICA

En [14], Mendel analizó el concepto de computación perceptual o computación con percepciones, como una instancia del paradigma de computación con palabras, para la realización de decisiones o creación de opiniones subjetivas. En [15] se presentó un repaso histórico a los orígenes de la computación con percepciones. En este análisis, a partir de la idea inicial de Tong y Bonissone [20], se establece como regla básica de la computación con percepciones que: “el modelado de la información subjetiva debe ser lingüístico y los resultados de operar con la misma para construir opiniones o tomar decisiones tiene que ser igualmente lingüística” (ver Fig. 5).

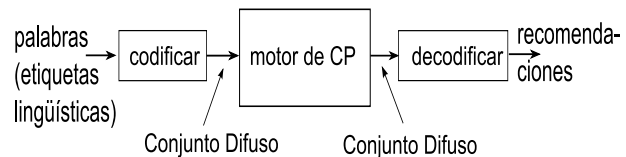


Figura 5: Computación perceptual [14]

En términos similares otros autores como Schumcker [17] o Yager [25, 26] han propuesto esquemas lingüísticos para el tratamiento de información subjetiva. La Fig. 6 muestra la propuesta de Yager. Zadeh



Figura 6: Computación con palabras [25]

en [28] indica como la manipulación de percepciones es una pieza clave en procesos de razonamiento y decisión. También expone la diferencia básica entre medidas y percepciones, donde las primeras son generalmente precisas y las últimas difusas.

Teniendo en cuenta las ideas expuestas anteriormente, el estudio realizado en la sección 4 nos muestra como el modelo lingüístico de 2-Tupla es adecuado para la computación perceptual. Además distintas extensiones del modelo básico de 2-Tupla permitirán abordar ciertos problemas en la base de la computación con percepciones que con otros modelos no sería posible. A continuación justificaremos la adecuación de la 2-Tupla a la computación perceptual y mostraremos distintas extensiones útiles de dicho modelo para la computación con percepciones.

5.1 2-TUPLA LINGÜÍSTICA Y COMPUTACIÓN CON PERCEPCIONES

Tal y como acabamos de revisar, la teoría de computación con percepciones establece la regla que al operar con las mismas, tanto las entradas como las salidas de las operaciones deben estar representadas lingüísticamente. La teoría de conjuntos difusos [11] proporciona una representación semántica de la información lingüística que permite cumplir esta regla. En esta línea el modelo 2-Tupla cumple dicha regla, ya que las entradas y salidas son información lingüística basada en la teoría de conjuntos difusos (Ver Fig. 4).

El uso de percepciones es clave en problemas como la evaluación sensorial, donde modelos basados en la 2-Tupla han proporcionado resultados satisfactorios en distintos problemas [12, 16].

5.2 EXTENSIONES DEL MODELO 2-TUPLA EN COMPUTACIÓN CON PERCEPCIONES

Además de la utilización del modelo básico de 2-Tupla en computación con percepciones, se han presentado distintas extensiones formales sobre dicho modelo básico para operar con ellas, tales como:

- Tratamiento de información heterogénea [10]: se presentó un proceso para operar con información lingüística, numérica e intervalar basada en el modelo de 2-Tupla. Este tipo de información puede aparecer en problemas con distintos criterios, algunos subjetivos y otros objetivamente medibles.
- Tratamiento de información lingüística valorada en distintos conjuntos de términos [9]: en los problemas basados en percepciones, distintos expertos pueden tener distinto nivel de adquisición de las percepciones, ej: desarrollo de sentidos sensoriales. En esos casos, la posibilidad de tratar información con distinta granularidad produce mejores resultados en la resolución de dichos problemas.
- Tratamiento de escalas no balanceadas [6]: en muchos problemas basados en percepciones el uso

de escalas bipolares es común, pero no tienen porqué ser simétricamente distribuidas. Para estos problemas se ha desarrollado una metodología basada en 2-Tupla que permite tratarlos manteniendo una representación y precisión remarcable.

6 CONCLUSIONES

La computación con palabras y con percepciones son claves para la solución de problemas relacionados con modelos cognitivos. El uso de información lingüística para modelar información vaga, incierta o imprecisa propia en los procesos cognitivos, es adecuada tanto desde el punto de vista de tratamiento de la incertidumbre, como de la comprensión de los resultados.

En esta contribución hemos revisado los modelos computacionales simbólicos de 2-Tupla [7], Virtual [21] y 2-Tupla proporcional [22], para aplicarlos a un problema de TDL. Analizando los resultados obtenidos con dichos modelos, hemos visto como el modelo de 2-Tupla es el único que mantiene la base del Enfoque Lingüístico Difuso, ya que mantiene la representación lingüística (semántica y sintaxis) de sus resultados. Mientras que los otros modelos no lo hacen.

Aprovechando el estudio anterior, hemos incluido un apunte de cómo el modelo de 2-Tupla es un modelo adecuado para la computación perceptual [14] y que cumple su regla básica de representar la información de los expertos y los resultados de forma lingüística. Finalmente, hemos apuntado distintas extensiones realizadas sobre el modelo de 2-Tupla que se han mostrado útiles en procesos de computación perceptual.

Agradecimientos

Este trabajo está parcialmente subvencionado por los proyectos de investigación TIN-2009-08286, P08-TIC-3548 y fondos FEDER.

Referencias

- [1] P.P. Bonissone and K.S. Decker. *Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-Off Precision and Complexity*. In L.H. Kanal and J.F. Lemmer, Editors., *Uncertainty in Artificial Intelligence*. North-Holland, 1986.
- [2] J.J. Buckley. The multiple judge, multiple criteria ranking problem: A fuzzy set approach. *Fuzzy Sets and Systems*, 13:23–37, 1984.
- [3] G. Buyukozkan, D. Ruan, and O. Feyzioglu. Evaluating e-learning web site quality in a fuzzy environment. *International Journal of Intelligent System*, 22:567–586, 2007.
- [4] R. Degani and G. Bortolan. The problem of linguistic approximation in clinical decision making.

- International Journal of Approximate Reasoning*, 2:143–162, 1988.
- [5] M. Delgado, J.L. Verdegay, and M.A. Vila. On aggregation operations of linguistic labels. *International Journal of Intelligent Systems*, 8(3):351–370, 1993.
 - [6] F. Herrera, E. Herrera-Viedma, and L. Martínez. A fuzzy linguistic methodology to deal with unbalanced linguistic term sets. *IEEE Transactions on Fuzzy Systems*, 16(2):354–370, 2008.
 - [7] F. Herrera and L. Martínez. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 8(6):746–752, 2000.
 - [8] F. Herrera and L. Martínez. The 2-tuple linguistic computational model. Advantages of its linguistic description, accuracy and consistency. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(Suppl.):33–49, 2001.
 - [9] F. Herrera and L. Martínez. A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making. *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*, 31(2):227–234, 2001.
 - [10] F. Herrera, L. Martínez, and P.J. Sánchez. Managing non-homogeneous information in group decision making. *European Journal of Operational Research*, 166(1):115–132, 2005.
 - [11] G.J. Klir and B. Yuan. *Fuzzy sets and fuzzy logic: Theory and Applications*. Prentice-Hall PTR, 1995.
 - [12] L. Martínez. Sensory evaluation based on linguistic decision analysis. *International Journal of Approximate Reasoning*, 44 Num 2:148–164, 2007.
 - [13] L. Martínez, M.J. Barranco, L.G. Perez, , and M. Espinilla. A knowledge based recommender system with multigranular linguistic information. *International Journal of Computational Intelligence Systems*, 1(3):225 – 236, 2008.
 - [14] J.M. Mendel. The perceptual computer: An architecture for computing with words. In *FUZZ-IEEE*, pages 35–38, 2001.
 - [15] J.M. Mendel. Historical reflections and new positions on perceptual computing. *Fuzzy Optimization and Decision Making*, 2009. In press.
 - [16] D. Ruan and X. Zeng (Eds.). *Sensory Evaluation: Methodologies and Applications*. Springer, 2004.
 - [17] K.J. Schmucker. *Fuzzy sets, natural language computations, and risk analysis*. Computer Science Press, Rockville, 1984.
 - [18] M.S. Shendrik and B.G. Tamm. Approach to interactive solution of multicritical optimization problems with linguistic modeling of preferences. *Automatic Control and Computer Sciences*, 19(6):3–9, 1985.
 - [19] V.B. Silov and D.V. Vilenchik. Linguistic decision-making methods in multicriterial selection of models. *Soviet Journal of Automation and Information Sciences*, 18(4):92–94, 1985.
 - [20] M. Tong and P.P. Bonissone. A linguistic approach to decision making with fuzzy sets. *IEEE Transactions on Systems, Man and Cybernetics*, 10:716–723, 1980.
 - [21] J. Wang and J. Hao. A new version of 2-tuple fuzzy linguistic representation model for computing with words. *IEEE transactions on fuzzy systems*, 14:435–445, 2006.
 - [22] Z.S. Xu. A method based on linguistic aggregation operators for group decision making with linguistic preference relations. *Information Science*, 166:19–30, 2004.
 - [23] R.R. Yager. A new methodology for ordinal multiobjective decision based on fuzzy sets. *Decision Science*, 12:589–600, 1981.
 - [24] R.R. Yager. An approach to ordinal decision making. *International Journal of Approximate Reasoning*, 12:237–261, 1995.
 - [25] R.R. Yager. *Computing with words and information/intelligent systems 2:applications*, chapter Approximate reasoning as a basis for computing with words, pages 50–77. Physica Verlag, 1999.
 - [26] R.R. Yager. On the retranslation process in zadeh’s paradigm of computing with words. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 34:1184–1195, 2004.
 - [27] L.A. Zadeh. The concept of a linguistic variable and its applications to approximate reasoning. *Information Sciences, Part I, II, III*, 8,8,9:199–249,301–357,43–80, 1975.
 - [28] L.A. Zadeh. *Computing with Words in Information/Intelligent Systems 1*, chapter What is Computing with Words? Physica-Verlag, 1999.

USO DE JERARQUÍAS LINGÜÍSTICAS EXTENDIDAS PARA MEJORAR LA REPRESENTACIÓN DE INFORMACIÓN LINGÜÍSTICA NO BALANCEADA

M. Espinilla¹ L. Martínez¹

¹ Dpto. de Informática, Universidad de Jaén, 23071 - Jaén, España, {mestevez,martin}@ujaen.es

Resumen

La utilización de variables lingüísticas para modelar la información en problemas implica la realización de procesos de computación con palabras. Existen modelos computacionales para operar con precisión en un conjunto de términos lingüísticos simétricos y uniformemente distribuidos. Sin embargo, podemos encontrar problemas que necesitan de conjuntos de términos lingüísticos no balanceados para representar la información que manejan. Recientemente, se ha presentado una metodología para representar conjuntos de términos lingüísticos no balanceados y operar sobre ellos sin pérdida de información. Sin embargo, esta metodología tiene distintas limitaciones de representación. En esta contribución, revisamos éste método para representar conjuntos de términos lingüísticos no balanceados y presentamos una nueva metodología que supera sus limitaciones de representación.

1. INTRODUCCIÓN

La necesidad de modelar información vaga e imprecisa en problemas reales de Evaluación Sensorial, Toma de Decisiones, Consenso, Sistemas de Recuperación de Información, etc., ha producido que surjan diferentes modelos y técnicas para su tratamiento. En esta contribución nos centramos en el uso del Enfoque Lingüístico Difuso [9] para modelar dicha información mediante el uso de variables lingüísticas que implican la realización de procesos de computacionales con palabras.

En muchos problemas que trabajan con información lingüística se ha modelado dicha información con conjuntos de etiquetas simétricas y uniformemente distribuidas, existiendo modelos computacionales precisos

para este tipo de información [5]. Sin embargo, en distintos problemas [1, 7] aparece la necesidad de trabajar con información lingüística no balanceada. En [3, 4] se presentó una metodología para representar conjuntos de términos lingüísticos no balanceados (CTLNB) y operar de forma precisa sobre los mismos. Para ello, esta metodología hace uso de las Jerarquías Lingüísticas [6] que son una estructura lingüística que permite operar con variables lingüísticas valoradas en conjuntos lingüísticos con distinta granularidad, imponiendo un conjunto de reglas. En [2, 8] se presentó una mejora sobre las Jerarquías Lingüísticas denominada Jerarquías Lingüísticas Extendidas que relaja las reglas de construcción permitiendo una mayor flexibilidad de representación.

En esta contribución, utilizamos las Jerarquías Lingüísticas Extendidas para representar CTLNB y así superar las restricciones de representación debidas al uso de las Jerarquías Lingüísticas. Igualmente, esta nueva propuesta podrá operar con el CTLNB de forma precisa, aunque el modelo computacional no es objeto de estudio en este trabajo. Debido a la limitación de espacio, hacemos una propuesta inicial que muestra las ideas básicas de la nueva metodología y que será extendida en el futuro.

Este trabajo se estructura como sigue: en la Sección 2 revisamos los conceptos básicos necesarios para nuestra propuesta. En la Sección 3, vemos brevemente la metodología actual para representar CTLNB. En la Sección 4, presentamos una nueva metodología para representar CTLNB basada en el uso de las jerarquías lingüísticas extendidas. Finalmente, indicaremos una serie de conclusiones.

2. INFORMACIÓN LINGÜÍSTICA

Hemos indicado el uso de las Jerarquías Lingüísticas y de las Jerarquías Lingüísticas Extendidas como base de las metodologías de representación de CTLNB, por lo que en esta sección, hacemos una breve revisión de

las mismas.

2.1. JERARQUÍAS LINGÜÍSTICAS

Las Jerarquías Lingüísticas se presentaron como una estructura para realizar procesos computacionales con información lingüística multi-granular [6].

Una Jerarquía Lingüística se compone de un conjunto de niveles, donde cada nivel es un conjunto de términos lingüísticos simétrico y uniformemente distribuido con distinta granularidad al resto de niveles de su jerarquía. Cada nivel de la jerarquía es notado como, $l(t, n(t))$, siendo t un número que indica el nivel de la jerarquía y $n(t)$ la granularidad del conjunto lingüístico del nivel t .

Los niveles dentro de una jerarquía están ordenados de acuerdo a su granularidad, es decir, para dos niveles sucesivos t y $t + 1$, $n(t + 1) > n(t)$. Una Jerarquía Lingüística, LH, se define como la unión de todos los niveles t : $LH = \bigcup_t l(t, n(t))$ donde el conjunto de etiquetas del nivel t es representado como $S^{n(t)}$.

La construcción de una Jerarquía Lingüística debe satisfacer las siguientes reglas básicas [6]:

- *Regla 1.* Preservar todos los puntos modales previos de las funciones de pertenencia de cada uno de los términos lingüísticos de cada nivel con respecto a los del nivel siguiente.
- *Regla 2.* Realizar las transiciones entre dos niveles consecutivos suaves. El propósito es construir un nuevo conjunto de términos lingüísticos, $S^{n(t+1)}$, de modo que se añada un nuevo término lingüístico entre cada pareja de términos pertenecientes al conjunto de términos del nivel anterior t . Para realizar esta inserción de nuevos términos, se reduce el soporte de las etiquetas lingüísticas para dejar hueco entre ellas para la nueva etiqueta.

Estas reglas imponen que la granularidad de un conjunto de términos de un nivel $t + 1$ se obtiene de su predecesor t con la siguiente expresión: $l(t, n(t)) \rightarrow l(t + 1, 2 \cdot n(t) - 1)$, lo que implica una limitación en los conjuntos de términos lingüísticos que pueden pertenecer a una LH como se puede observar gráficamente en las Figuras 1 y 2.

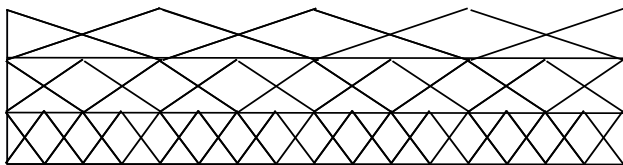


Figura 1: Jerarquía Lingüística de 5, 9 y 17 etiquetas.

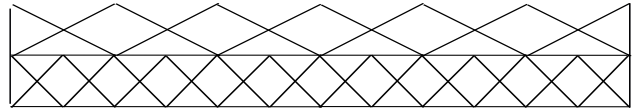


Figura 2: Jerarquía Lingüística de 7 y 13 etiquetas.

Junto a las Jerarquías Lingüísticas se ha desarrollado un modelo computacional que permite operar sin pérdida de información [6].

2.2. JERARQUÍAS LINGÜÍSTICAS EXTENDIDAS

Las Jerarquías Lingüísticas Extendidas se presentaron en [2, 8] para eliminar la limitación de los conjuntos de etiquetas que pueden pertenecer a una Jerarquía Lingüística y seguir operando de forma precisa sobre ellos. Para ello, las Jerarquías Lingüísticas Extendidas relajaron las reglas de construcción de una Jerarquía Lingüística, tal como sigue:

- *Regla 1 Extendida:* Incluir en la Jerarquía Lingüística Extendida, ELH, el conjunto de niveles necesarios ordenados por su granularidad, $l(t, n(t))$, $t = 1, \dots, m$.
- *Regla 2 Extendida:* Añadir un último nivel, $l(t', n(t'))$ con $t' = m + 1$ con la granularidad $n(t') = m.c.m.(n(t) - 1) + 1$, $t = 1, \dots, m$, siendo *m.c.m.* el *mínimo común múltiplo*. Este nivel se utilizará únicamente para realizar los procesos de computación.

Por tanto, la construcción de una Jerarquía Lingüística Extendida implica la inclusión de los m conjuntos de términos lingüísticos a utilizar y la adición final de un nuevo nivel $l(t', n(t'))$, con $t' = m + 1$ y con la granularidad mencionada anteriormente. En este último nivel se llevan a cabo los procesos computacionales de forma precisa [2, 8]. Una Jerarquía Lingüística Extendida puede incluir niveles con 5 y 7 etiquetas lingüísticas (ver Figura 3), niveles que no podían pertenecer a una misma Jerarquía Lingüística (ver Figuras 1 y 2).

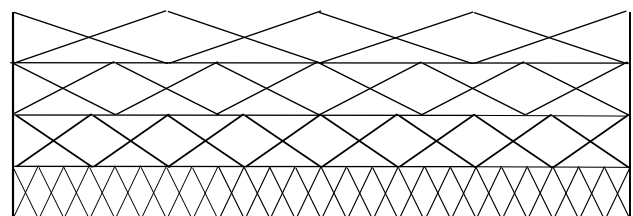


Figura 3: Jerarquía Lingüística Extendida de 5, 7, 9 y 25 etiquetas.

3. REPRESENTACIÓN DE CTLNB. ANTECEDENTES

Para entender nuestra propuesta de representación de CTLNB, en primer lugar haremos una revisión del método de representación presentado en [3, 4].

En el método un CTLNB, S^{nb} , se representa utilizando una Jerarquía Lingüística dada. S^{nb} está caracterizado por una etiqueta central y para representarlo se divide el conjunto en 3 subconjuntos $S^{nb} = S_L \cup S_C \cup S_R$.

- S_L contiene las etiquetas situadas a la izquierda de la etiqueta central. El número de etiquetas del subconjunto es representado como $\#(S_L)$.
- S_C contiene la etiqueta central y,
- S_R contiene las etiquetas situadas a la derecha del término central. El número de etiquetas del subconjunto es representado como $\#(S_R)$.

Básicamente, el método busca en una Jerarquía Lingüística, LH, un nivel con una granularidad igual a la de cada subconjunto, S_L y S_R . Si existe un nivel con dicha granularidad representa el subconjunto de forma uniforme en ese nivel. En el caso de que no exista un nivel adecuado, el subconjunto es representado de forma no uniforme mediante el uso de dos niveles consecutivos, $\{l(t, n(t)), l(t+1, n(t+1))\} \in LH$. De modo que la zona con mayor concentración de etiquetas se representa en el nivel de mayor granularidad, $n(t+1)$, y la zona con una menor concentración de etiquetas en el nivel con menor granularidad, $n(t)$.

A continuación, mostramos brevemente el procedimiento:

- Si S_L o S_R satisface (1) entonces el subconjunto es representado con una distribución uniforme, usando el nivel de la Jerarquía Lingüística que cumple (1).

$$\exists t \in LH, \frac{n(t) - 1}{2} = \#(S_{[L,R]}) \quad (1)$$

- Si S_L o S_R no satisface (1) entonces se buscan dos niveles consecutivos $\{t, t+1\} \in LH$, tal que:

$$\frac{n(t) - 1}{2} < \#(S_{[L,R]}) < \frac{n(t+1) - 1}{2} \quad (2)$$

En este caso, se define la distribución del subconjunto: extrema o central, siendo *extrema* cuando se agrupa un mayor número de etiquetas en el extremo del CTLNB y *centrada* cuando se concentra un mayor número de etiquetas cerca de S_C .

Este método presenta limitaciones debido a que las Jerarquías Lingüísticas están prefijadas, es decir, los conjuntos de términos lingüísticos que pertenecen a LH no pueden modificarse libremente sino que deben de respetar las reglas de construcción. A continuación, mostramos algunas de las limitaciones más evidentes:

1. Cuando existe un subconjunto del CTLNB distribuido de forma uniforme, pero la LH que se proporciona no tiene un nivel con una granularidad adecuada, el método solicita la distribución, centrada o extrema, y representa el subconjunto de forma no uniforme con la distribución señalada.
2. Cuando se quiere representar un subconjunto del CTLNB de forma no uniforme y LH contiene un nivel con una granularidad adecuada para representar ese subconjunto, el método no solicita ninguna distribución y representa el subconjunto de forma uniforme.

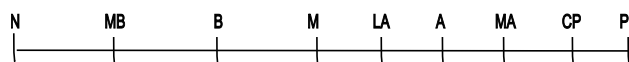


Figura 4: CTLNB de 9 etiquetas.

Por ejemplo, dada la LH de la Figura 1 y el CTLNB mostrado en la Figura 4, el método revisado representaría el CTLNB de la forma que se observa en la Figura 5.

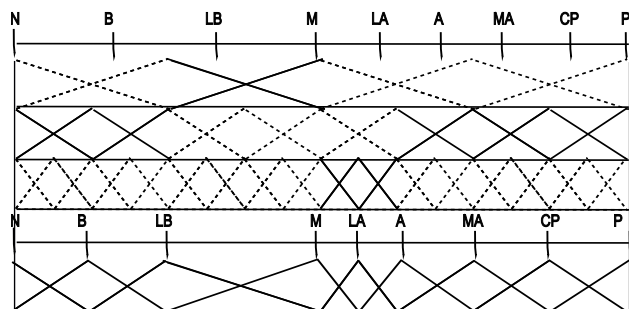


Figura 5: Representación de 9 etiquetas.

Vemos que aunque S_R está distribuido uniformemente, el método lo representa de forma no uniforme, en este caso con una distribución centrada, ya que no hay ningún nivel en LH que sea adecuado para representar el subconjunto. Del mismo modo ha representado S_L que aún siendo uniforme, el método lo ha representado de forma no uniforme.

4. REPRESENTACIÓN DE UN CTLNB CON JERARQUÍAS LINGÜÍSTICAS EXTENDIDAS

En esta sección, presentamos una metodología que representa CTLNB mediante el uso de las Jerarquías Lingüísticas Extendidas, con el objetivo de superar las limitaciones provocadas por el uso de las Jerarquías Lingüísticas de la metodología propuesta en [4].

La idea principal de la metodología que presentamos se fundamenta en la flexibilidad en la construcción de las Jerarquías Lingüísticas Extendidas, ya que permiten incluir cualquier conjunto de términos frente a los conjuntos predefinidos de las Jerarquías Lingüísticas. Para representar un CTLNB, nuestra propuesta aprovecha la flexibilidad construyendo una Jerarquía Lingüística Extendida *ad hoc*, atendiendo a las necesidades de representación del CTLNB y no al revés, como sucede en [4], que es el CTLNB el que debe adaptarse a la Jerarquía Lingüística.

La nueva metodología de representación consiste en dos fases, la primera de construcción de la ELH *ad hoc* y la segunda de representación del CTLNB (ver Figura 6).

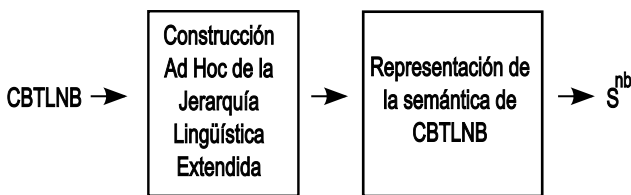


Figura 6: Fases de la nueva metodología.

4.1. CONSTRUCCIÓN DE LA JERARQUÍA LINGÜÍSTICA EXTENDIDA AD HOC

En esta fase se incluyen los niveles necesarios para representar el CTLNB, S^{nb} , en la Jerarquía Lingüística Extendida, ELH, y que son determinados por la distribución de S_L y S_R . Primero, se incluyen los conjuntos para S_L y, a continuación, para S_R en ELH.

Atendiendo a la distribución de etiquetas se pueden dar los siguientes casos:

- *Caso 1. La distribución en S_L es uniforme.* Añadir a la jerarquía un nivel, $l(k, n(k))$ cuya granularidad se determina como sigue:

$$n(k) = 2 \cdot \#(S_L) + 1$$

- *Caso 2. La distribución de S_L es extrema y $\#(S_L)$ es impar.* Añadir dos niveles a la jerarquía,

$l(i, n(i))$ y $l(j, n(j))$, cuyas granularidades son determinadas por las siguientes expresiones:

$$n(j) = (\#S_L - 1) \cdot 2 + 1, \quad n(i) = (2 \cdot n(j)) - 1$$

- *Caso 3. La distribución de S_L es centrada y $\#(S_L)$ es impar.* Se añaden dos niveles, cuyas granularidades son similares al caso 2.

Evidentemente, existe un mayor número de casos de distribución de etiquetas en el CTLNB, pero inicialmente nos vamos a centrar en estos tres para mostrar el funcionamiento general de nuestra propuesta.

Una vez incluidos los niveles para S_L , continuamos con los niveles necesarios para S_R . El procedimiento es similar, salvo que algún nivel necesario para S_R ya hubiese sido incluido para S_L entonces no hace falta añadirlo.

Para finalizar la construcción de la Jerarquía Lingüística Extendida, se incluye un último nivel siguiendo la Regla Extendida 2. En este nivel se llevarán a cabo los procesos de computación con palabras.

4.2. REPRESENTACIÓN DE LA SEMÁNTICA DE CBTLNB

La representación de la semántica de un CTLNB depende de la distribución y del número de las etiquetas de los subconjuntos S_L y S_R . Para representar la semántica de CTLNB es indistinto qué subconjunto, S_L o S_R , se represente primero, ya que el procedimiento para cada subconjunto es independiente. Aquí, describimos el procedimiento para la representación del subconjunto S_L , siendo el procedimiento para representar S_R análogo.

Dado que la representación de la semántica del CTLNB en la Jerarquía Lingüística Extendida depende de la distribución y del número de etiquetas del subconjunto. El primer paso, consiste en conocer las etiquetas que se asignará a cada nivel.

Si la distribución es uniforme se utiliza un solo nivel $l(k, n(k))$ para representar el subconjunto, asignando las etiquetas del subconjunto $\#S_L$ la semántica de las etiquetas del nivel $l(k, n(k))$.

Si la distribución es centrada o extrema, se utilizan para la representación 2 niveles, $l(j, n(j))$ y $l(i, n(i))$, que han sido incluidos en ELH en la fase anterior. El nivel de mayor granularidad, $l(i, n(i))$, representa la zona con mayor concentración de etiquetas y el nivel de menor granularidad, $l(j, n(j))$, representa la zona con menor concentración de etiquetas. Para asignar la semántica, calculamos el número de etiquetas que se representarán en $l(i, n(i))$ y en $l(j, n(j))$:

$$lab_j = \frac{n(i) - 1}{2} - \#S_L, \quad lab_i = \#S_L - lab_j$$

Siendo lab_j el número de etiquetas de S_L representadas en el nivel $l(j, n(j))$ y lab_i en número de etiquetas de S_L representadas en el nivel $l(i, n(i))$. Cabe mencionar que habrá una etiqueta, llamada *punte*, que su semántica se representará utilizando media etiqueta del nivel $l(i, n(i))$ y media etiqueta del nivel $l(j, n(j))$. Para ello, primero asignamos a la etiqueta punte una etiqueta completa en el correspondiente nivel y a continuación, reasignamos media etiqueta con la correspondiente media etiqueta en el otro nivel.

A continuación, atendiendo a la distribución S_L y su número de etiquetas, describimos el procedimiento de representación en cada uno de los casos existentes.

Representación de S_L

- *Caso 1. La distribución de S_L es uniforme.*
 1. A las etiquetas del lado izquierdo de S^{nb} se les asignan todas las etiquetas del lado izquierdo del nivel $l(k, n(k))$.

$$S_L^{nb} \leftarrow S_L^{n(k)}$$

- *Caso 2. La distribución de S_L es extrema, siendo $\#(S_L)$ impar.*
 1. Se asignan lab_i etiquetas del extremo del nivel $l(i, n(i))$ a las lab_i etiquetas del extremo de S^{nb} .

$$\{s_0^{nb}, \dots, s_{lab_i-1}^{nb}\} \leftarrow \{s_0^{n(i)}, \dots, s_{lab_i-1}^{n(i)}\}$$

2. Se asignan el resto de etiquetas del subconjunto en el nivel $l(j, n(j))$.

$$\{s_{lab_i}^{nb}, \dots, s_{\#S_L-1}^{nb}\} \leftarrow \{s_{\frac{lab_i}{2}}^{n(j)}, \dots, s_{\frac{n(j)-1}{2}-1}^{n(j)}\}$$

3. Se realiza el puente modificando la semántica de la parte baja de la etiqueta $s_{lab_i}^{nb}$.

$$\underline{s_{lab_i}^{nb}} \leftarrow \underline{s_{lab_i}^{n(i)}}$$

- *Caso 3. La distribución de S_L es centrada, siendo $\#(S_L)$ impar.*

1. Se asignan lab_j etiquetas del extremo del nivel $l(j, n(j))$ a las lab_j etiquetas del extremo de S^{nb} .

$$\{s_0^{nb}, \dots, s_{lab_j-1}^{nb}\} \leftarrow \{s_0^{n(j)}, \dots, s_{lab_j-1}^{n(j)}\}$$

2. Se asignan el resto de etiquetas del subconjunto al nivel $l(i, n(i))$.

$$\{s_{lab_j}^{nb}, \dots, s_{\#S_L-1}^{nb}\} \leftarrow \{s_{lab_j \cdot 2}^{n(i)}, \dots, s_{\frac{n(i)-1}{2}-1}^{n(i)}\}$$

- Se realiza el puente la semántica de la etiqueta $s_{lab_j}^{nb}$.

$$\underline{s_{lab_j}^{nb}} \leftarrow \underline{s_{lab_j}^{n(j)}}$$

Representación de S_C

- *Caso 1. La distribución de S_L es uniforme.*

1. A la parte alta de la etiqueta central de S^{nb} se le asigna la parte alta de la etiqueta central del nivel $l(k, n(k))$.

$$\overline{s_c^{nb}} \leftarrow \overline{s_c^{n(k)}}$$

- *Caso 2. La distribución de S_L es extrema, siendo $\#(S_L)$ impar.*

1. A la parte alta de la etiqueta central de S^{nb} se le asigna la parte alta de la etiqueta central del nivel $l(j, n(j))$.

$$\overline{s_c^{nb}} \leftarrow \overline{s_c^{n(j)}}$$

- *Caso 3. La distribución de S_L es centrada, siendo $\#(S_L)$ impar.*

1. A la parte alta de la etiqueta central de S^{nb} se le asigna la parte alta de la etiqueta central del nivel $l(i, n(i))$.

$$\overline{s_c^{nb}} \leftarrow \overline{s_c^{n(i)}}$$

A continuación, mostramos distintos ejemplos que ilustran cada uno de los casos.

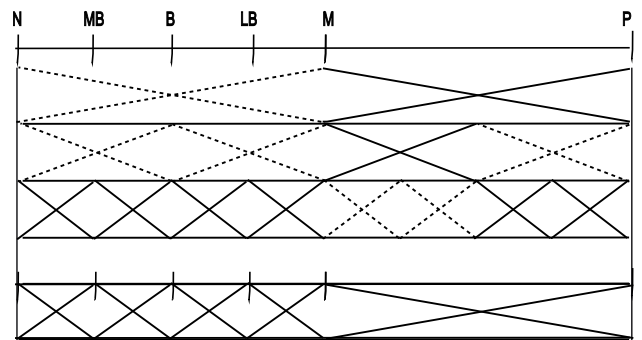


Figura 7: Caso 1. Representación de S_L

Por último, asumiendo el CTLNB de la Figura 4, el nuevo método representa el CTLNB de la forma que se observa en la Figura 10. Podemos observar como el método propuesto ha representado fielmente el CTLNB, utilizando la ELH *ad hoc* construida para dicho CTLNB.

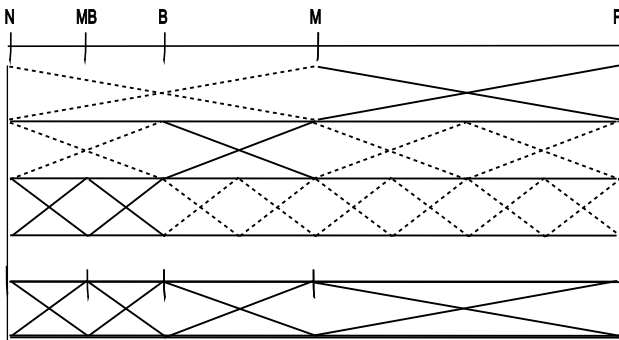
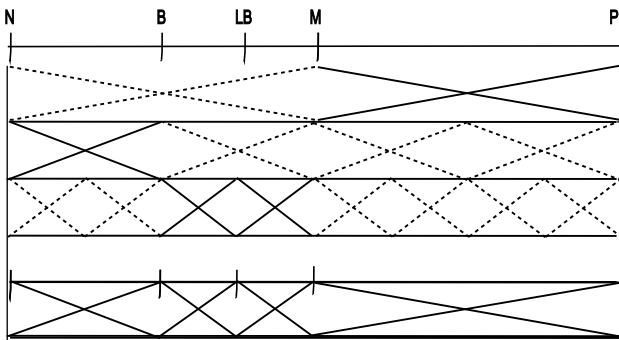
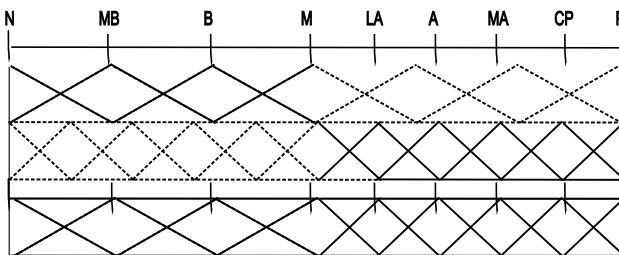
Figura 8: Caso 2. Representación de S_L Figura 9: Caso 3. Representación de S_L 

Figura 10: Representación del CTLNB utilizando el nuevo método

5. COMENTARIOS FINALES

En este trabajo, hemos desarrollado una nueva metodología que representa conjuntos de términos lingüísticos no balanceados, venciendo las limitaciones de la metodología revisada. Para ello, hemos utilizado una nueva estructura denominada extensión de las jerarquías lingüísticas. Nuestras líneas de actuación futuras se encaminan en desarrollar el modelo computacional para poder operar sobre las etiquetas del conjunto de términos lingüísticos no balanceado sin pérdida de información. Para ello, utilizaremos el último nivel de la jerarquía lingüística ampliada.

Agradecimientos

Este artículo ha sido realizado con la ayuda de los

Proyectos P08-TIC-3548, TIN2009-08286 y los Fondos Feder.

Referencias

- [1] F.J. Cabrerizo, S. Alonso, and E. Herrera-Viedma. A consensus model for group decision making problems with unbalanced fuzzy linguistic information. *International Journal of Information Technology and Decision Making*, 8(1):109–131, 2009.
- [2] M. Espinilla. Nuevos Modelos de Evaluación Sensorial con Información Lingüística. PhD thesis, Universidad de Jaén. <http://sinbad2.ujaen.es/sinbad2/files/TesisGrupo/TesisMacarena.pdf>, 2009.
- [3] F. Herrera, E. Herrera-Viedma, and L. Martínez. A hierarchical ordinal model for managing unbalanced linguistic term sets based on the linguistic 2-tuple model. In *Proceedings of the Eurofuse Workshop on Preference Modelling and Applications*, pages 201–206, Granada (Spain), April 2001.
- [4] F. Herrera, E. Herrera-Viedma, and L. Martínez. A fuzzy linguistic methodology to deal with unbalanced linguistic term sets. *IEEE Transactions on Fuzzy Systems*, (2), 2008.
- [5] F. Herrera and L. Martínez. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 8(6):746–752, 2000.
- [6] F. Herrera and L. Martínez. A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making. *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics*, 31(2):227–234, 2001.
- [7] E. Herrera-Viedma and A.G. López-Herrera. A model of information retrieval system with unbalanced fuzzy linguistic information. *International Journal of Intelligent Systems*, 22(11):1197–1214, 2007.
- [8] L. Martínez, M. Espinilla, and J. Liu. Extended linguistic hierarchies. *International Conference on Intelligent System and Knowledge Engineering*, Xiamen (China), 2008.
- [9] L.A. Zadeh. The concept of a linguistic variable and its applications to approximate reasoning. *Information Sciences, Part I, II, III*, 8,8,9:199–249,301–357,43–80, 1975.

A Multigranular Hierarchical Linguistic Model for Pharmacy and Therapeutics Committee

J. M. Doña¹ A. J. García² J. I. Peláez³

¹ Dept. of Information and Computer Tech., Hospital Univ. Virgen de la Victoria, 29010 Málaga, España
jesusm.dona.exts@juntadeandalucia.es

² Dept of Health Economics and Pharmacology, University of Malaga, 29071 Málaga, España
ajgr@uma.es

³ Dept. of Languages and Computer Science, University of Malaga, 29071 Málaga, España
jignacio@lcc.uma.es

Abstract

The pharmacy and therapeutics committee is one of the most important clinical commissions in Hospital environments. Its main function is the selection and evaluation of medicaments and therapies that should shape the hospital's pharmaceutical guide according with World Health Organization recommendations. These decision problems requires considering multiple evaluation aspects and criteria where different people are involved with different roles, valuations and preferences with the aim of analyzing a great number of factors and characteristics from a huge number of resources with imprecise assessments. We propose to manage this evaluation environment a multigranular hierarchical linguistic model to help the pharmacy and therapeutics committee in its decision problems.

Keywords: Group decision making, medical decision support systems, linguistic approach, multigranular information, majority operators.

1 INTRODUCTION

The decision process for therapies and drug selection is developed by one of the most important clinical commission in a Hospital: the pharmacy and therapeutics committee. Usually, this committee is composed by the pharmacy department head, the medical director, doctors of different departments (Oncology, Haematology, Emergency, Intensive care, Internal Medicine and Infectious Diseases), and primary care pharmacist of the health area. This commission represents a set of experts

involved in decision problems to develop the management health strategy of a hospital area. This decision making considers different criteria as pharmacotherapeutical, economics, efficiency and safety; processing complex information to make decisions about the inclusion/exclusion of therapies and drugs in a Hospital [1, 9].

According to World Health Organization (WHO), the pharmacy and therapeutics committee works in an environment which is a continuous and multidisciplinary process which should be based on the efficiency, safety, quality and costs to have a rational use of the health resources [15, 16]. Based on methodology used by the WHO, hospitals have developed their own essential drug lists, available therapies and health strategic which are supposed to be reviewed regularly in order to improve services quality and health resources.

In general, there are several issues involved in these decision problems. The first is the determination of evaluation aspects/criteria and their degrees of importance. This problem often needs to consider multiple evaluation criteria that are in a hierarchy. These evaluation criteria may also have different weights for different situations or medical scenarios. These problems need to extend the normal decision methods to deal with a hierarchical evaluation model with dynamic weights assigned, but also needs to consider the level of evaluators' experience and knowledge in dealing with the model.

Furthermore, some quantitative values can be obtained from specific medical reports and cohort studies as well. To integrate numerical values and linguistic evaluations can more precisely characterize medical quality.

Other issue is how to present and fuse linguistic values given by evaluators to each option under each criterion.

Evaluators may have different points of view about Hospital management, different preferences for different options, and different feelings for the same solution. These differences will directly impact on the final evaluation results. It is very hard to describe the feelings and preferences of evaluators by numbers, since they are often expressed in linguistic terms. Linguistic terms reflect conventional qualitative expressions and uncertainty, inaccuracy and fuzziness of human evaluation, linguistic variables and fuzzy sets techniques are very suitable for dealing with this situation [3, 10]. Therefore, based on the requirements of the hospital, the final ranking can be obtained through suitable fuzzy fusion of these individuals' viewpoints for all criteria.

In this work a multigranular hierarchical decision model is presented to deal with quantitative and qualitative values through linguistic valuations. Also, the problem of uniformity in the criteria valuations is solved using the 2-tuple fuzzy linguistic representation model for computing with words.

The paper is structured as follows: in section 2 the decision environment is determined, the hierarchical model is defined and the representation of information system is introduced; in section 4 the new decision model is presented. Finally, the conclusions are exposed.

2 DEFINITION OF DECISION ENVIRONMENT

The process by which health policy, therapies and drugs are selected is critical. It provides a framework within which the activities of the hospital area can be coordinated and it should be developed through a systematic process of consultation with all interested parties. In this process the objectives must be defined, priorities must be set, strategies must be developed and commitment must be built. An essential requirements list that is imposed from above will not reflect the needs of the users or be accepted by them.

It is therefore very important that:

- i. the process is consultative and transparent;
- ii. the selection criteria are explicit;
- iii. selection of the therapies and drugs are linked to evidence-based clinical guidelines;
- iv. and the clinical guidelines are divided into levels of care, and are regularly reviewed and updated.

This scenario is considered a very complex decision problem, where different people are involved with different roles, valuations and preferences with the aim of analyzing a great number of factors and characteristics

from a huge number of resources with imprecise assessments [4, 9, 14].

In these conditions, the information should be easy to obtain, so there must be established mechanisms to express their opinion in terms or expressions more usual to them, avoiding any imposition neither in the way of expression nor the number of values to be used to express themselves.

This brings a double necessity, on the one hand, to establish tools in the decision process that allow operating with the linguistic information, and, on the other hand, to use a methodology able to make a fusion process with information represented in different expression scales (multigranular linguistic information).

Figure 1 shows the process, a group of evaluators provides their opinions on a set of options using a hierarchical model. The weights of all aspects and criteria are determined based on the features of the available therapies, drugs, and medical scenarios. A weight described by a linguistic term is assigned to each evaluator. A data fusion method will be necessary to fuse all these data from decision hierarchy to obtain a final ranking.

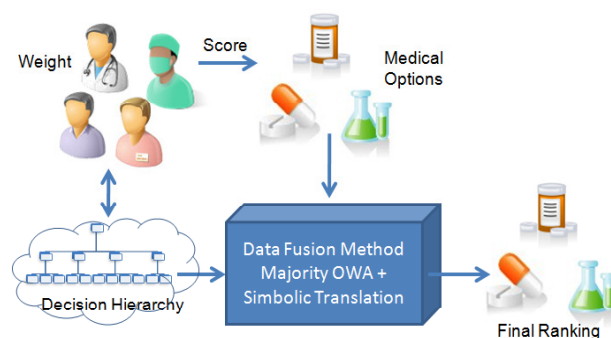


Figure 1: Representation of the decision environment.

2.1. MODEL SCHEME

The hierarchical model structure developed for the decision and evaluation problem in the pharmacy and therapeutics committee is based on two levels:

- I. The first level represents four main abstract attributes based on methodology used by the WHO; e.g. efficiency, adequacy, clinical efficacy, and safety.
- II. The second level represents the decomposition results of related aspect of the previous level, e.g. cost, budget impact, incremental cost-effective, etc. These criteria are more concrete evaluation attributes of each option and are

determined by Pharmacy and Therapeutics commission according to quality standards and norms of national health organization.

In Figure 2 the hierarchy developed for the decision model is shown for standard evaluation.

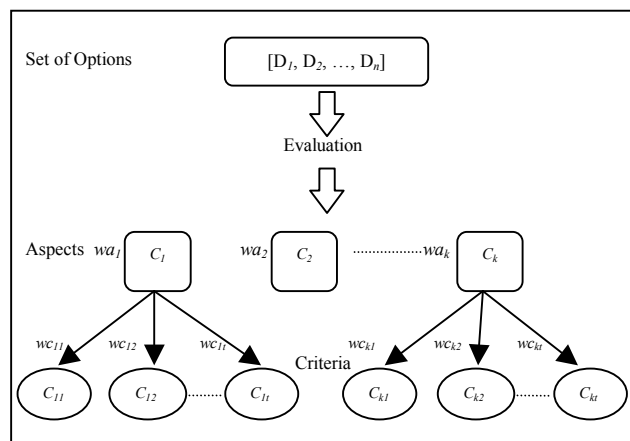


Figure 2. The hierarchical model.

2.2. USE OF MULTIGRANULAR LINGUISTIC INFORMATION

The main aim of establishing the linguistic descriptors of a linguistic variable is to supply the user with a few words by which he can express his information needs [8, 10, 17]. Hence the burden of qualifying a qualitative concept is eliminated. The fuzzy linguistic approach represents qualitative aspects as linguistic values by means of linguistic variable [5, 8, 10].

In order to accomplish this objective an important aspect to analyze is the “granularity of uncertainty”, i.e. the level of discrimination among different counts of uncertainty. The cardinality of this term set must be small enough so that it does not impose useless precision on the users and it must be rich enough in order to allow a discrimination of the assessments in a limited number of grades. Typical values of cardinality used in the linguistic models are odd ones such as 7 or 9 with an upper limit of granularity 11 or no more than 13 where the mid term represents an assessment of “approximately 0.5” and with the rest of the terms being placed symmetrically around it [2].

In the pharmacy and therapeutics committee the decision makers must be able to make their evaluation using their own expression domain for the weights and evaluation values. It is needed to determine the level of fuzziness, that is, the granularity of the labels, and to establish the semantic for the linguistic values.

To work with multigranular information the 2-tuple fuzzy linguistic representation model was presented in [6, 7], where different advantages of this approach are shown to represent the linguistic information over classical models.

The linguistic information is represented by means of 2-tuples (r_i, α_i) , $r_i \in S$ and $\alpha_i \in [-0.5, 0.5)$. S is the set of linguistic terms, r_i represents the linguistic label centre of the information and α_i is a numerical value that represents the translation from the original result β to the closest index label in the linguistic term set (r_i), i.e., the symbolic translation. This linguistic representation model defines a set of functions to make transformations among linguistic terms, 2-tuples and numerical values [6, 7].

A linguistic hierarchy is a set of levels, where each level is a linguistic term set with different granularity to the rest of levels of the hierarchy. Each level belonging to a linguistic hierarchy is denoted as $L(t, n(t))$, where t is a number that indicates the level of the hierarchy and $n(t)$ is the granularity of the linguistic term set of the level t . The levels belonging to a linguistic hierarchy are ordered according to their granularity. A graphical example of a linguistic hierarchy is shown in Figure 3.

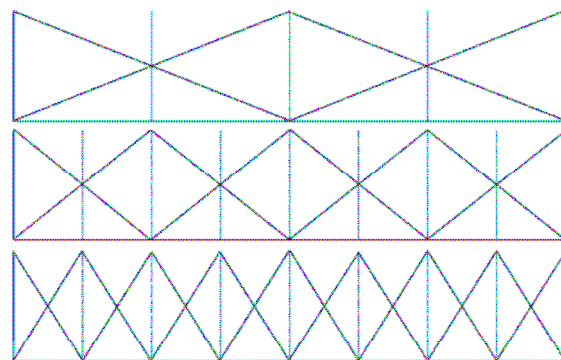


Figure 3. Linguistic hierarchy of 3, 5 and 9 labels.

From the above concepts, we shall define a linguistic hierarchy (LH) as the union of all levels t .

$$LH = \bigcup_t l(t, n(t)) \quad (1)$$

To build a linguistic hierarchy it must be taken into account that its hierarchical order is given by the increase of the granularity of the linguistic term sets in each level. Then the definition of S is extended to a set of linguistic term sets, $S_n(t)$, each term set belongs to a level of the hierarchy and has a granularity of uncertainty $n(t)$.

$$S^{n(t)} = \{s_0^{n(t)}, \dots, s_{n(t)-1}^{n(t)}\} \quad (2)$$

Generically, the linguistic term set of level $t + 1$ is obtained from its predecessor as

$$L(t, n(t)) \rightarrow L(t+1, 2 \cdot n(t) - 1) \quad (3)$$

The main problem for aggregating multigranular linguistic information is the loss of information produced in the normalization process. To avoid this problem, we shall use linguistic hierarchies term sets as multigranular linguistic contexts, but also we need transformation functions among the linguistic terms of the linguistic hierarchy term sets that carry out these transformation processes without loss of information.

The transformation function from a linguistic label in level t to a label in level $t + 1$, satisfying the linguistic hierarchy basic rules, is defined as

$$TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)}) = \Delta \left(\frac{\Delta^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t') - 1)}{n(t) - 1} \right) \quad (4)$$

The combination of 2-tuples and linguistics hierarchies allow us to fusion information without loss of information working with different expression domain.

2.3. LINGUISTIC OPERATOR

To solve the aggregation problem the operator must be capable of computing with words in the decision making process. Also, it must work with vague evaluations moved from the natural language to fuzzy terms or linguistic variables.

In order to add the information in the evaluation process the linguistic majority OWA operator LAMA will be used, because this operator is adapted for computing with words and is able to synthesize linguistic information in decision making environments producing results with a semantic of majority [11, 12, 13].

The LAMA operator is a mapping function $F: R^n \rightarrow R$ that has associated a weighting vector $W = [w_1, w_2, \dots, w_n]$.

Where $w_i \in [0, 1]$ and $\sum_{i=1}^n w_i = 1$.

$$LAMA(a_1, a_2, \dots, a_n) = b_1 \otimes w_1 \oplus b_2 \otimes w_2 \oplus \dots \oplus b_n \otimes w_n \quad (5)$$

with b_j is the j th largest element of the a_i , $\oplus$ is the sum of labels and $\otimes$ is the product of a label by a positive real.

The weight used in the LAMA operator is usually calculated from the majority process and importance function described in [12].

$$w_i = f_i(b_1, \dots, b_n) = \frac{\gamma_i^{\delta_{\min}}}{\theta_{\delta_{\max}} \cdot \dots \cdot \theta_{\delta_{\min}}} + \dots + \frac{\gamma_i^{\delta_{\max}}}{\theta_{\delta_{\max}}} \quad (6)$$

where

$$\gamma_i^k = \begin{cases} 1 & \text{if } \delta_i \geq k \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

and

$$\theta_i = \begin{cases} (\text{number of item with } \delta \geq i) + 1 & \text{if } i \neq \delta_{\min} \\ \text{number of item with } \delta \geq i & \text{otherwise} \end{cases} \quad (8)$$

The majority operators aggregate in function of δ_i that generally represents the importance of the element i . The calculation method for the value δ_i is independent from the definition of the majority operators.

3 A MULTIGRANULAR HIERARCHICAL LINGUISTIC MODEL FOR PHARMACY AND THERAPEUTICS COMMITTEE

To handle the above decision problem this paper defines a multigranular hierarchical linguistic model. This method applies fuzzy sets techniques to deal with linguistic terms used in the weights of aspects and criteria, the weights of experts and the evaluation scores given by evaluators. It can integrate human linguistics values with quantitative parameters to calculate the relevance degrees. Finally a fusion technique is applied to rank all options.

The evaluation model is composed of five steps on two levels.

Level 1: Determine the decision scenario.

1. Determine the expression of evaluators and linguistic hierarchies.

Suppose an evaluation group with m ($m \geq 1$) evaluators:

$$E = \{e_1, e_2, \dots, e_m\}$$

A set of n ($n \geq 1$) samples to evaluate:

$$D = \{D_1, D_2, \dots, D_n\}$$

Also, a two-level evaluation model, as shown in Figure 2, has k aspects and k_i criteria.

$$C = \{C_1, C_2, \dots, C_k\}$$

$$C_k = \{C_{k1}, C_{k2}, \dots, C_{ki}\}$$

2. Determine importance degrees of evaluators.

Evaluators play different roles in the pharmacy and therapeutic commission and have different degrees of influence on the ranking of a set of drugs. The weights are determined through discussions in the group or assigned by a higher management level before the evaluation process.

$$We = \{we_1, \dots, we_m\}$$

3. Determine the weights of evaluation aspects and criteria for each expert.

Let $WA = \{wa_1, wa_2, \dots, wa_k\}$ be the weights of the aspects in the evaluation model. For each aspect C_k , let $WC_k = \{wc_{k1}, wc_{k2}, \dots, wc_{ki}\}$, be the weights for the set of criteria on level 2 of Figure 2.

In this level all weights are described by a set of linguistic terms. The evaluators can use their own expression granularity through the 2-tuple linguistic model and the symbolic translation.

At the end of this step a set of individual weighted hierarchy for each expert is obtained.

$$H = \{H_1, \dots, H_m\}$$

Level 2: Aggregation of values

4. Individual evaluation.

The relevance degree of each option on each criterion is set up based on two different kinds of evaluations, i.e. the subjective evaluations and objective values. Subjective evaluations are obtained from human evaluators; while objective values are collected through specific medical reports and/or cohort studies.

For objective values, the relevance degree is obtained using a transformation from numeric values into linguistic terms through 2-tuples functions.

The opinion values are aggregated for each evaluation aspect using the LAMA operator. In this case, the weight of each criterion represents the value of the importance function of the majority process [6, 7].

Following the aspects are aggregated in the same way with the OWA operator. Finally the obtained result represents the individual evaluation of each expert over each item.

$$IV_m = (IV_1^m, IV_2^m, \dots, IV_n^m)$$

5. Group evaluation.

The final group value is obtained through the majority process and the information of the individual evaluation.

The aggregation process applies the OWA operator using the importance degree of evaluators of step 2 as majority importance function.

$$GV_n = LAMA(IV_n^1, IV_n^2, \dots, IV_n^m) \quad (9)$$

At the end of the process all options are ordered by the linguistic aggregation value and can be considered for the final committee decision in the particular medical scenario.

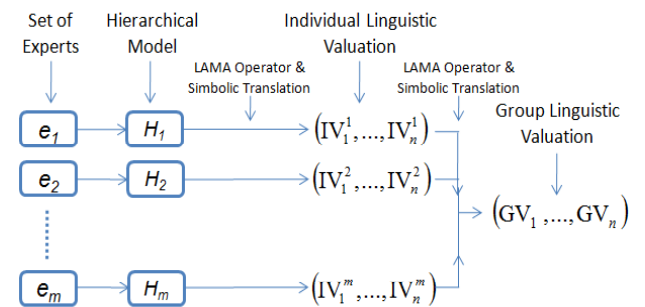


Figure 4. Representation of decision model.

4 CONCLUSIONS

The pharmacy and therapeutics committee makes important issues in Hospital management. This commission is involved in complex decision situation in which criteria are within a hierarchy structure and both subjective and objective data are considered. In particular, these data are often vague rather than in crisp numbers and in different dimensions.

We have presented a multigranular linguistic decision model to help the pharmacy and therapeutics committee in management problems. A decision hierarchy have been developed for health environments using two levels which

join the WHO and national health organization recommendations. This model uses the advantages of the 2-tuples fuzzy linguistic representation model and majority OWA operator to improve a natural language system to experts. It can integrate human linguistics values with quantitative parameters to calculate the relevance degrees of solutions and help and support the pharmacy and therapeutics committee decisions.

This decision system is being tested in the Hospital Universitario Virgen de la Victoria from Málaga, Spain.

Acknowledgements

Pharmacy department of Hospital Universitario Virgen de la Victoria from Málaga, Spain.

Economics and Pharmacology department of University of Malaga, Spain.

This work has been partially supported by the Research Project TIN2006-14285.

References

- [1] Ascione F.J. Principles of scientific literature evaluation: critiquing clinical drug trials. Washington DC: American Pharmaceutical Association. 2001.
- [2] Bonissone P. P. and Decker K. S. Selecting uncertainty calculi and granularity: An experiment in trading-off precision and complexity in Trading-off Precision and Complexity, in: L.H. Kanal and J.F. Lemmer, Eds., *Uncertainty in Artificial Intelligence*, 217-247. 1986.
- [3] Bordogna G., Fedrizzi M., Pasi G. A linguistic modeling of consensus in group decision making based on OWA operators. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, 27(1): 126-132, 1997.
- [4] Gibson, R. Knowledge management support for decision making in the pharmaceutical industry. In *Decision Making Support Systems: Achievements, Trends and Challenges For*. IGI Publishing, Hershey, PA, 143-156. 2003.
- [5] Herrera-Viedma E., Martínez L., Mata F., and Chiclana F. A consensus support system model for group decision-making problems with multigranular linguistic preference relations. *IEEE Transactions on Fuzzy Systems*, 13(5):644-658, 2005.
- [6] Herrera F. and Martínez L. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 8(6):746-752, 2000.
- [7] Herrera F. and Martínez L. A model based on linguistic 2-tuples for dealing with multigranular hierarchical linguistic context in multi-expert decision making. *IEEE Transactions on Systems, Man, And Cybernetics - Part B: Cybernetics*, 31(2):227-234, 2001.
- [8] Lu, J., Zhang, G., Ruan, D. & Wu, F. *Multi-objective Group Decision Making: Methods, Software and Applications with Fuzzy Set Technology*, London: Imperial College Press. 2007.
- [9] Mannebach M.A., Ascione F.J., Gaither Ca, et al. Activities, functions, and structure of pharmacy and therapeutics committees in large teaching hospitals. *Am J Health-Syst Pharm*; 56:622-8. 1999.
- [10] Martínez L., Liu J., Yang J.-B., and Herrera F. A multigranular hierarchical linguistic model for design evaluation based on safety and cost analysis. *International Journal of Intelligent Systems*, 20(12):1161-1194. 2005.
- [11] Marimin, Umano M., Hatono I., Tamura H. Linguistic Labels for Expressing Fuzzy Preference Relations in Fuzzy Group Decision Making. *Systems, Man, and Cybernetics, Part B: Cybernetics*, *IEEE Transactions* 28(2), 205 – 218. 1998.
- [12] Peláez J. I., Doña J. M. Majority Additive-Ordered Weighting Averaging: A New Neat Ordered Weighting Averaging Operators Based on the Majority Process. *International Journal of Intelligent Systems*. 18(4): 469-481, 2003.
- [13] Peláez J. I., Doña J. M. LAMA: A Linguistic Aggregation of Majority Additive Operator, *International Journal of Intelligent Systems* 18(7): 809-820. 2003.
- [14] Peláez J. I., Doña J. M., Gómez-Ruiz J. A. Analysis of OWA Operators in Decision Making for Modeling the Majority Concept. *Applied Mathematics and Computation*. 186: 1263-1275, 2007.
- [15] Santos B, Piña MJ, Carvajal E, et al. Decision analysis applied to the selection of angiotensin-converting enzyme inhibitors. *PWS*; 15: 219-224. 1993.
- [16] WHO. The use of essential drugs. Ninth report of the WHO Expert Committee. WHO Technical Report Series No.895. Geneva: World Health Organization. 2000.
- [17] WHO. How to Develop and Implement a National Drug Policy (Second Edition). Geneva. World Health Organization. 2001.
- [18] Zadeh L. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems*, 90:111-127. 1997.

FUZZY DECISION MAKING WITH THE PROBABILITY, THE WEIGHTED AVERAGE AND THE OWA OPERATOR

José M. Merigó

Department of Business Administration, University of Barcelona,
{jmerigo}@ub.edu

Abstract

We develop a new decision making model based on the use of a unified approach between the probability, the weighted average and the ordered weighted averaging (OWA) operator in an uncertain environment that can be assessed with fuzzy numbers (FNs). We present a new aggregation operator called the fuzzy probabilistic ordered weighted averaging – weighted averaging (FPOWAWA) operator. We study its main properties and we find a wide range of particular cases such as the fuzzy probabilistic OWA, the fuzzy OWAWA, the fuzzy probabilistic weighted averaging operator and the fuzzy OWA operator. We also develop an application of the new approach in a decision making model about political management.

Keywords: Decision making, Probability, OWA operator, Weighted average, Fuzzy numbers.

1 INTRODUCTION

Decision making models are very useful in a lot of disciplines [4,6-12,15,17-18]. Some of the most common techniques for dealing with decision making are the probability and the weighted average (WA) [7]. Usually, they represent some kind of belief that the decision maker has about the problem, being the probability the objective perspective and the WA the subjective perspective. Another useful tool for decision making is the ordered weighted averaging (OWA) operator [15]. It is an aggregation operator that provides a parameterized family of aggregation operators between the maximum and the minimum. Thus, it is able to underestimate or overestimate the initial results of the decision maker according to a degree of optimism or pessimism. Since its

appearance, the OWA operator has been studied by a lot of authors [1,4,6-18].

Several studies have tried to use the WA and the probability in the OWA operator. In [13], Torra developed a model for using OWAs and WAs in the same formulation called the weighted OWA (WOWA) operator. Later [14], Xu and Da developed another approach called the hybrid averaging (HA) operator. The use of the OWA operator in the probability has been analyzed in several papers. It is worth noting the work developed by [4,17] about immediate probabilities that has also been studied in [6-7,18]. Other studies that have used probabilistic information and OWAs are found in the analysis of the Dempster-Shafer theory of evidence [7,18]. More recent approaches for unifying the OWA with the probability and the WA are the probabilistic OWA (POWA) operator and the OWA weighted average (OWAWA) operator introduced in [7-9]. Another interesting approach is the one that unifies the probability with the weighted average, known as the probabilistic weighted averaging (PWA) operator [10]. A more general approach to the previous ones is the one suggested by Merigó in [11]. He suggested a unified model between the probability, the weighted average and the OWA operator. This new model is the probabilistic ordered weighted averaging – weighted averaging (POWAWA) operator.

Usually, when using these approaches it is considered that the available information are exact numbers. However, this may not be the real situation found in the specific problem considered. Sometimes, the available information is vague or imprecise and it is not possible to analyze it with exact numbers. Therefore, it is necessary to use another approach that is able to assess the uncertainty such as the use of fuzzy numbers (FNs). In order to develop the fuzzy approach, we will follow the ideas of [2-3,5,7,19-20].

The aim of this paper is to develop a new extension of the POWAWA operator for uncertain situations that can be represented with FNs. We will call it the fuzzy POWAWA (FPOWAWA) operator. The main advantage

of this new approach is that it is able to unify the probability, the WA and the OWA in the same framework and considering the degree of importance that each concept has in the analysis. Moreover, this approach deals with uncertain environments where the available is not clear and it is necessary to use another approach such as the FNs. By using FNs, we are able to represent the information in a more complete way because we can consider the maximum and the minimum results and the possibility that the internal results of the fuzzy interval will occur.

We also study the applicability of the new approach and we see that it is very broad because it can be implemented in statistics, economics, engineering, etc. We focus on a decision making application about political management. We study a decision process where a national government is analyzing its optimal fiscal policy.

The paper is structured as follows. In Section 2, we briefly review some basic concepts. Section 3 presents the FPOAWA operator. Section 4 develops a numerical example of the new approach and in Section 5 we summarize the main conclusions of the paper.

2 PRELIMINARIES

2.1. FUZZY NUMBERS

A FN A is defined as a fuzzy subset of a universe of discourse that is both convex (i.e., $\mu_A(\lambda x_1 + (1 - \lambda)x_2) \geq \min(\mu_A(x_1), \mu_A(x_2))$; for $\forall x_1, x_2 \in R$ and $\lambda \in [0, 1]$) and normal (i.e., $\sup_{x \in R} \mu_A(x) = 1$).

Note that the FN may be considered as a generalization of the interval number although it is not strictly the same because the interval numbers may have different meanings. In the literature, we find a wide range of FNs [2-3,5,7,19] such as the Triangular FN (TFN), the Trapezoidal FN (TpFN), the Interval-Valued FN (IVFN), the Generalized FN (GFN), and a lot of more complex structures.

For example, a TpFN A of a universe of discourse R can be characterized by a trapezoidal membership function (α -cut representation) $A = (\underline{a}, \bar{a})$ such that

$$\begin{aligned}\underline{a}(\alpha) &= a_1 + \alpha(a_2 - a_1), \\ \bar{a}(\alpha) &= a_4 - \alpha(a_4 - a_3).\end{aligned}\quad (1)$$

where $\alpha \in [0, 1]$ and parameterized by (a_1, a_2, a_3, a_4) where $a_1 \leq a_2 \leq a_3 \leq a_4$, are real values. Note that if $a_1 = a_2 = a_3 = a_4$, then, the FN is a crisp value and if $a_2 = a_3$, the FN is represented by a TFN. Note that the TFN can be parameterized by (a_1, a_2, a_4) .

In the following, we are going to review some basic FN arithmetic operations as follows. Let A and B be two TFNs, where $A = (a_1, a_2, a_3)$ and $B = (b_1, b_2, b_3)$.

1. $A + B = (a_1 + b_1, a_2 + b_2, a_3 + b_3)$
2. $A - B = (a_1 - b_3, a_2 - b_2, a_3 - b_1)$
3. $A \times k = (k \times a_1, k \times a_2, k \times a_3)$; for $k > 0$.

Note that other operations could be studied but in this paper we will focus on these ones. For more complete information and overviews about FNs, see for example [2-3,5,7,19].

2.2. THE OWA OPERATOR

The OWA operator [15] provides with a parameterized family of aggregation operators that include the arithmetic mean, the maximum and the minimum. It can be defined as follows.

Definition 1. An OWA operator of dimension n is a mapping $OWA: R^n \rightarrow R$ that has an associated weighting vector W of dimension n such that the sum of the weights is 1 and $w_j \in [0, 1]$, then:

$$OWA(a_1, a_2, \dots, a_n) = \sum_{j=1}^n w_j b_j \quad (2)$$

where b_j is the j th largest of the a_i .

2.3. THE POWAWA OPERATOR

The probabilistic OWA weighted average (POWAWA) operator is an aggregation operator that uses probabilities, weighted averages and OWAs in the same formulation. It unifies these three concepts considering the degree of importance we want to give to each case depending on the situation considered. It can be defined as follows [7,11].

Definition 2. A POWAWA operator of dimension n is a mapping $POWAWA: R^n \rightarrow R$ that has an associated weighting vector W of dimension n with $w_j \in [0, 1]$ and $\sum_{j=1}^n w_j = 1$, such that:

$$POWAWA(a_1, \dots, a_n) = \sum_{j=1}^n \hat{v}_j b_j \quad (3)$$

where b_j is the j th largest of the a_i , each argument a_i has an associated weight v_i with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, a probability p_i with $\sum_{i=1}^n p_i = 1$ and $p_i \in [0, 1]$,

$\hat{v}_j = C_1 w_j + C_2 v_j + C_3 p_j$, with C_1, C_2 and $C_3 \in [0, 1]$, $C_1 + C_2 + C_3 = 1$, and v_j and p_j are the weights v_i and p_i ordered according to b_j , that is to say, according to the j th largest of the a_i .

3 THE FUZZY POWAWA OPERATOR

The FPOWAWA operator is an aggregation operator that unifies the probability, the WA and the OWA in the same formulation and considering the degree of importance that each concept has in the analysis. Thus, we can consider the objective and subjective information in the same problem and we are able to underestimate or overestimate the results according to our interests. Moreover, by using FNs we can represent the information considering the minimum and maximum results that may occur in the uncertain environment and the possibility (or membership level) that the internal values of the fuzzy interval will occur. The FPOWAWA operator can be defined as follows.

Definition 3. Let Ψ be the set of FNs. A FPOWAWA operator of dimension n is a mapping FPOWAWA: $\Psi^n \rightarrow \Psi$ that has an associated weighting vector W of dimension n with $w_j \in [0, 1]$ and $\sum_{j=1}^n w_j = 1$, such that:

$$\text{FPOWAWA}(\tilde{a}_1, \dots, \tilde{a}_n) = \sum_{j=1}^n \hat{v}_j b_j \quad (4)$$

where b_j is the j th largest of the $\tilde{a}_i$, each argument $\tilde{a}_i$ has an associated weight v_i with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, a probability p_i with $\sum_{i=1}^n p_i = 1$ and $p_i \in [0, 1]$, $\hat{v}_j = C_1 w_j + C_2 v_j + C_3 p_j$, with C_1, C_2 and $C_3 \in [0, 1]$, $C_1 + C_2 + C_3 = 1$, and v_j and p_j are the weights v_i and p_i ordered according to b_j , that is to say, according to the j th largest of the $\tilde{a}_i$.

Note that this definition could also be presented using the following equivalent definition.

Definition 4. Let Ψ be the set of FNs. A FPOWAWA operator of dimension n is a mapping FPOWAWA: $\Psi^n \rightarrow \Psi$, that has an associated weighting vector W , with $\sum_{j=1}^n w_j = 1$ and $w_j \in [0, 1]$, a weighting vector V , with $\sum_{i=1}^n v_i = 1$ and $v_i \in [0, 1]$, and a probabilistic vector P , with $\sum_{i=1}^n p_i = 1$ and $p_i \in [0, 1]$, such that:

FPOWAWA $(\tilde{a}_1, \dots, \tilde{a}_n) =$

$$C_1 \sum_{j=1}^n w_j b_j + C_2 \sum_{i=1}^n v_i \tilde{a}_i + C_3 \sum_{i=1}^n p_i \tilde{a}_i \quad (5)$$

where b_j is the j th largest of the $\tilde{a}_i$, the $\tilde{a}_i$ are FNs and C_1, C_2 and $C_3 \in [0, 1]$ with $C_1 + C_2 + C_3 = 1$.

As we can see, the OWA, the WA and the probability are included in this formulation as special cases.

- If $C_1 = 1$, we get the usual FOWA operator.
- If $C_2 = 1$, we get the usual fuzzy WA (FWA).
- If $C_3 = 1$, we get the usual fuzzy probability.
- If $C_1 = 0$, we form the fuzzy probabilistic weighted average (FPWA).
- If $C_2 = 0$, we form the fuzzy probabilistic OWA (FPOWA) operator.
- If $C_3 = 0$, we form the FOWAWA operator.

Note that it is possible to distinguish between the descending FPOWAWA (DFPOWAWA) and the ascending FPOWAWA (AFPOWAWA) operator by using $w_j = w_{n-j+1}^*$, where w_j is the j th weight of the DFPOWAWA and w_{n-j+1}^* the j th weight of the AFPOWAWA operator.

Note that if the weighting vectors of the three concepts are not normalized, i.e., $W = \sum_{j=1}^n w_j \neq 1$, $V = \sum_{i=1}^n v_i \neq 1$, $P = \sum_{i=1}^n p_i \neq 1$, then, the FPOWAWA operator can be expressed as:

FPOWAWA $(\tilde{a}_1, \dots, \tilde{a}_n) =$

$$\frac{C_1}{W} \sum_{j=1}^n w_j b_j + \frac{C_2}{V} \sum_{i=1}^n v_i \tilde{a}_i + \frac{C_3}{P} \sum_{i=1}^n p_i \tilde{a}_i \quad (6)$$

The FPOWAWA is monotonic, bounded and idempotent. Note that it is interesting to mention that with the FPOWAWA operator, we get new boundary conditions based on the minimum and the maximum probabilistic weighted aggregation. Other properties and particular cases will be considered in future research.

Note that it is possible to develop a more general formulation of the FPOWAWA operator that uses other concepts apart from the probability, the WA and the OWA. We believe that the FPOWAWA operator is a very natural unification but other unifications in a more artificial way could be developed such as the use of this

formulation with the minimization of regret approach, the analytic hierarchy process (AHP), the TOPSIS approach, the utility model and a lot of other situations [7]. For doing so, instead of using a convex combination with three concepts we could use the following definition for m concepts.

Definition 5. Let Ψ be the set of FNs. A unified aggregation operator of dimension m is a mapping FUA: $\Psi^n \rightarrow \Psi$, that has associated m concepts with a degree of importance of C_h such that:

$$\text{FUA}(\tilde{a}_1, \dots, \tilde{a}_n) = \sum_{h=1}^m C_h K_h \quad (7)$$

where C_h is the degree of importance that each concept has in the aggregation, with $C_h \in [0, 1]$ and $\sum_{h=1}^m C_h = 1$, the arguments $\tilde{a}_i$ are FNs and K_h is the h concept considered in the aggregation.

As we can see, with this formulation we can unify in the same aggregation all the concepts we want. For example, with the FPOWAWA operator we have $h = 3$ and $K = 1$ is the OWA, $K = 2$ is the WA and $K = 3$ is the probabilistic aggregation. Note that these unified aggregations will be studied in future research but as we can see they follow a similar methodology than the POWAWA although we expect to find some other properties and particular cases.

Focussing on the FPOWAWA, if B is a vector corresponding to the ordered arguments b_j , we shall call this the ordered argument vector, and W^T is the transpose of the weighting vector, then, the FPOWAWA operator can be represented as follows:

$$\text{FPOWAWA}(\tilde{a}_1, \dots, \tilde{a}_n) = W^T B \quad (8)$$

A further interesting issue of the FPOWAWA operator is to analyze different particular cases such as the following ones obtained by using different representations in the weighting vectors.

- The fuzzy arithmetic PWA aggregation (if $w_j = 1/n$, for all j).
- The fuzzy arithmetic POWA operator (if $v_i = 1/n$, for all i).
- The fuzzy arithmetic OWAWA operator (if $p_i = 1/n$, for all i).
- The fuzzy double arithmetic OWA operator (if $p_i = 1/n$, for all i , and $v_i = 1/n$, for all i).
- The fuzzy double arithmetic WA operator (if $p_i = 1/n$, for all i , and $w_j = 1/n$, for all j).
- The fuzzy double arithmetic probabilistic aggregation (if $v_i = 1/n$, for all i , and $w_j = 1/n$, for all j).

- The fuzzy average (if $v_i = 1/n$, for all i , $p_i = 1/n$, for all i , and $w_j = 1/n$, for all j).
- The fuzzy maximum probabilistic weighted aggregation ($w_1 = 1$ and $w_j = 0$, for all $j \neq 1$).
- The fuzzy minimum probabilistic weighted aggregation ($w_n = 1$ and $w_j = 0$, for all $j \neq n$).
- The fuzzy maximum arithmetic probabilistic aggregation ($w_1 = 1$ and $w_j = 0$, for all $j \neq 1$, and $v_i = 1/n$, for all i).
- The fuzzy maximum arithmetic WA aggregation ($w_1 = 1$ and $w_j = 0$, for all $j \neq 1$, and $p_i = 1/n$, for all i).
- The fuzzy minimum arithmetic probabilistic aggregation ($w_n = 1$ and $w_j = 0$, for all $j \neq n$, and $v_i = 1/n$, for all i).
- The fuzzy minimum arithmetic WA aggregation ($w_n = 1$ and $w_j = 0$, for all $j \neq n$, and $p_i = 1/n$, for all i).
- The FPOWAWA Hurwicz criteria ($w_1 = \alpha$, $w_n = 1 - \alpha$ and $w_j = 0$, for all $j \neq 1, n$).
- The step-FPOWAWA ($w_k = 1$ and $w_j = 0$, for all $j \neq k$).
- The olympic-FPOWAWA operator ($w_1 = w_n = 0$, and $w_j = 1/(n-2)$ for all others).
- The general olympic-FPOWAWA operator ($w_j = 0$ for $j = 1, 2, \dots, k, n, n-1, \dots, n-k+1$; and for all others $w_{j^*} = 1/(n-2k)$, where $k < n/2$).
- The S-FPOWAWA ($w_1 = (1/n)(1 - (\alpha + \beta) + \alpha)$, $w_n = (1/n)(1 - (\alpha + \beta) + \beta)$, and $w_j = (1/n)(1 - (\alpha + \beta))$ for $j = 2$ to $n-1$ where $\alpha, \beta \in [0, 1]$ and $\alpha + \beta \leq 1$).
- The centered-FPOWAWA (if the weighting vector W is symmetric, strongly decaying from the center to the maximum and the minimum, and inclusive).
- Etc.

Note that the FPOWAWA operator also includes the POWAWA operator as a particular case. Therefore, all this particular cases are also included in this approach. Furthermore, the uncertain POWAWA (UPOWAWA) operator is also included in this approach when the FNs are reduced to the interval numbers. These two cases are proved with the following theorems.

Theorem 1. If the FNs are reduced to the usual exact numbers, then, the FPOWAWA operator becomes the POWAWA operator [7,11].

Proof. Assume a TpFN = (a_1, a_2, a_3, a_4) . If $a_1 = a_2 = a_3 = a_4$, then $(a_1, a_2, a_3, a_4) = a$, thus, we get the POWAWA operator.

Remark 1. In a similar way, we could develop the same proof for all the other types of FNs available in the literature [2-3,5,7,19].

Theorem 2. If the FNs are reduced to the interval numbers, then, the FPOWAWA operator becomes the uncertain POWAWA (UPOWAWA) operator [7].

Proof. Assume a TpFN = (a_1, a_2, a_3, a_4) . If we only consider the points (a_1, a_2, a_3, a_4) , then, the FN becomes an interval number (a quadruplet). Therefore, the FPOWAWA operator becomes the UPOWAWA operator.

Remark 2. In a similar way, we could develop the same proof for all the other types of FNs.

Remark 3. Note that similar analysis could be developed for considering situations when the FNs are representing linguistic variables, probabilistic sets, expertons, etc.

Note also that other families of FPOWAWA operators may be used following a similar methodology as it has been developed for the OWA operator and its extensions [1,4,6-18]. Moreover, we could extend this analysis to other types of uncertain approaches such as the use of interval numbers, linguistic approaches and expertons [7].

4 ILLUSTRATIVE EXAMPLE

The FPOWAWA operator and more general unifications can be applied in an unlimited number of applications because all the previous studies that have used these concepts (probability, WA and OWA) can be revised and extended with this new approach. Moreover, this approach is very general because it can represent the information by using FNs and all its particular cases such as the usual exact numbers.

Summarizing some of the main fields where it is possible to apply the POWA operator, we can mention: Statistics,

Mathematics, Economics, Decision Theory, Engineering, Physics, Soft Computing, etc.

In this paper, we focus on an application in decision making about selection of fiscal policies. We analyze a country that is looking for its optimal fiscal policy for the next year and they consider 5 alternatives.

- A_1 = Develop a strong expansive fiscal policy.
- A_2 = Develop an expansive fiscal policy.
- A_3 = Do not develop any change.
- A_4 = Develop a contractive fiscal policy.
- A_5 = Develop a strong contractive fiscal policy.

In order to evaluate these policies, the government has brought together a group of experts. This group considers that the key factor is the economic situation of the world economy for the next period. They consider 5 possible states of nature that could happen in the future: N_1 = Very bad economic situation, N_2 = Bad economic situation, N_3 = Regular economic situation, N_4 = Good economic situation, N_5 = Very good economic situation.

The results of the available policies, depending on the state of nature N_i and the alternative A_k that the decision maker chooses, are shown in Table 1.

In this example, the experts assume the following weighting vectors: $W = (0.1, 0.2, 0.2, 0.2, 0.3)$; $V = (0.3, 0.2, 0.2, 0.2, 0.1)$; $P = (0.3, 0.3, 0.2, 0.1, 0.1)$. Note that they assume that the OWA operator has an importance of 20%, the WA of 40% and the probabilistic information of 40%. With this information, we can aggregate the expected results in order to make a decision. In Table 2, we present different results by using different types of FPOWAWA operators.

Table 1: Available investment alternatives

	N_1	N_2	N_3	N_4	N_5
A_1	(60,70,80)	(40,50,60)	(30,40,50)	(70,80,90)	(80,90,100)
A_2	(20,30,40)	(10,20,30)	(80,90,100)	(90,100,110)	(70,80,90)
A_3	(60,70,80)	(60,70,80)	(50,60,70)	(30,40,50)	(60,70,80)
A_4	(80,90,100)	(20,30,40)	(30,40,50)	(40,50,60)	(80,90,100)
A_5	(90,100,110)	(20,30,40)	(40,50,60)	(60,70,80)	(60,70,80)

Table 2: Aggregated results

	FProb.	FWA	FOWA	FPOWAWA
A_1	(51,61,71)	(54,64,74)	(51,61,71)	(52.2,62.2,72.2)
A_2	(41,51,61)	(49,59,69)	(46,56,66)	(45.2,55.2,65.2)
A_3	(55,65,75)	(52,62,72)	(49,59,69)	(52.6,62.6,72.6)
A_4	(48,58,68)	(50,60,70)	(44,54,64)	(48,58,68)
A_5	(53,63,73)	(57,67,77)	(47,57,67)	(53.4,63.4,73.4)

In this case, we get as optimal policy A_5 for the FWA and the FPOWAWA, A_3 for the probabilistic approach and A_1 for the FOWA.

5 CONCLUSIONS

We have introduced the FPOWAWA operator. It is a new aggregation operator that unifies the probability, the WA and the OWA in the same formulation and considering the degree of importance that each concept has in the aggregation. Moreover, we have used FNs which are very useful for assessing uncertain environments considering the minimum and maximum results and the possibility that the internal values of the fuzzy interval will occur. We have studied some of its main properties and particular cases and we have developed a more general unified aggregation operator by using more concepts in the aggregation. We have also developed an application of the new approach in political management. Moreover, we have seen that the applicability of the FPOWAWA and more general aggregations is very broad because most of the sciences are affected by these concepts.

In future research, we expect to develop further developments of this approach by adding more characteristics such as the use of quasi-arithmetic means, distance measures and other types of uncertain information. We will also consider more general formulations by using more general unifications and other applications such as in group decision making.

References

- [1] G. Beliakov, A. Pradera and T. Calvo, *Aggregation Functions: A Guide for Practitioners*. Berlin: Springer-Verlag, 2007.
- [2] S.S.L. Chang and L.A. Zadeh, On fuzzy mapping and control. *IEEE Transactions on Systems, Man and Cybernetics*, 2:30-34, 1972.
- [3] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, New York: Academic Press, 1980.
- [4] K.J. Engemann, D.P. Filev and R.R. Yager, Modelling decision making using immediate probabilities. *International Journal of General Systems*, 24:281-294, 1996.
- [5] A. Kaufmann and M.M. Gupta, *Introduction to fuzzy arithmetic*, Rheinhold: Publications Van Nostrand, 1985.
- [6] J.M. Merigó, Using immediate probabilities in fuzzy decision making. In: *Proceedings of the 22nd ASEPELT Conference*, Barcelona, Spain, pp. 1650-1664, 2008.
- [7] J.M. Merigó, *New extensions to the OWA operators and its application in decision making* (In Spanish). PhD Thesis, Department of Business Administration, University of Barcelona, 2008.
- [8] J.M. Merigó, On the use of the OWA operator in the weighted average and its application in decision making. In: *Proceedings of the WCE 2009 Conference*, London, United Kingdom, pp. 82-87, 2009.
- [9] J.M. Merigó, Probabilistic decision making with the OWA operator and its application in investment management. In: *Proceedings of the IFSA-EUSFLAT Conference*, Lisbon, Portugal, pp. 1364-1369, 2009.
- [10] J.M. Merigó, The probabilistic weighted averaging operator and its application in decision making. In: *Proceedings of the Intersymp Conference*, Baden-Baden, Germany, pp. 55-58, 2009.
- [11] J.M. Merigó, On the unification between the probability, the weighted average and the OWA operator. In: *Proceedings of the ISKE Conference*, Hasselt, Belgium, pp. 375-380, 2009.
- [12] J.M. Merigó and A.M. Gil-Lafuente, The induced generalized OWA operator. *Information Sciences*, 179:729-741, 2009.
- [13] V. Torra, The weighted OWA operator. *International Journal of Intelligent Systems*, 12:153-166, 1997.
- [14] Z.S. Xu and Q.L. Da, An overview of operators for aggregating information. *International Journal of Intelligent Systems*, 18:953-968, 2003.
- [15] R.R. Yager, On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man and Cybernetics B*, 18:183-190, 1988.
- [16] R.R. Yager, Families of OWA operators. *Fuzzy Sets and Systems*, 59:125-148, 1993.
- [17] R.R. Yager, K.J. Engemann and D.P. Filev, On the concept of immediate probabilities, *International Journal of Intelligent Systems*, 10:373-397, 1995.
- [18] R.R. Yager and J. Kacprzyk, *The Ordered Weighted Averaging Operators: Theory and Applications*. Norwell: Kluwer Academic Publishers, 1997.
- [19] L.A. Zadeh, "Fuzzy sets", *Information and Control*, 8, 1965, pp. 338-353.
- [20] L.A. Zadeh, "The Concept of a Linguistic Variable and its application to Approximate Reasoning. Part 1", *Information Sciences*, 8, 1975, pp. 199-249; "Part 2", *Information Sciences*, 8, 1975, pp. 301-357; "Part 3", *Information Sciences*, 9, 1975, pp. 43-80.

NUEVA APROXIMACIÓN AL MÉTODO TÓPSIS DIFUSO CON ETIQUETAS LINGÜÍSTICAS

M. Socorro Garcia-Cascales¹, M. Teresa Lamata²

¹ Dpto de Electrónica, Tecnología de Computadoras y Proyectos.
Universidad de Politécnica de Cartagena. Murcia, España, socorro.garcia@upct.es

² Dpto. Ciencias de la Computación e Inteligencia Artificial.
Universidad de Granada. 18071- Granada. España, mtl@decsai.ugr.es

Resumen

Es bien conocido que la mayoría de los modelos de toma de decisión no establecen su salida de resultados basados en la entrada de los mismos. En la mayor parte de los casos la salida de datos de los modelos es de tipo numérico. Un ejemplo podemos verlo en el método TOPSIS que evalúa las alternativas mediante un ranking. El presente trabajo se basa en el método TOPSIS, pero ha sido adaptado de manera que la salida de datos sea lingüística así como lo es la entrada de los mismos. Finalmente, se ilustra el modelo propuesto mediante un ejemplo.

Palabras Clave: TOPSIS, Etiquetas lingüísticas

Pero existen también algunas desventajas como son:

- La existencia de orden inverso
 - Y que se pueden utilizar etiquetas lingüísticas como entrada de datos, mientras que la salida es un número asociado con el índice de proximidad, por tanto numérico. Con respecto a esto, intentamos hacer una modificación en el algoritmo del modelo TOPSIS difuso de manera que si la entrada de datos es lingüística entonces la salida de los mismos también lo sea.
- El artículo está organizado de la siguiente manera: en la sección 2, se describen las variables lingüísticas y los conjuntos difusos. La sección 3 presenta el método TOPSIS así como el método algorítmico propuesto. En la sección 4 se presenta un ejemplo de aplicación y finalmente se resumen las principales conclusiones.

1 INTRODUCCION

La Toma de Decisión Multicriterio es un proceso que consiste en encontrar la mejor alternativas de entre un conjunto de alternativas factibles.

Entre los diversos métodos compensatorios de la Toma de Decisión Multicriterio, es posible considerar un subgrupo que incluye aspectos de costes y beneficios. Uno de ellos es el método TOPSIS (Technique for Order Performance by Similarity to Idea Solution) Técnica para ordenar las preferencias mediante similitud a la solución ideal. Esta aproximación se utiliza por cuatro razones principales [11]:

- a. La lógica aplicada en TOPSIS es racional y comprensible;
- b. Los procesos de computación son sencillos
- c. El concepto permite la búsqueda de las mejores alternativas para cada criterio mostrándose en una forma matemática simple.
- d. La importancia de los pesos se incorporan en los procesos de comparación

2 VARIABLES LINGÜÍSTICAS Y CONJUNTOS DIFUSOS

2.1. VARIABLE LINGÜÍSTICA

La mayor parte de las veces, el decisor no es capaz de definir la importancia de los criterios y/o la bondad de las alternativas con respecto a cada criterio en términos numéricos. En la mayoría de las situaciones, estas valoraciones son lingüísticas, dando lugar a medidas o cantidades que no son exactas sino aproximadas.

Desde que Zadeh [12] introdujera el concepto de conjunto difuso y el de variable lingüística la popularidad y el uso de la lógica difusa ha sido extraordinario. Estamos particularmente interesados en el papel de las variables lingüísticas como una escala ordinal así como en sus términos asociados, en este caso de decisión multicriterio. Por *variable lingüística* [13,14,15] se entiende una variable cuyos valores son palabras o sentencias en un lenguaje natural o artificial. Por ejemplo la *edad* es una variable lingüística si sus valores son lingüísticos en vez de numéricos, por ejemplo, joven, no joven, muy joven, bastante joven, viejo, no muy viejo, no muy joven, etc en lugar de números como 20,21,22,23,...etc. En general, las

variables lingüísticas forman un conjunto ordenado $\mathfrak{A} = \{L_i / i = 1, 2, \dots, n\}$ siendo $L_i < L_j \Leftrightarrow i < j$.

Definición 1: Una variable lingüística se caracteriza por un quintuplo

$$\{X; T(X); U; G; M\} \quad (1)$$

Donde:

1. X es el nombre de la variable
2. $T(X)$ es el término del conjunto de X , esto es, la colección de sus valores lingüísticos.
3. U es un universo de discusión de la variable
4. G es la regla sintáctica para la generación de los elementos de $T(X)$ y
5. M es la regla semántica que asocia a cada elemento de $T(X)$ su significado.

En general para el decisor es más fácil cuando el/ella evalúa sus juicios mediante términos lingüísticos que cuando lo hace con números. Siendo, en estos casos, el concepto de número difuso más adecuado que el de número real para su representación

2.2. CONJUNTOS DIFUSOS

La teoría de conjuntos difusos, introducida por Zadeh [12] para tratar problemas vagos inciertos e imprecisos se ha utilizado como una herramienta de modelado para sistemas complejos que pueden ser controlados por los humanos pero que son difíciles de definir con precisión. El fundamento teórico de los números triangulares difusos puede verse en Klir [9]

Definición 2: Si T_1 y T_2 son dos números triangulares difusos definidos por las tripletas (a_1, b_1, c_1) y (a_2, b_2, c_2) respectivamente. Entonces, para este caso, las operaciones aritméticas necesarias con números difusos positivos son:

a) *Adición*

$$T_1 \oplus T_2 = [a_1 + a_2, b_1 + b_2, c_1 + c_2] \quad (2)$$

b) *Sustracción*

$$T_1 \ominus T_2 = T_1 + (-T_2) \text{ donde el opuesto } -T_2 = (-c_2, -b_2, -a_2)$$

$$\text{entonces } T_1 \ominus T_2 = [a_1 - c_2, b_1 - b_2, c_1 - a_2] \quad (3)$$

c) *Multipliación*

$$T_1 \otimes T_2 = [a_1 \times a_2, b_1 \times b_2, c_1 \times c_2] \quad (4)$$

d) *División*

$$T_1 \oslash T_2 = \left[\frac{a_1}{c_2}, \frac{b_1}{b_2}, \frac{c_1}{a_2} \right], \quad 0 \neq [a_2, b_2, c_2] \quad (5)$$

e) *Raíz cuadrada.*

$$T_1^{1/2} = [a_1^{1/2}, b_1^{1/2}, c_1^{1/2}] \quad (6)$$

3. TOPSIS

El método TOPSIS es un modelo de decisión propuesto para ordenar preferencias por similitud a una solución ideal, es por tanto un método de ranking. Fue desarrollado por Hwang y Yoon [6] en 1981 y mejorada por los propios autores en 1987 y 1992, también trabajaron Zeleny [17], Lai et al [10], García-Cascales y Lamata [5] y muchos más.

TOPSIS es un método de decisión multicriterio de ordenación para identificar las soluciones de un conjunto finito de alternativas. El principio básico es que la alternativa elegida debe tener la menor distancia a la solución ideal positiva y la mayor distancia a la solución ideal negativa. Una solución ideal se define como una colección de puntuaciones o valores en todos los atributos considerados en la decisión, pudiendo suceder que tal solución sea inalcanzable. El vector compuesto por los mejores valores del j-ésimo atributo respecto de todas las alternativas posibles es quien recibe el nombre de “solución ideal positiva” (SIP); reciprocamente, la “solución ideal negativa” (SIN) será aquella cuyo vector contenga los peores valores en todos los atributos.

El concepto intuitivo de alternativa ideal es que sería aquella que, sin dudarlo, siempre elegiría el decisor. De igual modo, la alternativa anti-ideal sería aquella que, sin dudarlo, nunca elegiría el decisor.

De este modo, puede suceder que una alternativa seleccionada desde el punto de vista de su “distancia” más corta respecto de la solución ideal positiva, Zeleny [17] deba competir con otra alternativa seleccionada como la más lejana de la solución ideal negativa. Por ello, y a fin de definir la solución ideal, el método TOPSIS define un índice de similaridad que se construye combinando la proximidad al ideal positivo y la lejanía respecto al ideal negativo.

El procedimiento de TOPSIS puede expresarse en una serie de pasos que pueden verse en Chen y Hwang [2], y que nosotros reproducimos modificado para el problema que resolvemos a continuación.

3.1. METODO ALGORÍTMICO PROPUESTO

Proponemos un algoritmo para determinar la opción preferible entre todas las posibles alternativas:

Paso 1: Identificación de los criterios de evaluación de forma que relacione las capacidades del sistema con los objetivos.

Paso 2: Generación de alternativas determinando el sistema finito de alternativas para lograr el objetivo.

Paso 3: Evaluación de las alternativas en función de los criterios.

Paso 4: Identificación del peso de los criterios. Es importante considerar que los criterios puedan tener distinta importancia.

Paso 5: Construcción de la matriz de decisión con números difusos.

Paso 5bis: Introducción de alternativas ficticias, donde al conjunto de alternativas le añadiremos las alternativas correspondientes a la valoración de todos los criterios por el valor correspondiente de cada una de las etiquetas utilizadas en el problema. Entenderemos estas variables como ficticias

Paso 6: La matriz de decisión normalizada asociada. Para ello los valores de cada criterio los dividimos por la norma con objeto de que la escala sea la misma para todos los criterios.

$$\bar{n}_{ij} = x_{ij} / \sqrt{\sum_{j=1}^n (x_{ij})^2}, j = 1, \dots, n, i = 1, \dots, m \quad (7)$$

utilizando para los números difusos las operaciones definidas (2), (4), (5) y (6), siendo el resultado $\bar{n}_{ij} = (n_{ij}^a, n_{ij}^b, n_{ij}^c)$ otro número difuso.

Paso 7: Construcción de la matriz ponderada normalizada asociada.

$$\bar{v}_{ij} = w_j \bar{n}_{ij}, j = 1, \dots, n, i = 1, \dots, m, \quad (8)$$

Donde $\bar{v}_{ij} = (v_{ij}^a, v_{ij}^b, v_{ij}^c)$ y operaremos utilizando la ecuación (4).

Paso 8: Determinación de la solución ideal positiva y la solución ideal negativa, respectivamente.

$$\bar{A}^+ = \{\bar{v}_1^+, \dots, \bar{v}_n^+\} = \left\{ \begin{array}{l} (\max \bar{v}_{ij}, j \in J), \\ (\min \bar{v}_{ij}, j \in J') \end{array} \right\} \quad (9)$$

$$\bar{A}^- = \{\bar{v}_1^-, \dots, \bar{v}_n^-\} = \left\{ \begin{array}{l} (\min \bar{v}_{ij}, j \in J), \\ (\max \bar{v}_{ij}, j \in J') \end{array} \right\} \quad (10)$$

donde J está asociado con aquellos criterios que indican ganancias o beneficios y J' está asociado con los criterios que indican pérdidas o costes.

Al haber introducido en la matriz las distintas variables lingüísticas, podemos observar que $A^+ = L_n$ y $A^- = L_1$.

Paso 9: Cálculo de la separación de cada alternativa con respecto a la solución ideal positiva y a la solución ideal negativa respectivamente:

$$\bar{d}_i^+ = \left\{ \sum_{j \in J} (\bar{v}_{ij} - \bar{v}_j^+)^2 \right\}^{\frac{1}{2}}, i = 1, \dots, m \quad (11)$$

$$\bar{d}_i^- = \left\{ \sum_{j \in J'} (\bar{v}_{ij} - \bar{v}_j^-)^2 \right\}^{\frac{1}{2}}, i = 1, \dots, m \quad (12)$$

en este caso usamos la distancia euclídea, pero podríamos haber tenido en cuenta cualquier otra.

Paso 10: Cálculo de la proximidad relativa de cada alternativa a la solución ideal positiva y negativa mediante el índice de proximidad.

$$\bar{R}_i = \frac{\bar{d}_i^-}{\bar{d}_i^+ + \bar{d}_i^-}, i = 1, \dots, m \quad (13)$$

Si $\rightarrow \bar{R}_i \cong 1 \rightarrow A_i = \bar{A}^+$

Si $\rightarrow \bar{R}_i \cong 0 \rightarrow A_i = \bar{A}^-$

Paso 11: Establecimiento del orden de preferencias de todas las alternativas de acuerdo al coeficiente de proximidad. Obviamente, una alternativa A_j está más próxima a la $\bar{A}^+$ y más lejana de $\bar{A}^-$ cuanto más se aproxima a 1 el índice $\bar{R}_i$

Paso 12: Defuzzificación

De entre los muchos métodos posibles existentes en la literatura nos quedamos con el propuesto en [3]. Para ello el cálculo lo haremos sobre la siguiente expresión

$$I_{\beta, \lambda}(A_i) = \beta S_M(A_i) + (1 - \beta) \lambda S_R(A_i) + (1 - \beta)(1 - \lambda) S_L(A_i)$$

Donde:

- $S_L(A_i)$ es el valor inferior medio asociado con la función inversa $g_A^L(x)$.
- $S_M(A_i)$ es el área del corazón del número difuso y
- $S_R(A_i)$ representa el valor superior medio del número difuso que está asociado con la función inversa $f_A^R(x)$.
- $\beta \in [0, 1]$ es el índice de modalidad representando la importancia del valor central y
- $\lambda \in [0, 1]$ es el grado de optimismo del decisor.

Estas áreas están representadas en la fig1.

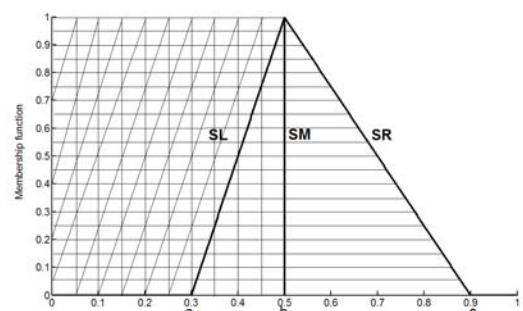


Figura 1. Representación de $S_L(A_i)$, $S_M(A_i)$ y $S_R(A_i)$

En este problema hemos considerado que las tres áreas tienen la misma importancia ó peso. Correspondiéndose con un decisor neutral donde $\alpha = 1/2$ y $\beta = 1/3$

Paso 13. Salida de valores lingüísticos.

Una vez obtenida la defuzzificación en el paso anterior, los valores obtenidos son reales y por lo tanto tenemos un orden definido entre ellos, así como la posibilidad de encontrar la mínima distancia de cada una de las alternativas a las alternativas ficticias introducidas en el paso 5bis.

4. CASO PRÁCTICO

En esta sección, trabajamos con un caso real de decisión para ilustrar el método de decisión TOPSIS mediante el algoritmo propuesto.

El objetivo global del problema es decidir cual es el mejor sistema de lavado de las piezas para un motor diesel y su

valoración final en los mismos términos que la entrada. El problema del que se parte García-Cascales y Lamata [5] es el siguiente; contamos con piezas que tienen diversos grados de suciedad, con geometría y peso muy diferente, y necesitamos un proceso de trabajo que exige rapidez y flexibilidad a la hora de limpiar piezas y donde se deben tener en cuenta criterios como el coste total de operación anual, la productividad del sistema empleado, la capacidad de carga del sistema, la eficiencia en la limpieza y la salubridad de los productos utilizados. Al abordar el problema nos encontramos con unos criterios cuyos datos se podían cuantificar y otros que no, dadas sus características, la evaluación solo podía obtenerse mediante valores lingüísticos.

4.1. ESTRUCTURACIÓN DEL PROBLEMA DE DECISIÓN

Establecer los criterios de decisión

- C1: Coste Total de operación anual.
- C2: Productividad volumétrica de la Pila
- C3: Capacidad de carga
- C4: Eficiencia en la limpieza
- C5: Salubridad

Determinar las alternativas

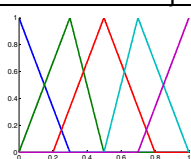
- A1: Lavadora convencional
- A2: Lavado químico
- A3: Lavado ultrasonidos

4.2. ANÁLISIS DEL PROBLEMA DE DECISIÓN

Consiste en evaluar las alternativas en términos de los criterios mediante las siguientes etiquetas que modelizaremos a través de números difusos triangulares y que recogemos en la tabla 1.

Tabla 1: Valores numéricos asociados a las etiquetas de bondad

	Pesos de los C_j	Pesos de los C_j
Muy malo (Mm)	[0, 0, 0.3]	
Malo (m)	[0, 0.3, 0.5]	
Regular (R)	[0.2, 0.5, 0.8]	
Bueno (B)	[0.5, 0.7, 1]	
Muy bueno (MB)	[0.7, 1, 1]	



En este caso, se opta por números difusos no simétricos como es el caso de las etiquetas bueno y malo pero que si son simétricas respecto de la etiqueta central (regular), por estar más acordes con aquellos valores que expresó el decisor sobre la valoración de las etiquetas.

Paso 4. Identificar el peso de los criterios.

Las valoraciones para cada criterio fueron las siguientes:

C1: Muy Importante [0.7, 1, 1],

C2: Importante [0.2, 0.5, 0.8],

C3: Poco importante [0, 0, 0.3],

C4: Importante [0.2, 0.5, 0.8],

C5: Moderadamente importante [0, 0.3, 0.5],

Datos que una vez normalizados están expresados en la tabla 2.

Tabla 2: Pesos normalizados de la importancia de los criterios

	Pesos de los C_j	Pesos de los C_j normalizados
MI	[0.5, 0.7, 1]	[0.20, 0.43, 0.91]
I	[0.2, 0.5, 0.8]	[0.06, 0.22, 0.73]
PI	[0, 0, 0.3]	[0, 0, 0.27]
I	[0.2, 0.5, 0.8]	[0.06, 0.22, 0.73]
MoI	[0, 0.3, 0.5]	[0, 0.13, 0.45]

Paso 5 y 5b: Construir la matriz de decisión correspondiente junto con las cinco etiquetas de valoración (tabla 3).

Paso 6: así como la matriz de decisión normalizada. (Tabla 4 y Tabla 5 ver apéndice)

Tabla 3: Información lingüística del problema de decisión

	MI	I	PI	I	MoI
	C_1	C_2	C_3	C_4	C_5
A_1	B	B	M	R	R
A_2	R	M	MB	B	Mm
A_3	M	MB	R	B	B
Mm	Mm	Mm	Mm	Mm	Mm
m	M	m	m	m	m
R	R	R	R	R	R
B	B	B	B	B	B
MB	MB	MB	MB	MB	MB

Paso 7: Construir la matriz ponderada normalizada. (Tabla 6, ver apéndice), para ello operamos mediante la expresión (4).

Paso 8: Determinar la solución ideal positiva y la solución ideal negativa, que se exponen en la (tabla 7 ver apéndice). Se puede comprobar que se corresponden con las etiquetas: Muy bueno y Muy malo, respectivamente.

Paso 9: Calcular la distancia de cada alternativa a la solución ideal positiva y a la solución ideal negativa que reflejamos en la tabla 8.

Tabla 8: Valores de las distancias a las alternativas ideal y anti-ideal

	0.0227	0.1146	0.1940	$d-A1$	0.0474	0.2159	0.8375
$d+A1$							
$d+A2$	0.0492	0.1798	0.4597	$d-A2$	0.0221	0.1625	0.6577
$d+A3$	0.0636	0.1899	0.4424	$d-A3$	0.0223	0.1773	0.7122
$d+Mm$	0.0684	0.3257	0.9321	$d-Mm$	0.0000	0.0000	0.0000
$d+m$	0.0684	0.2280	0.6658	$d-m$	0.0000	0.0977	0.2663
$d+R$	0.0488	0.1629	0.2663	$d-R$	0.0195	0.1629	0.6658
$d+B$	0.0195	0.0977	0.0000	$d-B$	0.0488	0.2280	0.9321
$d+MB$	0.0000	0.0000	0.0000	$d-MB$	0.0684	0.3257	0.9321

Paso 10: Calcular el coeficiente de proximidad. (Tabla 9).

Tabla 9: Coeficiente de proximidad

RA1	0.0460	0.6533	11.9366
RA2	0.0198	0.4747	9.2296
RA3	0.0193	0.4827	8.2866
RMm	0.0000	0.0000	0.0000
Rm	0.0000	0.3000	3.8947
RR	0.0210	0.5000	9.7367
RB	0.0524	0.7000	13.6314
RMB	0.0734	1.0000	13.6314

Pasos 11 y 12: Defuzzificación, ordenación de las alternativas y salida lingüística de resultados.

	Salida defuzzificada	Orden	Salida lingüística
IA1	2.4326	1	Regular-Bueno
IA2	1.8580	2	Malo-Regular
IA3	1.7061	3	Malo-Regular
IMm	0.0000		
Im	0.8491		
IR	1.9596		
IB	2.7473		
IMB	2.9508		

5. CONCLUSIONES

En la mayoría de los casos los sistemas de decisión dan las salidas de los datos en forma numérica. Parece más lógico que el decisor quiera que su resultado sea dado en los mismos términos que el input. Por ello, hemos incluido ciertas modificaciones al método TOPSIS de manera que la salida de los datos de esta metodología sea en los mismos términos que la entrada de datos, que en este caso se ha realizado mediante términos lingüísticos. En este artículo, hemos estudiado un problema de decisión multicriterio difuso como ejemplo de aplicación de las modificaciones propuestas al método TOPSIS difuso, para así apreciar la bondad de las aportaciones al mismo.

Agradecimientos

Este artículo se ha elaborado bajo los proyectos financiados por la DGICYT, TIN2008-06872-C04-04 y Junta de Andalucía (P07-TIC02970), respectivamente.

Referencias

[1] R.E. Bellman, L.A. Zadeh. Decision-making in a fuzzy environment. *Management Sci.*, 17, 141-164.,1970.

[2] Chen, S.J y Hwang, C.L. *Fuzzy Multiple Attribute Decision Making: Methods and Applications*, Springer-Verlang, Berlin, 1992.

[3] M.S, Garcia-Cascales,., M.T. Lamata,: A modification to the index of Liou and Wang for ranking fuzzy number, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 411-424, 2007.

[4] García-Cascales, M.S y Lamata, M.T. Multi-criteria análisis for a maintenance mangement problema in an Enghien factory: racional Choice. In press *Journal of Intelligent Manufacturing*. 2009.

[5] García-Cascales, M.S y Lamata, M.T. Selection of a cleaning system for a engine maintenance based on the analytic hierarchy process. *Computer and Industrial Engineering* 56, 1442-1451, 2009

[6] Hwang, C.L y Yoon, K. *Multiple Attibute Decision Methods and Applications*, Springer, Berlin Heidelberg, 1981

[7] J. Kacprzyk, R.R. Yager. Linguistic summaries of data using fuzzy logic. *International Journal of General Systems*. 30, 2, 133-154, 2001.

[8] E.E. Kerre. The use of fuzzy set theory in electrocardiological diagnostics, in: M.M. Gupta and E. Sanchez, Eds., *Approximate Reasoning in Decision Analysis* Amsterdam; North-Holland 277-282, 1982.

[9] G.J. Klir and B. Yuan, *Fuzzy sets and Fuzzy Logic*, Prentice Hall PTR, New Jersey e d. 1995

[10] Lai, Y.J, Liu, T.Y y Hwang, C.L. TOPSIS for MODM, *European Journal of Operational Research* 76 (3), 486-500, 1994.

[11] T.C. Wang, T.H. Chang. Application of TOPSIS in evaluating initial training aircraft under a fuzzy environment. *Expert Systems with Applications* 33, (4), 870 (2007).

[12] L. A. Zadeh . Fuzzy sets. *Information and Control*, 8:338-353, 1965.

[13] L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning: Part 1. *Information Sciences*, 8, 199-249, 1975.

[14] L. A. Zadeh, J. Kacprzyk (eds). *Computing with Words in Information / Intelligent Systems 1. Foundations. Studies in Fuzziness and Soft Computing*, Physica-Verlag (Springer-Verlag), Heidelberg and New York. Vol. 34 1999.

[15] L. A. Zadeh, J. Kacprzyk (eds). *Computing with Words in Information / Intelligent Systems 2. Applications. Studies in Fuzziness and Soft Computing*. Physica-Verlag (Springer-Verlag), Heidelberg and New York. Vol. 33 1999.

[16] Zeleny, M. Multiple Criteria Decision Making, McGraw-Hill, New York, 1982

Apéndice A

Tabla 4: Matriz de decisión

	C1	C2	C3	C4	C5
Pesos	0.2000 0.4300 0.9100	0.0600 0.2200 0.7300	0.0000 0.0000 0.2700	0.0600 0.2200 0.7300	0.0000 0.1300 0.4500
A1	0.5000 0.7000 1.0000	0.5000 0.7000 1.0000	0.0000 0.3000 0.5000	0.2000 0.5000 0.8000	0.2000 0.5000 0.8000
A2	0.2000 0.5000 0.8000	0.0000 0.3000 0.5000	0.7000 1.0000 1.0000	0.5000 0.7000 1.0000	0.0000 0.0000 0.3000
A3	0.0000 0.3000 0.5000	0.7000 1.0000 1.0000	0.2000 0.5000 0.8000	0.5000 0.7000 1.0000	0.5000 0.7000 1.0000
Mm	0.0000 0.0000 0.3000	0.0000 0.0000 0.3000	0.0000 0.0000 0.3000	0.0000 0.0000 0.3000	0.0000 0.0000 0.3000
m	0.0000 0.3000 0.5000	0.0000 0.3000 0.5000	0.0000 0.3000 0.5000	0.0000 0.3000 0.5000	0.0000 0.3000 0.5000
R	0.2000 0.5000 0.8000	0.2000 0.5000 0.8000	0.2000 0.5000 0.8000	0.2000 0.5000 0.8000	0.2000 0.5000 0.8000
B	0.5000 0.7000 1.0000	0.5000 0.7000 1.0000	0.5000 0.7000 1.0000	0.5000 0.7000 1.0000	0.5000 0.7000 1.0000
MB	0.7000 1.0000 1.0000	0.7000 1.0000 1.0000	0.7000 1.0000 1.0000	0.7000 1.0000 1.0000	0.7000 1.0000 1.0000

Tabla 5: Matriz normalizada

	C1	C2	C3	C4	C5
nA1	0.2266 0.4292 0.9667	0.2186 0.3791 0.8111	0.0000 0.1685 0.4369	0.0844 0.2858 0.6963	0.0922 0.3119 0.7734
nA2	0.0906 0.3066 0.7734	0.0000 0.1625 0.4056	0.3172 0.5617 0.8737	0.2109 0.4002 0.8704	0.0000 0.0000 0.2900
nA3	0.0000 0.1839 0.4834	0.3061 0.5415 0.8111	0.0906 0.2808 0.6990	0.2109 0.4002 0.8704	0.2304 0.4366 0.9667
nMm	0.0000 0.0000 0.2900	0.0000 0.0000 0.2433	0.0000 0.0000 0.2621	0.0000 0.0000 0.2611	0.0000 0.0000 0.2900
nm	0.0000 0.1839 0.4834	0.0000 0.1625 0.4056	0.0000 0.1685 0.4369	0.0000 0.1715 0.4352	0.0000 0.1871 0.4834
nR	0.0906 0.3066 0.7734	0.0875 0.2708 0.6489	0.0906 0.2808 0.6990	0.0844 0.2858 0.6963	0.0922 0.3119 0.7734
nB	0.2266 0.4292 0.9667	0.2186 0.3791 0.8111	0.2266 0.3932 0.8737	0.2109 0.4002 0.8704	0.2304 0.4366 0.9667
nMB	0.3172 0.6131 0.9667	0.3061 0.5415 0.8111	0.3172 0.5617 0.8737	0.2953 0.5717 0.8704	0.3225 0.6238 0.9667

Tabla 6: Matriz ponderada normalizada

	C1	C2	C3	C4	C5
vA1	0.0453 0.1846 0.8797	0.0131 0.0834 0.5921	0.0000 0.0000 0.1180	0.0051 0.0629 0.5083	0.0000 0.0405 0.3480
vA2	0.0181 0.1318 0.7038	0.0000 0.0357 0.2961	0.0000 0.0000 0.2359	0.0127 0.0880 0.6354	0.0000 0.0000 0.1305
vA3	0.0000 0.0791 0.4399	0.0184 0.1191 0.5921	0.0000 0.0000 0.1887	0.0127 0.0880 0.6354	0.0000 0.0568 0.4350
vMm	0.0000 0.0000 0.2639	0.0000 0.0000 0.1776	0.0000 0.0000 0.0708	0.0000 0.0000 0.1906	0.0000 0.0000 0.1305
vm	0.0000 0.0791 0.4399	0.0000 0.0357 0.2961	0.0000 0.0000 0.1180	0.0000 0.0377 0.3177	0.0000 0.0243 0.2175
vR	0.0181 0.1318 0.7038	0.0052 0.0596 0.4737	0.0000 0.0000 0.1887	0.0051 0.0629 0.5083	0.0000 0.0405 0.3480
vB	0.0453 0.1846 0.8797	0.0131 0.0834 0.5921	0.0000 0.0000 0.2359	0.0127 0.0880 0.6354	0.0000 0.0568 0.4350
vMB	0.0634 0.2636 0.8797	0.0184 0.1191 0.5921	0.0000 0.0000 0.2359	0.0177 0.1258 0.6354	0.0000 0.0811 0.4350

Tabla 7: Solución ideal positiva y negativa

	C1	C2	C3	C4	C5
A+	0.0634 0.2636 0.8797	0.0184 0.1191 0.5921	0.0000 0.0000 0.2359	0.0177 0.1258 0.6354	0.0000 0.0811 0.4350
A-	0.0000 0.0000 0.2639	0.0000 0.0000 0.1776	0.0000 0.0000 0.0708	0.0000 0.0000 0.1906	0.0000 0.0000 0.1305

OPTIMIZATION OF ROUTE DISTRIBUTION PROBLEMS WITH FUZZY OBJECTIVES

Julio Brito¹ F. Javier Martínez¹ José A. Moreno¹ José L. Verdegay²

¹ I.U.D.R. Universidad de La Laguna, E-38271 La Laguna, Spain, {jbrito, fmartinez, jamoreno}@ull.es

² Dep. CCIA Universidad de Granada, E-18071 Granada, Spain, verdegay@decsai.ugr.es

Abstract

Logistics planning is an area which offers many opportunities to study optimization problems such as route planning. These problems can be modeled according to the standard forms seen in vehicle route problems. One of the most common objectives in these kinds of problems is to minimize the time to cover the vehicle routes. In many real life situations the values for time can not be obtained in a precise way. In this article we propose a methodological approximation to obtain solutions to the vehicle route problem when the objective function is expressed in terms of fuzzy times.

Keywords: Vehicle route problems, fuzzy Optimization, Metaheuristics.

1 INTRODUCTION

Route distribution planning problems play a key role in supply chain management and especially in logistical systems. Proper planning can help improve the effectiveness and efficiency in these systems. Route planning problems in the present day context of the global economy have an important associated impact beside the obvious transport costs. They also affect the efficient use of resources, added value in service benefits and client satisfaction.

A traditional objective of these product distribution problems from a depot to a set of geographically disperse nodes is to determine the set of routes for the available vehicles that minimize total operating costs of the fleet, satisfying some, but not all, constraints. These problems have been studied under varied approaches and many different results have been ob-

tained through the formulation of vehicle route problems (VRP). The VRP is concerned with finding an optimal set of routes that begin and end in a depot for a specified fleet of vehicles that satisfy client demand (see Cordeau et al. [4]). Client, depot and vehicle characteristics, in addition to different route operating restrictions, offer different variations to the problem.

The two lines of research we wish to highlight provide solutions to the models that increasingly account for those characteristics observed in real cases and those that specifically seek more effective procedures to solve these problems.

Apart from the clearly practical aspects of this problem, the formal, academic study of the combinatorial optimization aspects of the problem are noteworthy. They are, for the most part, NP hard problems and it is not possible to find exact procedures and algorithms that solve large instances of the problem in polynomial time. It is there where the relevance of the utilized heuristic methods is acquired, specifically in general search strategies that carry out a limited or intensive study of the solution space and where reasonable quality solutions are produced with modest computation times.

In many typical vehicle route planning problems it is not always possible to have access to all of the necessary information, consequently leaving a problem characterized by incomplete or imprecise information of the problem parameters and variables. One of the current approaches in the modeling and resolution of problems containing uncertainty is found in an area known as Soft Computing and in the Theory of Fuzzy Sets (Zadeh, 1965). These approaches are used to construct computation systems that solve decision and optimization problems and where the modeling is difficult to define with precision, managing vagueness and the imprecision of available information, in addition to the formulation of preferences, constraints and objectives expressed by the decision makers[1].

In this paper we deal with the problem in which travel times required by a vehicle to go from one demand node to another is imprecise. Traffic and road conditions cause difficulties in the calculation of exact travel times. A great amount of uncertainty is present in the calculation of the travel times, for instance the day of the week or the time of day that a vehicle travels or other specific circumstances, such as traffic density and the speed at which the vehicle moves in roads and highways can be more or less distinct. On the other hand, in route planning, the travel time of the routes is usually one of the criteria to measure costs of the available fleet of vehicles and the efficient use of that fleet. Thus, in order to model the vehicle route problem, find solutions and evaluate the best ones, we use an objective function that is expressed in terms of duration times of the routes.

In this situation we consider the classic model of the VRP, known as the capacitated vehicle route problem (CVRP), where the coefficients of the objective function are imprecise, and the set of constraints are known exactly, even though the capacity of the vehicles is homogenous. To model this kind of imprecision in the coefficients we can use fuzzy quantities, namely triangular fuzzy numbers that can be expressed in terms of known values, using the knowledge at hand, derived from the expertise of the drivers of the vehicles or using the historic information provided by positioning tools in the vehicles.

The objective of this paper is to formulate the problem based on the factors mentioned earlier, and find a procedure that will allow optimal solutions to be found in an efficient way. After introducing the problem and its context in this section, the organization of the paper is as follows. In section 2 we formulate the CVRP by taking into account the conditions of uncertainty in the travel times. In Section 3 we explain how we apply some models proposed in the literature to obtain a solution to the combinatorial optimization problem where there are fuzzy coefficients in the objective function. In section 4 we solve this model with an example, using different heuristics, which generate satisfactory solutions.

2 CAPACITATED VEHICLE ROUTING PROBLEMS

The standard VRP (usually called capacitated VRP; CVRP) calls for the determination of a set of m routes whose total travel length is minimized such that: (a) each customer is visited exactly once by one route, (b) each route starts and ends at a single depot, and (c) the total demand of the customers served by a route does not exceed a given vehicle capacity C_k where q_j

is the demand of a node j . Travel times and travel costs are considered equivalent. If each vehicle i is assigned to a route R_i , a feasible solution for VRP is made up of a partition from nodes set V into m routes $R_1, R_2, \dots, R_m$ and the corresponding permutations of R_i that specify client order along the routes. A formulation of the VRP as Integer Linear Programming problem can be described [6], and assumed that the depot is the node 0 and $n+1$, n is the number of customers to be served by m vehicles and the decision variables are $x_{ij}^k \in \{0, 1\}$, $i = 0, 1, \dots, n$, $j = 1, 2, \dots, n+1$, $k = 1, 2, \dots, m$, where $x_{ij}^k = 1$ if vehicle k travels from customer i to j and 0 otherwise. We also consider the continuous variables r_i^k , representing the load of vehicle k after visiting node i . If the vehicle k goes from customer i to customer j ($x_{ij}^k = 1$) then $r_j^k = r_i^k - q_i$.

In particular, objective function may be formulated as follows:

$$\min \sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} c_{ij}^k x_{ij}^k \quad (1)$$

where c_{ij}^k is the *cost of travelling* from customer i to customer j by vehicle k . The total travel cost is an objective function to be minimized and is expressed in term of total time, then

$$\min \left(\sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} t_{ij}^k x_{ij}^k + \sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} u_i^k x_{ij}^k \right) \quad (2)$$

where t_{ij}^k denotes the time needed to go from customer i to customer j and u_i^k is the required time by vehicle k to unload demand to customer i . The objective function is to minimize the total travel time.

The CVRP has the follow constraints:

- Constraints ensuring that each customer is served exactly once.

$$\sum_{k=1}^m \sum_{i=0}^n x_{ij}^k = 1, \quad j \in [1..n]$$

$$\sum_{k=1}^m \sum_{i=1}^{n+1} x_{ji}^k = 1, \quad j \in [1..n]$$

- Constraints ensuring that each vehicle is used no more than once.

$$\sum_{j=1}^n x_{0j}^k = 1, \quad k \in [1..m]$$

$$\sum_{i=1}^n x_{in+1}^k = 1, \quad k \in [1..m]$$

- Constraints ensuring route continuity.

$$\sum_{i=0}^n x_{ij}^k - \sum_{i=1}^{n+1} x_{ji}^k = 0, \quad j \in [1..n], k \in [1..m]$$

- Constraints ensuring that the load of each vehicle is not greater than its capacity.

$$r_i^k + q_j - r_j^k \leq C_k(1 - x_{ij}^k) \\ i \in [1..n], j \in [1..n], k \in [1..m]$$

$$q_i \leq r_i^k \leq C_k, \quad i \in [0..n], k \in [1..m]$$

These constraints are an alternative set of constraints equivalent to the standard constraints of capacity cuts and ensure, in addition to the restriction of capacities, the connectivity of the routes and the elimination of subtours.

- In addition the conditions for the variables:

$$x_{ij}^k \in \{0, 1\}, r_i^k \geq 0, \\ i \in [0..n], j \in [1..n], k \in [1..m]$$

This formalization assumes that the decision-maker has access to specific information on the components that define the problem; that is, on objective functions and constraints. However in a logistics context the information of time travel is actually imprecise or incomplete.

Thus, if we suppose that all parameters of the problems are crisp: demand q_i , capacity C_k , unload time u_i^k , and available load r_i^k , travel times t_{ij}^k that can be fuzzy, the traditional model becomes a Fuzzy optimization problem. Intuitively, when any of these quantities are fuzzy numbers, the objective functions become fuzzy as well. If these parameters are approximately known, they can be represented by the fuzzy numbers $\tilde{t}_{ij}^k$, with their corresponding membership functions. Then, for instance, the objective function (1) and (2) can be expressed as:

$$\min \sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} \tilde{c}_{ij}^k x_{ij}^k$$

and

$$\min \left(\sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} \tilde{t}_{ij}^k x_{ij}^k + \sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} u_i^k x_{ij}^k \right) \quad (3)$$

The summation symbol Σ in the objective functions and constraints refers to an addition of fuzzy numbers. Hence there is a need to seek appropriate procedures for its solution.

3 FUZZY OPTIMIZATION PROBLEMS

An *optimization problem* can be described as the search for the value of specific *decision variables* so that identified *objective functions* attain their optimum values. The value of the variables is subject to stated *constraints*. In these problems the objective functions are defined on a set of solutions that we will denote by X . The objective function is not subject to any condition or property nor is the definition of the set X . Typically the number of elements of X is very high, essentially eliminating the possibility of a complete evaluation of all its solutions while determining the optimal solution.

Optimization problems in their most general form involve finding an optimal solution according to stated criteria. In practice, however, many situations lack the exact information that is needed in the problem, including its constraints, or in other cases, where it is unreasonable to access such specific constraints or clearly defined objective functions. In these situations it is advantageous to model and solve the problem using soft computing and fuzzy techniques.

Among all the optimization problems, the models that have received the most attention and have offered the most useful applications in different areas are Linear Programming (LP) models, which is the single objective linear case with linear constraints. The classic problem of LP is to find the maximum or minimum values of a linear function subject to constraints that are represented by linear equations or inequalities. The most general formulation of the LP problem is:

$$\begin{array}{ll} \min & Z = cx \\ \text{subject to} & Ax \leq b \\ & x \geq 0 \end{array}$$

The vector $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ represents the decision variables. The objective function is denoted by z , the numbers c_j are coefficients and the vector $c = (c_1, c_2, \dots, c_n) \in \mathbb{R}^n$ is known as the cost vector. The matrix $A = [a_{ij}] \in \mathbb{R}^{n \times m}$ is called the constraint or technological matrix and the vector $b = (b_1, b_2, \dots, b_m) \in \mathbb{R}^m$ represents the independent terms or right-hand-side of the constraints.

In many real situations not all the constraints and objective functions can be valued in a precise way. In these situations we are dealing with the general problem form of Fuzzy Linear Programming (FLP). FLP is characterized as follows: a_{ij} , b_j and c_i can be expressed as *fuzzy numbers*, x_i as variables whose states are fuzzy numbers, addition and multiplication operates with fuzzy numbers, and the inequalities are

among fuzzy numbers.

Different FLP models can be considered according to the elements that contain imprecise information that is taken as a basis for the classification proposed in [2], [5]. In our case use a *model with fuzzy costs as objective function coefficients*.

These models are those whose costs are not fully known (with imprecision). Therefore, they are represented by an m -dimensional fuzzy vector $c^f = (c_1^f, c_2^f, \dots, c_n^f)$, and the following model:

$$\begin{aligned} \min \quad & Z = c^f x \\ \text{subject to} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

Obviously, z is also a fuzzy number, but x can be a vector of fuzzy or non-fuzzy numbers, and each fuzzy cost is described by its corresponding membership function $\mu_i(x)$.

This model can be transformed into a simpler auxiliary model following the approach proposed at [3], [7]. This method proposed the use of an ordering function g that allows the comparison between fuzzy numbers, which facilitates minimizing the objective function. If $\tilde{u}$ and $\tilde{v}$ are two fuzzy numbers, $\tilde{u} \leq \tilde{v}$ if $g(\tilde{u}) \leq g(\tilde{v})$. In our problem we use the Yager third index as an ordering function for triangular fuzzy numbers $\tilde{u} = Tr(u_1, u_2, u_3)$ is $g(\tilde{u}) = u_1 + 2u_2 + u_3$. Therefore the objective function (3) can be replaced by:

$$\min \left(\sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} g(\tilde{t}_{ij}^k) x_{ij}^k + \sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} g(u_i^k) x_{ij}^k \right)$$

and using triangular fuzzy numbers and the Yager third index, we obtain the following objective function:

$$\begin{aligned} \min \quad & \left(\sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} (t_{ij}^{1k} x_{ij}^k + 2t_{ij}^{2k} x_{ij}^k + t_{ij}^{3k} x_{ij}^k) \right. \\ & \left. + \sum_{k=1}^m \sum_{i=0}^n \sum_{j=1}^{n+1} 4u_i^k x_{ij}^k \right) \end{aligned}$$

where $\tilde{t}_{ij}^k$ is a triangular fuzzy number $Tr(t_{ij}^{1k}, t_{ij}^{2k}, t_{ij}^{3k})$ and u_i^k is a crisp number equivalent to a triangular fuzzy number $Tr(u_i^k, u_i^k, u_i^k)$.

We use this fuzzy optimization approximation to solve the problem formulation described earlier. In the next section we provide specific examples.

Table 4: Results obtained

Heuristic	Total Time	route number
GRASP	1840	3
GRASP-LS	1840	3
VNS	1820	3
GRASP-VNS	1740	3

3.1 SOLUTIONS AND EXPERIMENTATION

An evaluation of the formulation and the proposed method is carried out using an example from Zheng and Liu [8]. The proposed instances with triangular travel times in their paper include 18 customers, labelled "1", "2", ..., "18", a depot, labelled '0' and vehicles with homogeneous capacity.

The amount of demand at each customer and the distances between customers (and the depot) are given in Table 1. The travel times are triangular fuzzy numbers shown in Tables 2 and 3, with the time windows of each customer. The unloading time at each location is 15 minutes and the capacity of each of the four available vehicles is 1000.

We have found solutions for the corresponding optimization problem using four heuristics: Greedy Randomize Adaptive Search Procedure (GRASP), GRASP with Local Search, a Variable Neighborhood Search (VNS) and a GRASP-VNS Hybrid procedure. The results were performed on a PC Intel CORE 2 Duo (2.26GHz) processors and 4GB RAM.

Table 4 presents the values of the best solutions obtained in run time 60s for the objective function.

The best solutions was obtained using VNS and GRASP-VNS hybrid heuristic in near 10min and consist in the following three routes, respectively:

$$\begin{aligned} R1: & 0 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 6 \rightarrow 4 \rightarrow 7 \rightarrow 10 \rightarrow 0 \\ R2: & 0 \rightarrow 8 \rightarrow 11 \rightarrow 5 \rightarrow 9 \rightarrow 14 \rightarrow 12 \rightarrow 16 \rightarrow 13 \rightarrow 0 \\ R3: & 0 \rightarrow 15 \rightarrow 18 \rightarrow 17 \rightarrow 0 \end{aligned}$$

The total time is $T = 1700$ and the loads of the vehicles are 1000, 985 and 500.

$$\begin{aligned} R1: & 0 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 18 \rightarrow 14 \rightarrow 15 \rightarrow 17 \rightarrow 0 \\ R2: & 0 \rightarrow 10 \rightarrow 11 \rightarrow 7 \rightarrow 9 \rightarrow 6 \rightarrow 5 \rightarrow 13 \rightarrow 16 \rightarrow 0 \\ R3: & 0 \rightarrow 4 \rightarrow 8 \rightarrow 12 \rightarrow 0 \end{aligned}$$

The total time is $T = 1700$ and the loads of the vehicles are 1000, 990 and 495.

Table 1: Demands (q) and distances (d) between customers.

d	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	q
1	19.0																		200
2	17.5	6.0																	100
3	28.0	11.0	10.5																140
4	24.0	21.0	15.0	20.0															160
5	24.5	32.0	26.0	34.0	15.5														200
6	31.2	44.5	39.5	49.0	31.0	16.0													60
7	31.0	48.5	45.0	55.5	41.5	28.5	16.0												200
8	21.0	37.5	33.5	44.0	30.0	18.5	13.0	11.5											135
9	18.0	36.0	33.0	44.0	38.0	24.0	20.0	13.5	7.0										120
10	21.5	40.0	39.0	49.5	43.0	36.5	32.5	21.0	20.0	13.0									140
11	36.5	55.0	54.0	65.0	56.0	46.0	37.0	21.0	28.0	23.0	15.5								100
12	31.5	46.5	48.0	57.0	55.5	51.0	48.5	36.0	36.0	29.0	16.0	21.5							200
13	23.0	38.5	39.0	48.5	47.0	44.0	43.0	32.5	30.0	23.0	12.0	23.0	9.0						80
14	28.0	38.5	41.5	49.0	52.0	51.0	52.5	43.5	40.0	33.0	22.5	33.0	13.0	11.0					60
15	34.5	40.0	44.0	50.0	56.5	58.5	62.0	54.0	50.0	43.0	34.0	44.5	24.0	22.0	11.5				200
16	30.0	29.5	34.5	38.0	48.5	54.0	60.5	56.0	49.0	43.5	38.0	51.5	33.0	28.0	20.5	14.0			90
17	18.5	16.5	21.3	26.5	35.0	41.0	50.0	48.0	39.0	35.0	33.0	48.0	34.0	27.0	24.0	23.5	13.5		200
18	24.0	14.0	20.0	22.0	35.0	44.0	54.5	55.0	45.0	41.5	41.0	56.5	43.0	35.0	32.0	36.0	17.0	8.5	100

Table 2: fuzzy travel times between customers (part I).

T	0	1	2	3	4	5	6	7	8
1	(25,50,75)								
2	(5,10,15)	(20,40,60)							
3	(25,50,75)	(5,10,15)	(20,40,60)						
4	(7,15,23)	(25,50,75)	(7,15,23)	(22,45,68)					
5	(25,50,75)	(17,35,53)	(17,35,53)	(15,30,45)	(17,35,53)				
6	(25,50,75)	(7,15,23)	(20,40,60)	(2,5,8)	(22,45,68)	(15,30,45)			
7	(12,25,38)	(20,40,60)	(15,30,45)	(17,35,53)	(7,15,23)	(12,25,38)	(17,35,53)		
8	(7,15,23)	(20,40,60)	(5,10,15)	(22,45,68)	(10,20,30)	(17,35,53)	(20,40,60)	(17,35,53)	
9	(25,50,75)	(7,15,23)	(22,45,68)	(5,10,15)	(22,45,68)	(15,30,45)	(5,10,15)	(20,40,60)	(20,40,60)
10	(10,20,30)	(22,45,68)	(12,25,38)	(22,45,68)	(7,15,23)	(15,30,45)	(20,40,60)	(5,10,15)	(12,25,38)
11	(25,50,75)	(5,10,15)	(17,35,53)	(15,30,45)	(17,35,53)	(5,10,15)	(15,30,45)	(5,10,15)	(17,35,53)
12	(27,55,83)	(17,35,53)	(17,35,53)	(15,30,45)	(17,35,53)	(2,5,8)	(15,30,45)	(7,15,23)	(17,35,53)
13	(5,10,15)	(20,40,60)	(5,10,15)	(20,40,60)	(7,15,23)	(15,30,45)	(17,35,53)	(17,35,53)	(5,10,15)
14	(25,50,75)	(5,10,15)	(20,40,60)	(2,5,8)	(22,45,68)	(15,30,45)	(2,5,8)	(17,35,53)	(17,35,53)
15	(22,45,68)	(5,10,15)	(20,40,60)	(5,10,15)	(22,45,68)	(15,30,45)	(5,10,15)	(17,35,53)	(17,35,53)
16	(7,15,23)	(22,45,68)	(7,15,23)	(22,45,68)	(10,20,30)	(15,30,45)	(22,45,68)	(17,35,53)	(10,20,30)
17	(15,30,45)	(20,40,60)	(12,25,38)	(20,40,60)	(10,20,30)	(12,25,38)	(17,35,53)	(2,5,8)	(12,25,38)
18	(25,50,75)	(5,10,15)	(22,45,68)	(5,10,15)	(25,50,75)	(15,30,45)	(7,15,23)	(17,35,53)	(20,40,60)

4 CONCLUSIONS

TIC-02970-JA

Uncertainty needs to be managed in real logistic and transportation systems. Fuzzy Logic systems have been used to model the uncertainty. We consider the CVRP with uncertainty in the travel times as triangular fuzzy numbers. We propose a simple approximation to obtain optimal solutions for these problems when fuzzy times are objective function coefficients. We offer an approximation and an example in which several heuristics are used to obtain solutions in short periods of time. The best solution is obtained with the GRASP-VNS hybrid procedure. These results indicate new avenues of possible future research.

Acknowledgements

The first three authors have been supported by project TIN2008-06873-C04-01 (70% FEDER) from the Spanish Government and PI2007/019 from the Canary Islands Government. J.L. Verdegay has been supported by projects TIN2008-06872-C04-04/TIN and

References

- [1] R. Bellman, L. Zadeh. *Decision-making in fuzzy environment*. Management Science 17:141-64. 1970.
- [2] J.M. Cadenas, J.L. Verdegay. *Optimization Models with Imprecise Data*. Ed. Universidad de Murcia, 1999.(in Spanish).
- [3] L. Campos, J.L. Verdegay. Linear Programming Problems and Ranking of Fuzzy Numbers. *Fuzzy Sets and Systems*, 32, 1-11, 1989.
- [4] J.F. Cordeau, G. Laporte, M.W.P. Savelsbergh, D. Vigo. *Vehicle Routing. Handbook in OR & MS*, Vol. 14. Chapter 6. C. Barnhart and G. Laporte (Eds.), Elsevier, 2007.

Table 3: fuzzy travel times between customers (part II).

T	9	10	11	12	13	14	15	16	17
10	(22,45,68)								
11	(17,35,53)	(15,30,45)							
12	(17,35,53)	(12,25,38)	(7,15,23)						
13	(20,40,60)	(20,40,60)	(17,35,53)	(17,35,53)					
14	(5,10,15)	(20,40,60)	(15,30,45)	(15,30,45)	(20,40,60)				
15	(2,5,8)	(20,40,60)	(15,30,45)	(20,40,60)	(20,40,60)	(2,5,8)			
16	(22,45,68)	(15,30,45)	(17,35,53)	(7,15,23)	(7,15,23)	(22,45,68)	(22,45,68)		
17	(20,40,60)	(5,10,15)	(12,25,38)	(12,25,38)	(12,25,38)	(17,35,53)	(17,35,53)	(12,25,38)	
18	(7,15,23)	(20,40,60)	(15,30,45)	(20,40,60)	(20,40,60)	(7,15,23)	(7,15,23)	(20,40,60)	(20,40,60)

- [5] M. Delgado, J.L. Verdegay and M.A. Vila. A general model for fuzzy linear programming. *Control and Cybernet*, 16 (2), 113-121, 1989.
- [6] P. Toth, D.Vigo. The vehicle routing problem. *Monographs on Discrete Mathematics and Applications*, Vol. 9, SIAM, 2002.
- [7] J.L. Verdegay. Fuzzy optimization: models, methods and perspectives. *6thIFSA-95 World Congress*. 39-71, 1995.
- [8] Y. Zheng, B. Liu. Fuzzy Vehicle Routing Model with Credibility Measure and its Hybrid Intelligent Algorithm. *Applied Mathematics and Computation*, 176(2), 673-683, 2006.

ALGORITMOS PARA ABORDAR MODELOS DE INVERSIÓN AFECTADOS DE INCERTIDUMBRE

Clara Calvo¹ José M. Cadenas² Juan V. Carrillo²
M. Carmen Garrido² Carlos Ivorra¹ Vicente Liern¹

¹Dept. Matemática para la Economía y la Empresa, Universitat de València.

²Dep. Ingeniería de la Información y las Comunicaciones. Universidad de Murcia.

e-mail: vicente.liern@uv.es

Resumen

En los problemas de inversión, el decisor debe seleccionar la mejor opción futura basándose en datos históricos y bajo unas preferencias que no son precisas, dando lugar a dos tipos de incertidumbre. En este trabajo, haciendo uso de la lógica fuzzy, analizamos el problema de selección de carteras eficientes en los que la incertidumbre aparece en el propio modelo. Presentamos un algoritmo exacto y un procedimiento heurístico para resolverlo e ilustramos su uso con algunos ejemplos simples que aparecen en los trabajos de Markowitz y utilizando datos del índice bursátil IBEX35.

Palabras Clave: Problema de la cartera, lógica fuzzy, programación fuzzy.

1 INTRODUCCIÓN

Entre todas las alternativas para invertir un determinado capital, quizá una de las más estudiadas, y en la que más claramente se presentan condiciones de incertidumbre es el problema de cartera valores. Se dispone de un capital C y se trata de elegir un cierto número de activos entre n disponibles, que presentan un riesgo de inversión. Para escoger de forma eficiente se puede utilizar un modelo de programación cuadrática en el que el objetivo es conseguir una cartera óptima, es decir la más beneficiosa para el inversor, de manera que tenga en cuenta que se debe invertir todo el capital y que el inversor no puede endeudarse. Éste modelo fue introducido por Markowitz en 1952, [10], de ahí que se le considere como el padre de la teoría moderna de carteras (véanse [4,12]).

Los modelos cartera, como ocurre con la mayoría de modelos que aborda la Economía Financiera o modelos de

toma de decisiones en general, presentan una incertidumbre inherente, porque se deben tomar decisiones acerca de sucesos futuros con la única información de los datos históricos.

La forma habitual de operar en los modelos de inversión es hacer uso de métodos estadísticos para determinar cuál sería la rentabilidad y el riesgo esperados de cada uno de los activos [4,10,12]. Tras el trabajo inicial de Markowitz, se han propuesto diversos métodos alternativos (véase [7,8,9,13,15,16]), pero la mayoría de éstos se basan en distribuciones de probabilidad para caracterizar el riesgo y la rentabilidad ([6, 11, 12]). Por ejemplo, Markowitz estima el riesgo esperado mediante la matriz de varianzas-covarianzas de los activos y la rentabilidad de cada activo mediante la media de sus rentabilidades históricas. Sin embargo, esto no es así en todos los modelos que coexisten dentro de la teoría de carteras.

Sin embargo, está claro que el problema de selección de carteras depende de algunos parámetros para los que no estaría justificada la asignación de distribuciones de probabilidad, en concreto, los que expresan las preferencias del inversor. Sin duda, para este tipo de incertidumbre el uso de técnicas basadas en soft computing resulta muy apropiado.

En este trabajo proponemos un tratamiento fuzzy para el problema de la cartera, comprobamos su funcionamiento con un algoritmo exacto basado en la resolución explícita de las condiciones Kuhn y Tucker y proponemos una heurística capaz de abordar instancias grandes del problema.

2. ALGUNOS MODELOS PARA EL PROBLEMA DE SELECCIÓN DE CARTERAS

Para que se produzca el equilibrio entre riesgo y rentabilidad que resulte más beneficioso para el inversor, se parte de algunas hipótesis que son las habituales dentro de los modelos de inversión [10, 12]:

Hipótesis:

- 1) El inversor tiene un comportamiento racional, es decir que buscará la opción más beneficiosa.
- 2) Las rentabilidades y el riesgo esperados pueden ser estimados mediante el vector de medias y la matriz de varianzas-covarianzas (respectivamente) de los datos históricos.
- 3) Se va a invertir todo el capital disponible.
- 4) No es posible endeudarse para incrementar la inversión.

Con estas hipótesis, Markowitz ([10, 11]) introdujo dos modelos clásicos que son económicamente equivalentes:

$$\text{Min} \left\{ R = x^T Q x : \sum_{i=1}^n r_i x_i \geq r_0, \sum_{i=1}^n x_i = 1, x_i \geq 0, 1 \leq i \leq n \right\} \quad (1)$$

$$\text{Max} \left\{ \sum_{i=1}^n r_i x_i : x^T Q x \leq R_0, \sum_{i=1}^n x_i = 1, x_i \geq 0, 1 \leq i \leq n \right\} \quad (2)$$

En estos modelos, n es el número de activos en los que se puede hacer la inversión, x_i es la proporción que se invierte en el i -ésimo activo de la cartera, r_i es la media (para el periodo histórico analizado) de las rentabilidades del i -ésimo activo y Q expresa la matriz de varianzas-covarianzas. En el modelo (1) el inversor fija una rentabilidad mínima que está dispuesto a obtener, r_0 , y selecciona la cartera que minimiza el riesgo. En el modelo (2) la manera de plantearlo es la contraria, fijado un riesgo máximo que se quiere asumir, R_0 , se trata de encontrar la cartera que maximiza la rentabilidad.

A partir de estos modelos básicos, si queremos incluir otras condiciones que tengan en cuenta las preferencias del inversor o las condiciones del mercado, será necesario introducir nuevas restricciones en (1) y (2). Pero también podríamos modificar algunas de las hipótesis descritas anteriormente. Por ejemplo, si se admite el endeudamiento, la restricción $x_i \geq 0$ debería ser eliminada.

También es razonable que el inversor decida no invertir más de cierta proporción de su capital en determinados activos. Esto supondría añadir al modelo las siguientes restricciones:

$$x_i \leq u_i, \quad 1 \leq i \leq n. \quad (4)$$

Si el inversor quiere invertir, *a priori*, un parte de su capital en determinados activos, es necesario incorporar restricciones

$$l_i \leq x_i, \quad 1 \leq i \leq n. \quad (5)$$

Una justificación para añadir las restricciones (4), (5) puede encontrarse, por ejemplo, en [1], [2], [3].

Pero esta no es la única razón por la que deberíamos incorporar restricciones que fijen cotas inferiores y superiores a la inversión en los activos. El exceso de diversificación puede constituir un problema para la inversor, que no sólo debería invertir pequeñas cantidades en muchos de los activos sino que debería tenerse en cuenta los posteriores costes de transacción que no aparecen en el modelo. El inversor, puede decidir que si no invierte una proporción mínima en cierto activo, entonces prefiere no invertir. Esto se expresa mediante variables semi-continuas que en el modelo se escribirían añadiendo variables binarias. El nuevo modelo de cartera, que desde el punto de vista de la programación matemática es muy diferente de (1), es el siguiente:

$$\begin{aligned} \text{Min} \quad & R = x^T Q x \\ \text{s.t.} \quad & \sum_{i=1}^n r_i x_i \geq r_0 \\ & \sum_{i=1}^n x_i = 1 \\ & l_i y_i \leq x_i \leq u_i, \quad 1 \leq i \leq n \\ & y_i \in \{0, 1\} \end{aligned} \quad (6)$$

donde la variable binaria y_i toma el valor 1 si el i -ésimo activo aparece en la cartera y 0 si no lo hace.

Por supuesto, en lugar de presentar el modelo (6) basándonos en (1) podríamos haber hecho lo mismo partiendo del modelo (2).

Si nos centramos en el modelo (6), una cartera eficiente sería la solución óptima de (6) para unos valores dados r_0 , l_i , u_i . Es decir aquella cartera factible que hace mínimo el riesgo. Si vamos variando el valor de r_0 , obtendríamos un conjunto de carteras eficientes que se denomina **frontera eficiente**. En este sentido, una solución completa del problema sería encontrar la función $R(r)$ que proporciona un mínimo riesgo en función de la rentabilidad mínima fijada por el inversor (o equivalentemente encontrar una función $r(R)$ que maximiza la rentabilidad en función del riesgo máximo).

Por último, vamos a añadir una condición que supondremos en el trabajo. La matriz de varianzas covarianzas podemos suponer que es semidefinida positiva sin más que aceptar que las rentabilidades de los activos en los que se puede invertir son linealmente independientes.

3. MODELOS FUZZY DE SELECCIÓN DE CARTERAS EFICIENTES

En el problema de selección de carteras hay que distinguir dos tipos de parámetros:

- Los que están relacionados con los datos históricos de los activos (rentabilidad, varianzas, covarianzas) que se tratan con técnicas estadísticas.
- Los parámetros que reflejan preferencias del inversor: mínima rentabilidad exigida, máximo riesgo admitido y otros criterios que aparecerán en las restricciones del modelo

Desde luego, el tipo de incertidumbre que afecta a estos tipos de parámetros son completamente diferentes. Para tratar la de los primeros, en el modelo de Markowitz se asume que el rendimiento de los activos se distribuye de acuerdo con una distribución normal multivariante. Sin embargo, los segundos presentan una incertidumbre no probabilística que podríamos identificar con la vaguedad. Por ejemplo, un inversor pide una rentabilidad de un 3%, pero seguramente aceptaría pequeñas variaciones si esto supone disminuir sensiblemente el riesgo. Además, la idea que persigue el inversor es mucho más próxima a la de una solución satisfactoria que a la de una cartera óptima. Sin duda, la teoría de conjuntos fuzzy resultará muy útil para manejar este tipo de incertidumbre.

En un entorno fuzzy, resulta mucho más natural la manera introducir en el modelo aquellos parámetros que dependen del criterio del inversor (por ejemplo, se puede encontrar un análisis fuzzy del uso de cotas superiores e inferiores en [8, 9, 14, 15]).

Nos vamos a centrar en construir un modelo fuzzy para el problema consistente en encontrar la cartera que maximiza la rentabilidad para un riesgo dado, con algunas condiciones adicionales. El caso de flexibilizar el modelo alternativo, minimizar el riesgo para una rentabilidad mínima dada, se construiría de manera análoga.

1. Información dada por el inversor

Para poder construir el modelo fuzzy no preguntaremos sólo por la mínima rentabilidad que está dispuesto a aceptar, sino que además debe proporcionarnos el nivel de tolerancia p_f que está dispuesto a admitir. En otras palabras, debe informarnos de cuál es el valor

$$r - p_f$$

por debajo del cual consideraría que la rentabilidad de la cartera es completamente inaceptable. También deberá informarnos sobre cuál es la mínima reducción del riesgo que le llevaría a modificar su umbral de rentabilidad. Es decir, debe proporcionarnos un parámetro p_g que indique la mínima reducción del riesgo que le parecería interesante.

2. Funciones de pertenencia

Por simplicidad, vamos a trabajar con funciones de pertenencia lineales a trozos. Así, el grado de factibilidad de una cartera dada x (que verifica todas las restricciones del problema) está dada por

$$\mu_{\tilde{C}}(x) = f(r), \quad (7)$$

donde r es la rentabilidad de la cartera x y

$$f(r) = \begin{cases} 1 & \text{si } r \geq r_0 \\ 1 - \frac{r - r_0}{p_f} & \text{si } r_0 - p_f < r < r_0 \\ 0 & \text{si } r \leq r_0 - p_f \end{cases} \quad (8)$$

Ahora consideramos R_0 de una cartera eficiente x_0 con rentabilidad r_0 . Nótese que x_0 es la solución óptima del problema de la cartera crisp obtenido para el valor de rentabilidad mínima r_0 . Definimos la función

$$g(R) = \begin{cases} 0 & \text{si } R \geq R_0 \\ \frac{R - R_0}{p_g} & \text{si } R_0 - p_g < R < R_0 \\ 1 & \text{si } R \leq R_0 - p_g \end{cases} \quad (9)$$

con la cuál tenemos que el objetivo del problema de la cartera fuzzy está determinado por

$$\mu_{\tilde{G}}(x) = g(R), \quad (10)$$

donde R es el riesgo de la cartera x . Esto refleja en cuanto mejora la cartera fuzzy a la cartera crisp x con riesgo óptimo R_0 . De acuerdo con el criterio del inversor, una cartera con un riesgo menor o igual que $R_0 - p_g$ satisface completamente el objetivo del inversor, $\tilde{G}$.

3. Grado de satisfacción global

El grado de pertenencia de una cartera al conjunto de decisión del problema $\tilde{D}$ lo expresaremos por λ y le llamaremos **grado global de satisfacción** de la cartera. Por tanto, el problema fuzzy de la cartera consiste en encontrar una cartera que maximice λ . Si el valor óptimo de λ es alto, esto significa que una pequeña reducción en la rentabilidad esperada ha supuesto una reducción importante en el riesgo esperado.

4. ALGORITMOS PARA LA SELECCIÓN DE CARTERAS

4.1. ALGORITMO EXACTO

El análisis de las versiones fuzzy de un problema de optimización crisp puede proporcionar mucha información relevante al decisor además de la solución fuzzy. En otros trabajos anteriores de los autores se ha comprobado claramente que esto es así sobre el problema de p -mediana fuzzy [1, 2]. Sin embargo, para obtener resultados similares para el problema de cartera fuzzy tenemos que estudiarlo tan explícitamente como sea posible. De ahí que nos parezca útil un algoritmo exacto,

aunque sólo pueda ser aplicado a instancias pequeñas del problema. Para solucionar casos reales es necesario recurrir a procedimientos heurísticos.

Hemos desarrollado un algoritmo exacto basado en la resolución explícita de las condiciones de Kuhn y Tucker que calcula la frontera eficiente, el mínimo y el máximo la rentabilidad y el riesgo de carteras factibles, etc. El método se ha implementado con el programa Mathematica[®] y se ha hecho para el modelo (6) por varios motivos: generaliza el problema Markowitz original, y, añadir o eliminar cualquier restricción adicional (lineal) sería sencillo. Por ejemplo, el manejo del problema de cartera cuando se permite el endeudamiento conduce a una simplificación fuerte del algoritmo.

Esencialmente, el algoritmo enumera todas las posibilidades de dividir el conjunto de variables en: las que se anulan, las que alcanzan la cota l_i y las que son mayores que l_i . Una vez que hemos escogido las variables nulas, llegamos al problema no lineal (sin variables binarias) quitando estas variables de (1). Entonces, las variables x_i para las que las cotas l_i son activas corresponden a las condiciones de holgura complementaria. Cuando se fija una de ellas, el problema se reduce a un problema de programación clásico sin desigualdades, para el que la solución explícita para cada r_0 es bien conocida (ver, por ejemplo, [3]). Añadiendo las soluciones que se obtienen con las condiciones de factibilidad primal y dual obtenemos una parametrización $R(r)$ de la frontera eficiente del problema, es decir el menor riesgo posible para una rentabilidad r , así como la cartera óptima $x(r)$.

Partiendo de la solución óptima crisp para una rentabilidad r_0 y riesgo R_0 , tenemos una función creciente $f(r)$ y una función decreciente $g(R(r))$ (ver (8) y (9)) que proporcionan respectivamente el grado de factibilidad y de mejora del objetivo de la cartera eficiente con rentabilidad r . Está claro que el valor máximo valor de

$$\lambda = \min\{f(r), g(R(r))\}$$

se alcanza en el punto en el que ambas funciones se cortan (ver Figura 1).

4.2. MODELOS HEURÍSTICOS

La técnica propuesta es un híbrido entre un Algoritmo Genético y un Algoritmo de Enfriamiento Simulado que ha sido adaptado del que los autores utilizan para el problema de la p-mediana (véase [2]). Utilizando un decrecimiento adaptativo del ratio de mutación mediante Enfriamiento Simulado, se consigue maximizar la búsqueda en el espacio de soluciones inicialmente, y centrarla en soluciones cercanas a la actual conforme se avanza, para que el algoritmo no tenga problemas en converger.

La configuración del algoritmo es clásica, con una población de 50 individuos, muestreo proporcional a la

función de fitness, selección estocástica universal, y elitismo en el reemplazo.

La mutación utiliza Enfriamiento Simulado y, para ayudar aún más a mantener la diversidad, se introduce el operador de invasión, que introduce en la población actual soluciones generadas de manera aleatoria.

Para el cruce, usamos el esquema clásico de cruce a partir de único punto de corte. Para adaptarlo a la representación con números reales, mantenemos una parte fija de un padre, y la segunda parte (del segundo padre) es variable, en el sentido de que se añade al hijo mientras se cumpla la restricción de que la suma de sus genes es menor o igual que la unidad.

Dadas las características particulares del problema, el algoritmo debe funcionar de una manera muy específica. Primero evoluciona sin tener en cuenta las restricciones fuzzy; es decir, intenta resolver el problema de cartera clásico. Una vez encuentra una solución factible para este problema (riesgo menor o igual que R), evoluciona mejorando el coste de la solución crisp. El algoritmo no busca soluciones fuzzy mientras que no haya soluciones parciales que tengan un fitness mayor que cero; es decir, mientras no encontremos al menos una solución fuzzy con un grado de satisfacción $\lambda > 0$.

Este esquema de funcionamiento, unido a la probada eficacia del algoritmo híbrido [2], permite obtener la solución a ambos problemas, crisp y fuzzy, en tiempos reducidos.

5. PRUEBAS COMPUTACIONALES

Ejemplo 1. Hemos usado cinco activos de los datos históricos introducidos por Markowitz ([11]). En la Tabla 1 aparecen las rentabilidades de American Tobacco, AT&T, United States Steel, General Motors y Atcheson & Topeka & Santa Fe.

Tabla 1: Rentabilidad de cinco activos.

	AmT	AT&T	USS	GM	AT&S
1937	-0,305	-0,173	-0,318	-0,477	-0,457
1938	0,513	0,098	0,285	0,714	0,107
1940	0,055	0,2	-0,047	0,165	-0,424
1941	-0,126	0,03	0,104	-0,043	-0,189
1942	-0,003	0,067	-0,039	0,476	0,865
1943	0,428	0,3	0,149	0,225	0,313
1944	0,192	0,103	0,26	0,29	0,637
1945	0,446	0,216	0,419	0,216	0,373
1946	-0,088	-0,046	-0,078	-0,272	-0,037

Suponemos que el inversor no está interesado en invertir menos del 10% del total del capital. Es decir o invierte al menos un 10% o no invierte en el activo. i. e., $l_i=0.1$ para todo i . La rentabilidad y riesgo de las carteras eficientes para esta instancia varía entre los siguientes rangos:

$$9.27\% \leq \text{rentabilidad} \leq 14.26\%,$$

$$2.05\% \leq \text{riesgo} \leq 11.38\%$$

Asumimos que el inversor espera una rentabilidad de, al menos, un 14.2%. La cartera óptima crisp es $x_1 = (0, 0, 0, 0.849, 0.151)$ con un riesgo de 11.27%. Tomando esta solución como punto de partida, estudiamos la posibilidad de reducir la rentabilidad en, a lo sumo, un 1% ($p_f = 0.01$), considerando el 2% de reducción del riesgo como completamente satisfactoria ($p_g = 0.02$). La cartera fuzzy óptima es $x_2 = (0.235, 0, 0, 0.765, 0)$ que proporciona una rentabilidad del 13.9 % con un riesgo del 9.8%. Nótese que la inversión propuesta en x_2 es muy diferente de la que se propone en x_1 y habría pasado inadvertida si sólo se hubiese resuelto el problema crisp.

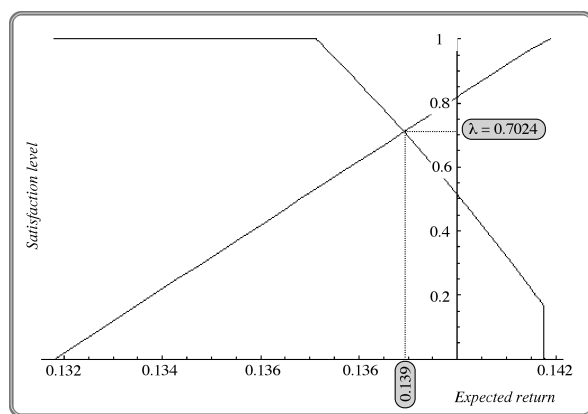


Figura 1: Funciones de pertenencia para la factibilidad y mejora del objetivo

La Figura 1 muestra las funciones $f(r)$ y $g(R(r))$ obtenidas con $r = 0.139$ que se corresponde con un nivel de satisfacción global $\lambda = 0.702438$. En la práctica, una reducción relativa de alrededor del 2% de la rentabilidad proporciona una reducción relativa del 12.5% en el riesgo crisp. Ahora debe ser el inversor quien considere si la cartera x_2 es una alternativa provechosa para él.

Es importante tener en cuenta que la mejora de la solución fuzzy respecto de la crisp puede depender de forma muy irregular de los niveles fijados. Veámoslo en el ejemplo siguiente.

Ejemplo 2. Consideramos una instancia muy simple expuesta por Markowitz [11], determinada por el vector de rentabilidades

$$(r_1, r_2, r_3) = (0.062, 0.146, 0.128)$$

y la matriz de varianzas covarianzas

$$Q = \begin{bmatrix} 0.0146 & 0.0187 & 0.0145 \\ 0.0187 & 0.0854 & 0.0104 \\ 0.0145 & 0.0104 & 0.0289 \end{bmatrix}$$

Tomamos $l_i = 0.1$ para todo i , $p_f = 0.01$ y $p_g = 0.02$.

La Figura 2 muestra la reducción relativa de la rentabilidad y del riesgo (ambos en porcentaje) obtenidos con la solución fuzzy para cada nivel de rentabilidad r_0 . Se ve que la mejora en la solución es mayor para valores grandes de r_0 , pero no de forma monótona.

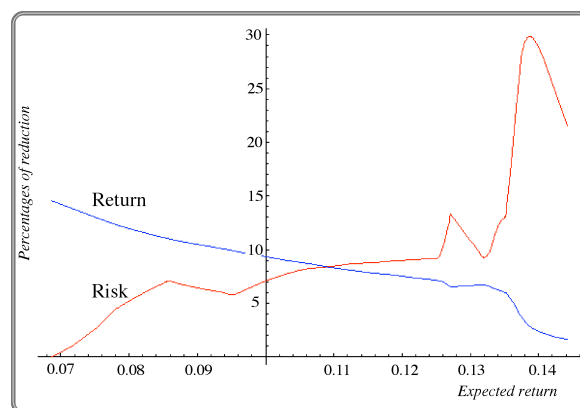


Figura 2: Reducciones del riesgo y rentabilidad en la cartera fuzzy óptima.

Ejemplo 3. Consideramos las rentabilidades de los miércoles (desde enero de 1998 a marzo de 2003) de 20 activos del IBEX35 Acesa(ACE), Arcelor (ACR), ACS, Altadis (ALT), BBVA, Bankinter (BKT), Dragados (DRC), Endesa (ELE), FCC, Iberdrola (IBE), Metrovacesa (MVC), NH Hoteles(NHH), Banco Popular (POP), Repsol (REP), SCH (SAN), Telefónica (TEF), Unión Fenosa (UNF), Vallermosto (VAL), Acerinox (ACX), Acciona (ANA)

En la Tabla 2 se muestran los resultados de la cartera óptima crisp y fuzzy obtenida con la metaheurística.

Tabla 2. Solución de los problemas crisp y fuzzy con datos del IBEX35.

	$R = 0.000494$		$R = 0.000839$	
	$p_f = 0.0002$ Crisp	$p_g = 0.002$ Fuzzy	$p_f = 0.0002$ Crisp	$p_g = 0.002$ Fuzzy
ACE	0.316	0.200	0	0
ACR	0.011	0	0	0
ALT	0.168	0.199	0.207	0.162
DRC	0.019	0.205	0.353	0.461
IBE	0.165	0.014	0	0
MVC	0.050	0.102	0.157	0.098
NHH	0.011	0.033	0.056	0.065
POP	0.62	0	0	0
REP	0.031	0	0.001	0
UNF	0	0.077	0.0395	0.052
VAL	0.036	0	0	0
ACX	0.053	0.069	0.051	0.017
ANA	0.078	0.100	0.139	0.145
λ	0	0.454078	0	0.145928
Rentabilidad	0.00202704	0.0029352	0.00353188	0.00382374
Riesgo	0.000491306	0.000603184	0.00083643	0.0010098

A pesar de que en los dos escenarios analizados, $R_1=0.000494$ y $R_2=0.000839$, se consiguen grados de satisfacción mayores que 0, en el caso R_1 la mejora de la cartera fuzzy respecto de la crisp es mucho más evidente. Desde el punto de vista computacional, es importante resaltar que el tiempo utilizado, en media, por la heurística ha sido de 1.61 segundos.

5. CONCLUSIONES

Se comprueba que el tratamiento fuzzy de las preferencias del inversor en el modelo de selección de carteras proporciona soluciones alternativas que pueden resultar más satisfactorias para el decisor que las proporcionadas por los modelos crisp.

Hemos propuesto un marco de trabajo para abordar la selección de carteras eficientes con metodología fuzzy. Consiste en aceptar carteras parcialmente factibles (en las que no se verifica alguna condición dada por el inversor) si esto supone una mejora considerable de la rentabilidad o del riesgo. Para resolver el problema presentamos un algoritmo exacto, basado en las condiciones de Kuhn y Tucker, y un método heurístico.

El algoritmo exacto se utiliza para resolver problemas pequeños y para obtener información de las soluciones que puede resultar muy interesante para abordar, en trabajos futuros, un análisis de sensibilidad de la frontera eficiente y de las tolerancias fijadas por el inversor p_f y p_g . Por otro lado, las técnicas heurísticas permiten trabajar con instancias grandes del problema.

Agradecimientos

Los autores agradecen la financiación proporcionada por los proyectos de investigación TIN2008-06872-C04-02 y TIN2008-06872-C04-03 del Ministerio de Ciencia e Innovación del gobierno español.

Referencias

- [1] M. J. Canós, C. Ivorra, y V. Liern, Marginal analysis for the fuzzy p-median problem, *European Journal of Operational Research*, 191 (2008), 264-271.
- [2] J.M. Cadenas, M.J. Canós, M.C. Garrido, C. Ivorra, y V. Liern, Soft-computing based heuristics for location on networks: The p-median problem, *Applied Soft Computing*, (2008), disponible online.
- [3] H. Chi-Fu y R. H. Litzenberg, "Foundations for financial economics", North-Holland, Amsterdam, 1988.
- [4] E. F. Fama, "Foundations of Finance", Basil Blackwell, Oxford, 1977.
- [5] Inuiguchi y J. Ramík, Possibilistic linear programming: a brief review of fuzzy mathematical programming and a comparison with stochastic programming in portfolio selection problem, *Fuzzy Sets and Systems*, 111 (2000), 3-28.
- [6] H. Konno y H. Yamazaki, Mean-absolute deviation portfolio optimization model and its applications Tokyo Stock Market, *Management Science*, 37 (1991), 519-531.
- [7] Y. J. Lai y C. L. Hwang, "Fuzzy Mathematical Programming: Theory and Applications", Springer, Berlin, 1992.
- [8] T. León, V. Liern y E. Vercher, Fuzzy mathematical programming for portfolio management, 24th EURO on Financial Modelling, Springer, Heidelberg, (2000), 241-256.
- [9] T. León, V. Liern y E. Vercher, Viability of infeasible portfolio selection problems: A fuzzy approach, *European Journal of Operational Research*, 139 (2002), 178-189.
- [10] H. M. Markowitz, Portfolio selection, *Journal of Finance*, 7 (1952), 79-91.
- [11] H. M. Markowitz, "Portfolio selection: Efficient Diversification of Investments", John Wiley, New York, 1959.
- [12] W. F. Sharpe, "Portfolio theory and capital market", McGraw-Hill, New York, 1970.
- [13] M. G. Speranza, Linear programming model for portfolio optimization, *Finance* 14 (1993), 107-123.
- [14] R. E. Steuer, Y. Qui y M. Hirschberger, Suitable-portfolio investors, nondominated frontier sensitivity, and the effect of multiple objectives on standard portfolio selection, *Annals of Operations Research*, 152 (2007), 297-317.
- [15] E. Vercher, J. D. Bermúdez y J. V. Segura, Fuzzy portfolio optimization under downside risk measures, *Fuzzy Sets and Systems*, 158 (2007), 769-782.
- [16] J. Watada, Fuzzy portfolio selection and its applications to decision making, *Tatra Mountains Mathematical Publications*, 13 (1997), 219-248.

TOMA DE DECISIONES EN ENSAMBLES BASADOS EN ÁRBOL FUZZY

Piero Bonissone¹ José M. Cadenas² M. Carmen Garrido²
 R. Andrés Díaz-Valladares³ Antonio Muñoz⁴

¹ GE Global Research. NY, USA, bonissone@crd.ge.com

² Universidad de Murcia. Spain, {jcadenas,carmengarrido}@um.es

³ Universidad de Montemorelos. Mexico, rdiaz@um.edu.mx

⁴ Instituto de Educación Secundaria. Murcia. Spain, aml-jmmr@ono.com

Resumen

En este trabajo, consideramos los principales ensambles de la literatura que usan un árbol de decisión como clasificador base. A partir de ellos, construimos sus equivalentes usando como clasificador base un árbol de decisión fuzzy, realizando una comparación entre ambos grupos: ensambles basados en árboles de decisión crisp versus ensambles basados en árboles de decisión fuzzy. Con los ensambles basados en árboles de decisión fuzzy pretendemos aprovechar la mejora que proporcionan las técnicas basadas en ensambles con respecto a los clasificadores individuales, incrementar la robustez al ruido y permitir el tratamiento de incertidumbre.

Palabras Clave: árbol de decisión fuzzy, ensamble, método de combinación.

1 INTRODUCCIÓN

Con el objetivo de mejorar la precisión de las predicciones, ha surgido un interés creciente en los últimos años en la definición de métodos que combinan hipótesis. Estos métodos construyen un conjunto de hipótesis (ensamble), y combinan las predicciones del conjunto de alguna manera (normalmente por votación) para clasificar nuevos ejemplos. La precisión obtenida por esta combinación de hipótesis supera, generalmente, la precisión de cada componente individual del conjunto [1]. A estas técnicas de combinación de hipótesis se les denomina métodos multclasificadores o métodos de ensamble [7].

Se han propuesto numerosos métodos de ensamble donde cada componente individual se construye usando el mismo método clasificador, denominado cla-

sificador base, modificando el conjunto de datos de aprendizaje de cada uno de ellos, o el conjunto de atributos del conjunto de datos de aprendizaje o introduciendo algún factor aleatorio en el proceso de construcción del clasificador [7]. En este trabajo nos centramos en ensambles que utilizan como clasificador base un árbol de decisión.

Sin embargo, aunque los árboles de decisión son interpretables, eficientes y capaces de trabajar con bases de datos grandes, son muy inestables cuando pequeñas cantidades de ruido se introducen en los datos de aprendizaje. Por esta razón, la lógica fuzzy se ha incorporado en las técnicas de construcción de árboles de decisión. En [11, 12, 13] podemos encontrar propuestas en las que la teoría de conjuntos fuzzy y su capacidad de razonamiento aproximado se ha combinado con la técnica de construcción de árboles de decisión. Esta integración preserva las ventajas de ambos componentes: el tratamiento de la incertidumbre con la comprensibilidad de las variables lingüísticas de la lógica fuzzy y la popularidad y fácil aplicación de los árboles de decisión. Los árboles resultantes muestran un incremento de su robustez ante el ruido y una aplicabilidad extendida a la incertidumbre o contextos vagos.

En este trabajo, consideramos los principales ensambles basados en árboles de decisión de la literatura y construimos sus equivalentes usando como clasificador base un árbol de decisión fuzzy, realizando una comparación entre ambos grupos: ensambles basados en árboles de decisión crisp versus ensambles basados en árboles de decisión fuzzy. Con los ensambles basados en árboles de decisión fuzzy pretendemos aprovechar la mejora en resultados que proporcionan las técnicas basadas en ensambles con respecto a los clasificadores individuales, incrementar la robustez al ruido de los ensambles basados en árboles de decisión crisp y permitir el manejo de incertidumbre.

De esta forma, en este trabajo presentamos en la

Sección 2 el árbol de decisión fuzzy que usaremos como clasificador base, mostrando sus fases de aprendizaje e inferencia. En la Sección 3 definimos los distintos ensambles basados en árboles de decisión fuzzy indicando sus fases de aprendizaje e inferencia. En la Sección 4, aplicamos los distintos ensambles a una serie de bases de datos y, por último, mostramos las conclusiones obtenidas de nuestro trabajo.

2 UN ÁRBOL DE DECISIÓN FUZZY

En esta sección describimos el árbol fuzzy que va a actuar como clasificador base de los ensambles. En primer lugar mostraremos el algoritmo de aprendizaje del árbol a partir de un conjunto de ejemplos y después pasaremos a describir el algoritmo de inferencia que permite predecir la clase desconocida de un ejemplo dado.

El conjunto de ejemplos E a partir del cual se construye el árbol, está formado por ejemplos descritos por atributos que pueden ser nominales y numéricos y donde al menos habrá un atributo nominal que actúa como atributo clase.

El algoritmo mediante el cual construimos el árbol fuzzy está basado en el Algoritmo ID3 donde todos los atributos numéricos han sido discretizados mediante una partición fuzzy previamente obtenida mediante un algoritmo de discretización basado en un algoritmo genético [6]. La partición fuzzy de cada atributo numérico garantiza: la completitud (ningún valor del dominio queda fuera de la partición fuzzy), y es una partición fuzzy fuerte (satisface que $\forall x \in E$, $\sum_{i=1}^f \mu_{A_i}(x) = 1$, donde $A_1, \dots, A_f$ son los conjuntos fuzzy de la partición). Por lo tanto, el dominio de cada atributo numérico es representado en el árbol mediante los conjuntos fuzzy trapeczoidales $A_1, \dots, A_f$.

A cada ejemplo e usado en el aprendizaje del árbol t se le asigna un valor inicial igual a 1 ($\chi_{t,raiz}(e) = 1$) indicando que inicialmente el ejemplo está sólo en el nodo raíz de t . Dicho valor seguirá siendo 1 mientras el ejemplo e no pertenezca a más de un nodo durante el proceso de construcción del árbol. En un árbol clásico, un ejemplo sólo puede pertenecer a un nodo en cada momento, por lo tanto, su valor inicial (si existiera) no se modificaría a lo largo del proceso de construcción. En el caso de un árbol fuzzy, este valor es modificado en dos situaciones:

- Cuando el ejemplo e tiene un valor missing en un atributo que se usa como test en un nodo. En este caso el ejemplo desciende a cada nodo hijo con un valor modificado como $\chi_{t,nodo\_hijo}(e) =$

$$\chi_{t,nodo}(e) \times \frac{1}{numero\_salidas_{test}}.$$

- De acuerdo al grado de pertenencia de e a los distintos conjuntos fuzzy de la partición cuando el test en un nodo está basado en un atributo numérico. En este caso, el ejemplo desciende a aquellos nodos hijos a los que el ejemplo pertenezca con grado mayor de 0 ($\mu_{conjunto\_fuzzy\_particion}(e) > 0$). Por las características de las particiones que usamos, el ejemplo podrá descender como mucho a dos nodos hijos. En este caso, $\chi_{t,nodo\_hijo}(e) = \chi_{t,nodo}(e) \times \mu_{conjunto\_fuzzy\_particion}(e)$.

Podemos decir que el valor $\chi_{t,nodo}(e)$ indica el grado con el cual el ejemplo satisface las condiciones que conducen a nodo $nodo$ del árbol t .

En el Algoritmo 1, se muestra el algoritmo de construcción de un árbol fuzzy a partir de un conjunto de ejemplos.

Algoritmo 1 - Aprendizaje de un árbol fuzzy

ArbolFuzzy(*in* : E , *Particion Fuzzy*;
 out : *Arbol Fuzzy* t)

begin

1. Sea el nodo raíz constituido por los ejemplos de E con valores $\chi_{t,raiz}(e) = 1$.
2. Sea M el conjunto de atributos que describen los ejemplos de E y donde los atributos numéricos han sido discretizados según la *Particion Fuzzy*.
3. Seleccionar de M un atributo como test en el nodo N . Para ello:
 - 3.1. Calcular la ganancia de información para cada atributo en M usando los valores $\chi_{t,N}(e)$ de cada ejemplo e en el nodo N .
 - 3.2. Elegir el atributo con ganancia máxima.
4. Dividir N en subnodos de acuerdo a las posibles salidas del atributo seleccionado en el paso previo y eliminar dicho atributo de M . Sean $E_{N_1}, \dots, E_{N_{numero\_salidas_{test}}}$, los conjuntos de datos en cada subnodo.
5. Repetir los pasos 3 y 4 para cada E_{N_i} y M con $i = 1, \dots, numero\_salidas_{test}$ hasta que se cumpla la condición de parada.

end

Hay que destacar, como se indica en el paso 4 del Algoritmo 1, que una vez seleccionado un atributo como test en un nodo, dicho atributo no volverá a ser seleccionado debido al hecho de que todos los atributos o son nominales o están particionados.

La condición de parada en el Algoritmo 1 está definida por la primera condición alcanzada de las siguientes:

(1) nodo puro, (2) conjunto M vacío, (3) alcanzar el mínimo número de ejemplos permitido en un nodo.

Una vez construido el árbol lo usamos para inferir la clase desconocida de un nuevo ejemplo siguiendo el Algoritmo 2.

Algoritmo 2 - Inferencia en un árbol fuzzy

ArbolFuzzy(*in* : *Arbol Fuzzy t*, *Particion Fuzzy*,
Ejemplo e; *out* : *Clase c*)

begin

1. Dado el ejemplo e a clasificar con valor inicial $\chi_{t,raiz}(e) = 1$, recorrer el árbol desde el nodo raíz.
2. Obtener el conjunto de hojas alcanzadas por e . e puede ramificarse por más de una rama si en algún test basado en un atributo numérico, e pertenece a más de un elemento de la partición de dicho atributo con grado mayor que 0. También se puede ramificar por más de una rama si tiene al menos un valor desconocido en los atributos usados por el árbol como test.
3. Para cada hoja alcanzada por e , calcular el soporte para cada clase. El soporte de una clase en una hoja dada n se obtiene sumando los valores $\chi_{t,n}(ejemplo)$ de todos los ejemplos que alcanzan dicha hoja durante la fase de aprendizaje y que pertenecen a dicha clase.
4. Obtener la decisión del árbol c a partir de la información proporcionada por el conjunto de hojas alcanzadas N_t y el valor $\chi_{t,n}(e)$; $n = 1, \dots, |N_t|$ con el que el ejemplo activa o alcanza cada hoja.

end

Como se indica en el paso 4 del Algoritmo 2, una característica de un árbol fuzzy frente a un árbol crisp es que un ejemplo, en su recorrido por el árbol, puede alcanzar o activar más de una hoja. Por lo tanto, al hacer la inferencia se pueden usar múltiples estrategias. Algunas de ellas son:

- Considerar solamente la información proporcionada por la hoja activada con mayor valor por el ejemplo a clasificar. En este caso, el árbol decide la clase con mayor soporte en esta hoja.
- Considerar todas las hojas alcanzadas por el ejemplo y obtener la decisión final del árbol en base al soporte proporcionado para cada clase por cada una de ellas y el valor $\chi_{t,n}(e)$ con el que el ejemplo e activa la hoja n , usando operadores de agregación como voto mayoritario, mínimo, máximo, promedio y versiones ponderadas de todos ellos.

3 ENSAMBLES BASADOS EN ÁRBOLES DE DECISIÓN FUZZY

En esta sección vamos a presentar cómo se construyen los principales ensambles basados en árboles que aparecen en la literatura a partir del algoritmo de construcción de un árbol fuzzy mostrado en la sección anterior (Algoritmo 1). Además presentaremos cómo realizar la inferencia con dichos ensambles.

3.1 FASE DE APRENDIZAJE

Nos vamos a centrar en los ensambles Bagging [4], Boosting [8, 14] y Random Forest [5], debido a que son los ensambles más utilizados en la literatura.

Bagging [4] es una de las técnicas más antigua, simple y bien conocida para crear un ensamble de clasificadores. En bagging, la diversidad se obtiene construyendo cada clasificador con un conjunto de ejemplos diferente que se obtienen seleccionando ejemplos con reemplazamiento del conjunto de ejemplos original. Por lo tanto, para construir un ensamble basado en bagging formamos diferentes conjuntos de ejemplos de la forma comentada anteriormente y le aplicamos a cada uno de ellos el algoritmo base que en nuestro caso es el Algoritmo 1.

Boosting [8, 14] es un algoritmo que crea un ensamble de clasificadores añadiendo un clasificador en cada paso. En el conjunto de ejemplos original, cada ejemplo tiene asignado un peso. El clasificador que se añade en el paso k es aprendido a partir del conjunto de ejemplos original donde el peso de cada ejemplo ha sido modificado en la iteración $k - 1$ del algoritmo. En cada iteración, boosting aprende un modelo que minimiza la suma de los pesos de los ejemplos clasificados erróneamente. Los errores en cada iteración se utilizan para actualizar los pesos de los ejemplos del conjunto de entrenamiento de manera que se incremente el peso de los ejemplos erróneos y se reduzca el peso de los ejemplos clasificados correctamente. Cuando usamos como algoritmo base el Algoritmo 1 en un ensamble boosting, el peso de cada ejemplo e en la base de datos original será el valor $\chi_{t,raiz}(e)$ definido anteriormente y estará inicializado a 1.

Finalmente, Breiman [5] presentó el ensamble denominado random forest donde se utiliza bagging junto a una selección aleatoria de atributos. En cada nodo de cada árbol del bosque, se selecciona aleatoriamente un subconjunto de los atributos disponibles en ese nodo y se elige el mejor de ellos de acuerdo al criterio de división empleado en el algoritmo base. El número de atributos seleccionado aleatoriamente es un parámetro de entrada y diversos estudios avalan que el valor $\log_2(|M| - 1)$, donde $|M|$ es el cardinal del conjunto de

atributos disponibles en ese nodo en cuestión, produce un buen comportamiento del ensamble. Si usamos el Algoritmo 1 como algoritmo base de un ensamble basado en random forest debemos modificar el paso 3 de la siguiente forma:

Modificación Algoritmo 1

3. Seleccionar aleatoriamente de M un subconjunto de atributos denominamos RM . Seleccionar de RM un atributo (test) en el nodo N . Para ello:
 - 3.1. Calcular la ganancia de información para cada atributo de RM usando los valores $\chi_{t,N}(e)$ de cada ejemplo e en el nodo N .
 - 3.2. Elegir el atributo con ganancia máxima.
-

Además, dado que el ensamble random forest hace uso de bagging, el conjunto de entrenamiento de cada clasificador se forma seleccionando ejemplos con reemplazamiento a partir del conjunto de ejemplos original. Por lo tanto, un ensamble random forest está basado en un doble nivel de aleatoriedad para conseguir diversidad.

3.2 FASE DE INFERENCIA

En esta sección vamos a describir cómo realizar la inferencia o clasificación de un nuevo ejemplo una vez que el ensamble ha sido construido. Dado que los ensambles a los que nos referimos usan como clasificador base el árbol fuzzy del Algoritmo 1 vamos, en primer lugar, a comentar aquellos aspectos que se pueden modificar en la inferencia de los ensambles basados en árboles crisp debido al uso del árbol fuzzy como clasificador base.

Como comentamos al presentar el Algoritmo 2, una característica del árbol fuzzy es que un ejemplo a clasificar puede alcanzar o activar más de un nodo hoja. Este hecho nos permite poder tener en cuenta la decisión de cada hoja alcanzada de cada árbol del ensamble como veremos en la definición de posibles formas de inferencia que veremos a continuación. Por otro lado, las hojas son alcanzadas o activadas por el ejemplo e con un valor $\chi_{t,n}(e); n = 1, \dots, |N_t|$. Este valor también puede ser considerado en la definición del proceso inferencia para obtener la decisión final del ensamble acerca de la clase. Además, en la decisión de cada hoja alcanzada se considera la suma de los valores $\chi_{t,n}(ejemplo)$ de los ejemplos que alcanzaron esa hoja durante la fase de aprendizaje en lugar del número de ejemplos en dicha hoja.

Teniendo en cuenta las consideraciones anteriores, en un ensamble basado en árboles fuzzy como clasificador base vamos a considerar dos estrategias generales de inferencia:

Estrategia 1: Consiste en agregar la información de las diferentes hojas alcanzadas de cada árbol para obtener la decisión individual de cada árbol y después agregar la información de cada árbol para generar la decisión global del ensamble.

Estrategia 2: Consiste en agregar la información de las hojas alcanzadas de cada árbol para generar la decisión del ensamble.

Para definir métodos de combinación concretos para cada una de estas estrategias, vamos a considerar qué información podemos obtener de cada hoja alcanzada por el ejemplo e en un ensamble. Esta información puede ser:

- voto mayoritario: la hoja puede proporcionar voto 1 a la clase con mayor soporte y 0 para las demás. El voto de cada hoja tiene la misma importancia.
- voto mayoritario ponderado: la hoja alcanzada asigna un voto con valor $\chi_{t,n}(e)$ a la clase con mayor soporte y 0 para las demás. De esta forma, el grado de activación de cada hoja alcanzada por el ejemplo e es considerado en el voto de la hoja.

Teniendo en cuenta lo anterior, vamos a definir algunos métodos de combinación concretos:

- Voto mayoritario simple.
 - En la estrategia 1 (SM1): Cada hoja alcanzada por el ejemplo e en el ensamble proporciona su voto mayoritario. Estos votos son sumados por árbol, es decir, se suman para cada clase los votos de las hojas alcanzadas pertenecientes a un mismo árbol de forma que dicho árbol realiza un voto mayoritario por la clase más votada entre sus hojas alcanzadas. El ensamble decide la clase más votada por sus árboles.
 - En la estrategia 2 (SM2): Cada hoja alcanzada por el ejemplo e en el ensamble proporciona su voto mayoritario. El ensamble decide la clase más votada por las hojas alcanzadas.
- Voto mayoritario ponderado por el valor de activación de la hoja.
 - En la estrategia 1 (MWL1): Cada hoja alcanzada por el ejemplo e en el ensamble proporciona su voto mayoritario ponderado. Para cada clase, los valores de estos votos son sumados por árbol, de forma que dicho árbol realiza un voto mayoritario por la clase con mayor valor final. El ensamble decide la clase más votada por sus árboles.
 - En la estrategia 2 (MWL2): Cada hoja alcanzada por el ejemplo e en el ensamble proporciona

su voto mayoritario ponderado. El ensamble decide la clase con mayor valor una vez sumados para cada clase los votos ponderados de cada hoja.

- Voto mayoritario ponderado por el valor de activación de la hoja y un peso por árbol.

- En la estrategia 1 (MWLT1): En este método asignamos un peso a cada árbol del ensamble en función de la proporción de errores cometidos al clasificar un conjunto de ejemplos con dicho árbol. De esta forma, el método de combinación queda definido de forma que cada hoja alcanzada por el ejemplo e en el ensamble proporciona su voto mayoritario ponderado. Para cada clase, los valores de estos votos son sumados por árbol, de forma que dicho árbol vota con su peso por la clase con mayor valor final. El ensamble decide la clase con mayor valor una vez sumados para cada clase los votos de cada árbol.

- En la estrategia 2 (MWLT2): De nuevo, en este método asignamos un peso a cada árbol del ensamble en función de la proporción de errores cometidos al clasificar un conjunto de ejemplos con dicho árbol. De esta forma, cada hoja alcanzada por el ejemplo e en el ensamble proporciona su voto mayoritario ponderado que en este caso considera el valor con el que fue activada multiplicado por el peso asignado al árbol al que pertenece. El ensamble decide la clase con mayor valor una vez sumados para cada clase los votos ponderados de cada hoja.

- Voto basado en una única hoja y ponderado por el peso del árbol para la estrategia 1 (WT1): En la decisión de cada árbol sólo se considera la hoja activada con mayor valor por el ejemplo e y el árbol asigna su voto, ponderado por el peso que tiene asignado como árbol, a la clase con mayor soporte en dicha hoja. El ensamble decide la clase con mayor valor una vez sumados para cada clase los votos ponderados de cada árbol.

En la siguiente sección, todos estos métodos de combinación serán aplicados a los ensambles basados en árboles fuzzy comentados anteriormente y compararemos sus resultados con los proporcionados por ensambles basados en árboles crisp de la literatura.

4 EXPERIMENTOS

En esta sección hemos seleccionado una serie de bases de datos de la UCI [2] para obtener una serie de resultados experimentales sobre los distintos ensambles basados en árboles fuzzy que hemos mencionado en este trabajo. Además se han aplicado dichas bases de datos a las versiones crisp de los ensambles anteriores,

es decir, ensambles que usan un árbol crisp como clasificador base. La descripción de las bases de datos empleadas se muestra en la Tabla 1.

Tabla 1: Descripción de las Bases de datos

Base de datos	Abbr	$ E $	$ M $	nº clases
Contraceptive Method	CMC	1473	9	3
Statlog Heart Disease	HEA	270	13	2
Ionosphere	ION	351	34	2
Iris Plants	IRP	150	4	3
Pima Indian Diabetes	PID	532	7	2
Attitude Smoking	SMO	2855	8	3

Para ejecutar los ensambles fuzzy hemos implementado el árbol fuzzy del Algoritmo 1 y lo hemos integrado en las distintas técnicas basadas en ensambles. Para ejecutar los ensambles crisp hemos usado las implementaciones bagging, boosting y random forest que proporciona la herramienta Weka [10] usando el árbol J48 como clasificador base. La ejecución de cada experimento ha sido realizada con los mismos parámetros o equivalentes. Los resultados muestran el porcentaje medio de precisión (media y desviación estándar) de una validación cruzada de 4×5 (repetimos 4 veces una validación cruzada de tamaño 5).

En la Tabla 2 se muestran los resultados para los ensambles que usan un árbol crisp como clasificador base.

Tabla 2: Ensambls basados en árboles crisp

BD	J48	Tamaño ensambles	Random Forest	Bagging	Boosting
CMC	50.98 ₍₀₎	120	51.44 _(0.35)	54.36 _(0.32)	51.12 ₍₀₎
HEA	77.78 ₍₀₎	120	81.48 _(0.78)	81.30 _(0.32)	81.48 ₍₀₎
ION	89.74 ₍₀₎	175	93.38 _(0.24)	92.24 _(0.13)	94.59 ₍₀₎
IRP	96.00 ₍₀₎	120	95.67 _(0.34)	96.00 _(0.00)	92.67 ₍₀₎
PID	75.94 ₍₀₎	50	75.75 _(0.55)	76.69 _(0.59)	74.44 ₍₀₎
SMO	69.53 ₍₀₎	75	61.05 _(0.14)	65.05 _(0.19)	58.63 ₍₀₎

En la Tabla 3 se muestran los resultados obtenidos con los ensambles basados en árbol fuzzy.

Tabla 3: Ensambls basados en árboles fuzzy

BD	Árbol fuzzy	Tamaño ensambles	Random Forest fuzzy	Bagging fuzzy	Boosting fuzzy
CMC	47.17 _(2.62)	120	53.31 _(2.32)	51.43 _(2.10)	49.59 _(1.78)
HEA	74.26 _(6.58)	120	82.31 _(4.70)	81.02 _(5.13)	75.37 _(5.60)
ION	92.59 _(3.29)	175	94.66 _(2.19)	93.95 _(3.12)	94.59 _(2.25)
IRP	97.00 _(2.08)	120	97.33 _(2.14)	96.67 _(2.53)	97.50 _(2.60)
PID	71.85 _(3.95)	50	79.32 _(3.72)	77.54 _(3.82)	75.89 _(3.76)
SMO	55.29 _(2.42)	75	69.54 _(1.97)	69.48 _(1.87)	59.27 _(3.52)

Hemos realizado un análisis de los resultados usando técnicas estadísticas. Siguiendo la metodología de [9], usamos tests no paramétricos. Para ello usamos el test de Friedman y el procedimiento de Benjamin-Hochberger [3] como test post-hoc.

Para analizar el rendimiento de los 6 ensambles y los 2 clasificadores base, hemos aplicado primero el test de Friedman. Se rechaza la hipótesis nula con un nivel de confianza del 95%, es decir, aceptamos que hay diferencias significativas entre los métodos. Cuando hacemos

el análisis post-hoc, obtenemos que el ensamble Random Forest fuzzy tiene diferencias significativas con todos los demás métodos con un nivel de confianza del 95%, siendo el mejor de ellos. Además, obtenemos que Bagging fuzzy es mejor que el árbol fuzzy, Boosting y J48 con un nivel de confianza del 93.7% y mejor que Boosting fuzzy y Random Forest al 90.6%. Por último, obtenemos que Boosting fuzzy es mejor que el árbol fuzzy con un nivel de confianza del 92.15%.

De los métodos de combinación presentados en la sección anterior, los que obtienen los mejores resultados (en 13 de las 18 pruebas realizadas) son los denominados MWLT1 y MWLT2.

5 CONCLUSIONES

En este trabajo hemos presentado los principales ensambles de la literatura basados en árboles de decisión sustituyendo el árbol crisp por un árbol fuzzy como clasificador base de los ensambles. Hemos indicado los principales aspectos a modificar tanto en la construcción como en la inferencia de dichos ensambles, presentando diversos métodos de combinación para fusionar las salidas de los árboles del ensamble.

Hemos llevado a cabo la comparación de los ensambles basados en árboles crisp y los basados en árboles fuzzy realizando una serie de experimentos. Los resultados obtenidos para un conjunto de bases de datos muestran que los ensambles fuzzy obtienen siempre un mejor resultado que el clasificador individual fuzzy. Por otro lado, el ensamble fuzzy basado en random forest es el que consistentemente obtiene los mejores resultados y muestra el mejor comportamiento. Los métodos de combinación que obtienen los mejores resultados son los métodos basados en voto mayoritario ponderado por el valor de activación de la hoja y peso por árbol. Estos resultados han sido validados mediante la realización de un análisis estadístico.

Agradecimientos

Los autores agradecen al “Ministerio de Educación y Ciencia” y al “Fondo Europeo de Desarrollo Regional” por el soporte, bajo el proyecto TIN2008-06872-C04-03, para el desarrollo de este trabajo. Gracias también a la “Fundación Séneca” por el programa para Grupos de Investigación de Excelencia con código 04552/GERM/06.

Referencias

[1] H. Ahn, H. Moon, J. Fazzari, N. Lim, J. Chen, R. Kodell. Classification by ensembles from random partitions of high dimensional data. *Computatio-*

nal Statistics and Data Analysis, 51, Pág.6166-6179, 2007.

- [2] A. Asuncion, D.J. Newman. UCI Machine Learning Repository. <http://www.ics.uci.edu/mlearn/MLRepository.html>, Irvine, CA: University of California, 2007.
- [3] Y. Benjamini, Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B*, 57, Pág.289-300, 1995.
- [4] L. Breiman. Bagging predictors. *Machine Learning*, 24(2), Pág.123-140, 1996.
- [5] L. Breiman. Random forests. *Machine Learning*, 45(1), Pág.5-32, 2001.
- [6] J.M. Cadenas, M.C. Garrido, R. Martínez. Una estrategia de particionamiento fuzzy basada en combinación de algoritmos. *Proceedings XIII Conferencia de la Asociación Española para la Inteligencia Artificial*, Pág. 379-388, 2009.
- [7] T.G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine Learning*, 40(2), Pág.139-157, 2000.
- [8] Y. Freund, R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, 55(1), Pág.119-139, 1997.
- [9] S. García, A. Fernández, J. Luengo, F. Herrera. A study statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. *Soft Computing* doi:10.1007/s00500-0080392-y.
- [10] I.H. Witten, E. Frank. Data Mining: Practical machine learning tools and techniques. Segunda Edición, Morgan Kaufmann, 2005.
- [11] J. Jang. Structure determination in fuzzy modeling: A Fuzzy CART approach. *Proceedings IEEE Conference on Fuzzy Systems*, Pág.480-485, 1994.
- [12] C.Z. Janikov. Fuzzy Decision trees: Issues and Methods. *IEEE transactions on Systems, Man and Cybernetics*, vol. 28, Pág.1-14, 1998.
- [13] L. Koen-Myung, L. Jee-Hyong, L. Kyung-Mi, L. Hyung. A Fuzzy Decision Tree Induction Method for Fuzzy Data. *Proceedings IEEE Conference on Fuzzy Systems*, Pág.22-25, 1999.
- [14] R.E. Schapire. Theoretical views of boosting. *Proc. 4th Eur. Conf. Computational Learning Theory*, Pág.1-10, 1999.

MEJORAS EN EL USO DE APRENDIZAJE ACTIVO CON ÁRBOLES FUZZY: UN EJEMPLO DE SU APLICACIÓN EN LA TOMA DE DECISIONES DE UN SISTEMA COORDINADO

Giovanni Acampora¹ Giuseppe Fenza¹ Enrique Muñoz² Bernardino Romera³

¹ Università degli Studi di Salerno, {gacampora,gfenza}@unisa.it

² Universidad de Murcia, enriquemuba@um.es

³ University College London, bernard24@gmail.com

Resumen

El aprendizaje computacional se ha utilizado frecuentemente para ayudar en la toma de decisiones y al desarrollo de controladores, sin embargo, en ciertas ocasiones su elevado coste impide su uso. El aprendizaje activo trata de reducir este coste seleccionando aquellos ejemplos que aportan más información. Sin embargo cuando se aplica a los modelos basados en árboles de decisión los resultados no suelen ser competitivos debido a su inherente inestabilidad. En este artículo se propone una técnica que pretende paliar estos problemas consistente en combinar el aprendizaje semisupervisado con el descarte selectivo de aquellos ejemplos que una vez aprendidos empeoran el comportamiento del modelo. La eficacia de esta técnica se ha comprobado con distintas bases de datos UCI y con un problema real: el diseño del sistema de toma de decisiones en un sistema cooperativo para resolver problemas de optimización.

Palabras Clave: Aprendizaje activo, Toma decisiones, Árboles de decisión fuzzy, Sistema coordinado.

1 INTRODUCCIÓN

Es frecuente el uso de técnicas de aprendizaje computacional para ayudar en la toma de decisiones o en el desarrollo de controladores. Pero en algunos casos la aplicación de este tipo de técnicas resulta complicada debido al alto coste que supone. Una de las razones por las que puede ser costoso es el etiquetado de los ejemplos que componen la base de datos de aprendizaje, entendiendo como etiquetado la asignación de la

clase o el valor a inferir para un determinado ejemplo. Algunos casos concretos en los que se da este problema son aquellos en los que el etiquetado requiere la supervisión humana o aquellos que requieren la ejecución de algún sistema software que consuma muchos recursos.

El aprendizaje activo surge como respuesta a este tipo de problemas. La idea que utiliza es otorgar al modelo de aprendizaje la libertad de elegir qué ejemplo o ejemplos quiere aprender a continuación. Esto contrasta con el paradigma tradicional o pasivo en el que los ejemplos de entrenamiento son aprendidos de forma aleatoria. Empleando aprendizaje activo, se aprovecha el grado creciente de inteligencia adquirida por el modelo para seleccionar en cada momento aquel ejemplo o ejemplos de la base de datos, que pueden proporcionarle un mayor grado de información.

Dentro del aprendizaje activo el uso de árboles de decisión como modelo base no está muy extendido, debido a los problemas de inestabilidad [4] que muestran. En este artículo se propone una técnica de aprendizaje activo que, usando árboles de decisión fuzzy como técnica base, pretende paliar los problemas antes mencionados. Además esta técnica será aplicada al problema concreto del modelado del proceso de toma de decisiones del coordinador de un sistema cooperativo para así probar su efectividad. Para ello el artículo será estructurado de la siguiente forma, en la Sección 2 se hará una pequeña introducción al aprendizaje activo, en la Sección 3 se presentará la técnica propuesta, comprobándose su eficacia en la Sección 4. Posteriormente, en la Sección 5, se presentará un problema real, el modelado de la toma de decisiones de un sistema cooperativo de metaheurísticas, probándose la validez de la técnica propuesta en la Sección 6. Finalmente en la sección 7 se presentan las conclusiones del trabajo.

2 APERNDIZAJE ACTIVO

En el aprendizaje activo hacemos que sea la propia técnica de aprendizaje la encargada de seleccionar un

número limitado de ejemplos para pedir su etiqueta de modo que maximice su precisión en cada nueva iteración. Una de las primeras técnicas en aparecer [3] consistía en elegir ejemplos de la región de incertidumbre de modo que en cada iteración esta región fuera decreciendo. Sin embargo este enfoque resulta demasiado costoso, por lo que se han propuesto otros enfoques donde la región de incertidumbre no se calcula de forma explícita, como en Uncertainty Sampling [7], que consiste en elegir el ejemplo para el cuál existe una mayor incertidumbre. O Query By Committee [5], que consiste en mantener un comité de clasificadores para inferir la etiqueta de cada ejemplo candidato. Finalmente se elige aquel cuyo grado de desacuerdo entre el comité es mayor.

En el contexto de aprendizaje activo usando árboles de decisión, se han realizado algunos proyectos tales como [9]. Sin embargo el uso de este tipo de modelo no es muy común dentro del paradigma activo. De acuerdo a lo que se comenta en [4], las técnicas de construcción de árboles son inestables en el sentido de que se pueden construir árboles completamente distintos a partir de un conjunto de entrenamiento prácticamente igual. Esto se debe a la naturaleza recursiva del algoritmo de construcción del árbol, que tiene en cuenta todos los ejemplos de entrenamiento en cada momento.

Esta inestabilidad resulta especialmente importante si se usan árboles como modelos para realizar aprendizaje activo. En efecto, esto es así ya que el algoritmo de aprendizaje activo adquiere un ejemplo (o un conjunto limitado de ellos) en cada iteración basándose en el modelo aprendido hasta ese momento. Sabemos que un nuevo ejemplo puede hacer cambios bruscos en los árboles construidos haciendo que varíe su precisión de predicción, ya sea para bien o para mal. Cuando se da este último caso, los modelos se basan en mala información para poder elegir el siguiente ejemplo, con lo cuál se puede iniciar un ciclo que conlleva a que los malos resultados puedan fácilmente superar a los progresos hechos cuando los cambios bruscos de los árboles son positivos. La técnica propuesta en este artículo pretende paliar estos problemas, ayudando en la utilización de árboles de decisión en el aprendizaje activo, además el uso de árboles de decisión fuzzy, debido a su mayor robustez, también ayudará.

3 COMBINANDO APRENDIZAJE SEMISUPERVISADO Y ARREPENTIMIENTO

3.1 APRENDIZAJE SEMISUPERVISADO

La técnica del aprendizaje semisupervisado consiste en añadir a la base de datos de aprendizaje aquellos ejem-

plos, que aun no estando etiquetados se tiene una gran certeza acerca de la etiqueta que deberían tener, en este caso se utiliza la etiqueta esperada como si realmente se hubiera averiguado la misma.

En un esquema de aprendizaje activo típico, se eligen de entre todos los ejemplos disponibles, n ejemplos para ser evaluados por la técnica de aprendizaje. A continuación, se aplica, para cada uno, un criterio que mide la incertidumbre que se tiene sobre los resultados obtenidos. En función de este criterio, se seleccionará de los n ejemplos aquella con un mayor índice de incertidumbre, x_1 .

Una vez que se ha elegido el ejemplo x_1 , éste es inferido y seguidamente se dispone de su etiqueta correspondiente y_1 , de modo que se añade $\langle x_1, y_1 \rangle$ al conjunto de entrenamiento. Además se puede aprovechar la información aportada por la etiqueta para compararla con la predicción realizada por la técnica de aprendizaje. De esta manera se puede descubrir si se estaba realizando una predicción acertada o no. En caso de que la predicción fuera acertada, significa que, aun teniendo en cuenta que x_1 era el ejemplo con una mayor incertidumbre, el resultado podría haber sido predicho. Es en este momento donde se aplica la modalidad de aprendizaje semisupervisado, la cuál supone que, de la misma forma que el ejemplo con mayor incertidumbre pudo ser etiquetado correctamente, los ejemplos con menor incertidumbre también podrán ser etiquetados correctamente. De este modo, a estos ejemplos $x_2, x_3, \dots, x_n$, le son asignados su correspondiente etiqueta en función de los resultados predichos $y'_2, y'_3, \dots, y'_n$ y así los pares $\langle x_2, y'_2 \rangle, \langle x_3, y'_3 \rangle, \dots, \langle x_n, y'_n \rangle$ son añadidos al conjunto de entrenamiento.

3.2 CORRIGIENDO LAS DECISIONES ERRÓNEAS

Cuando un ente inteligente tiene cierto conocimiento sobre un tema y recibe nueva información sobre el mismo, el primer proceso que realiza es asimilar esta nueva información e intentar integrarla con el conocimiento previo. Si este ente se ve incapaz de encajar la nueva información con la base que tenía previamente, dudará acerca de la veracidad de la última información recibida. La idea clave tras la técnica propuesta se basa en esta idea: comprobar que el nuevo ejemplo no sólo aporta nueva información al sistema sino que refuerza lo que se había adquirido anteriormente. En caso contrario se duda de él y se descarta. A este enfoque le llamaremos aprendizaje con arrepentimiento.

A grandes rasgos, el aprendizaje con arrepentimiento se refiere a la modificación del esquema activo de modo que una vez elegido y etiquetado el nuevo ejemplo, el

modelo comprueba si la incorporación de ese nuevo ejemplo a su base de entrenamiento va a proporcionar una mejor o, por contra, una peor capacidad de predicción ante la inferencia de nuevos ejemplos.

Para dilucidar si el nuevo ejemplo compensa ser aprendido o no, lo ideal sería realizar una validación tomando como conjunto de prueba todos los ejemplos de la base de datos: en primer lugar se realiza un test utilizando el modelo actual y a continuación se realiza la misma prueba pero añadiendo al conjunto de entrenamiento el ejemplo recién etiquetado. Finalmente se comparan los resultados y se añade este último ejemplo en caso favorable. Obviamente esto no se puede hacer debido a que la mayoría de los ejemplos están sin etiquetar. De hecho los ejemplos que están etiquetados son los que ha aprendido el modelo. Por esto se debe hacer una estimación de esta medida. En concreto la implementación de esto ha sido realizar lo explicado anteriormente pero tomando el conjunto de entrenamiento como conjunto de prueba. De esta forma:

$$ErrorAnterior = \frac{Error(M_T(T))}{|T|}$$

$$ErrorActual = \frac{Error(M_{T \cup \{n\}}(T \cup \{n\}))}{|T|+1}$$

donde

- T es el conjunto de ejemplos que forman el conjunto de entrenamiento.
- n es el nuevo ejemplo recién etiquetado.
- $Error(M_E(T))$ es el error cometido al inferir los ejemplos del conjunto T a partir de un modelo entrenado con los ejemplos del conjunto E .

Además se introduce el concepto de margen de confianza como una variable de control que permite al algoritmo adaptarse a cualquier posible evolución del proceso de aprendizaje, de modo que nunca se rechazan o aceptan ejemplos en exceso.

De esta forma, una vez que se ha calculado $ErrorAnterior$ y $ErrorActual$ se comprueba si ($ErrorActual > ErrorAnterior * MargenConfianza$) en caso afirmativo se rechaza el nuevo ejemplo x_1 y se incrementa el margen de confianza, en caso negativo se decrementa el margen de confianza.

Podría pensarse que este enfoque desaprovecha los recursos ya que éstos pueden consumirse para etiquetar un ejemplo que posteriormente no será utilizado. Sin embargo, su uso ha propiciado mejores resultados a los obtenidos anteriormente.

La técnica propuesta consiste en una mezcla del aprendizaje semisupervisado con el aprendizaje con arrepentimiento. En este artículo se probará esta solución sobre un esquema basado en comité, a pesar de ello este

enfoque es extrapolable a otros esquemas de aprendizaje activo.

4 TEST DE LA APROXIMACIÓN

En este apartado se probará la eficiencia de la aproximación propuesta, comparándola con el aprendizaje activo tradicional y el aprendizaje pasivo. Las pruebas se realizarán utilizando como modelo de aprendizaje básico un árbol de decisión fuzzy. Como base de datos de pruebas se usarán distintos problemas de aprendizaje obtenidos de la base de datos UCI [1]. Estas bases de datos serán: Breast Cancer Wisconsin (BCW), German Credit (Ger), Ionosphere (Iono), Pima Indians Diabetes (Pima) y Image Segmentation (Seg). Todas estas bases de datos tienen en común el alto número de ejemplos que las componen, y por tanto podría ser conveniente aplicarles aprendizaje activo.

En la Tabla 1 se muestran los resultados obtenidos: en la segunda columna se muestra el número de ejemplos que componen la base de datos, en la siguiente cual es el porcentaje de acierto obtenido por el aprendizaje pasivo después de haber aplicado sobre la base de datos completa usando validación cruzada de 10 conjuntos. En las siguientes se muestra el número de ejemplos que fue necesario utilizar por el aprendizaje activo (act.) y el aprendizaje activo semisupervisado con arrepentimiento (act. mod.) para obtener un porcentaje de acierto igual o mayor.

Tabla 1: Comparativa aprendizajes.

BD	n. inst.	pas.	act.	act. mod.
BCW	699	72.3%	218	78
Ger	1000	95.07%	272	77
Iono	351	91.8%	176	91
Pima	768	66.42%	329	78
Seg	2310	81.13%	416	139

Como puede observarse, el uso de cualquier técnica de aprendizaje activo supone una mejora sustancial en el número de ejemplos que se deben usar para alcanzar el porcentaje de acierto que alcanza el aprendizaje activo. De igual manera se puede ver como el uso de la técnica propuesta (act. mod.) también proporciona una mejora con respecto al aprendizaje activo clásico.

5 APLICACIÓN AL DISEÑO DE UN SISTEMA INTELIGENTE

En esta sección pretendemos mostrar la validez del esquema propuesto para un problema real, como es el modelado de la toma de decisiones del coordinador de un sistema cooperativo de metaheurísticas para resolver problemas de optimización.

5.1 UN SISTEMA HÍBRIDO DE METAHEURÍSTICAS

El sistema de metaheurísticas que centrará nuestra atención [2] está basado en un sistema multiagente, en el cual cada agente implementa una metaheurística y coopera con el resto de agentes para resolver una instancia del problema de optimización. Para controlar la cooperación entre los distintos agentes se hace uso de un agente coordinador, que tiene dos tareas principales: recolectar información acerca del rendimiento de cada agente y basándose en esta decidir como se debe modificar el comportamiento de cada agente de tal manera que se mejore el comportamiento global. La Figura 1 muestra una imagen conceptual del sistema.

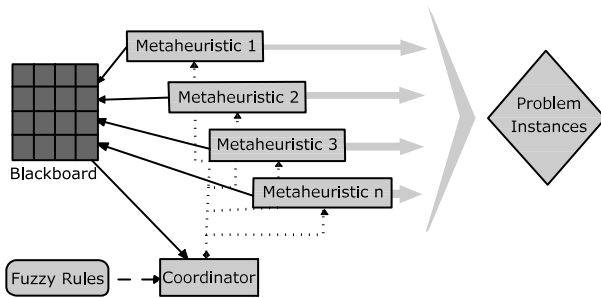


Figura 1: El sistema cooperativo de metaheurísticas.

Para que el coordinador pueda realizar las tareas antes mencionadas se le debe dotar de cierta “inteligencia”, para hacer esto se realiza un proceso de aprendizaje supervisado con el objetivo de completar un conjunto de reglas fuzzy que controla el comportamiento del coordinador. Este conjunto de reglas se compone de la siguiente regla replicada una vez por cada metaheurística:

- SI $[(wm_1 * d_1 \text{ O } \dots \text{ O } wm_n * d_n) \text{ ES suficiente}]$ ENTONCES cambiar la solución actual de *MH*.

donde:

- n el el número de metaheurísticas.
- *MH* es la metaheurística evaluada por la regla.
- $d_i = (r_i - perf_{MH}) / maximum(r_i, r_{MH})$, donde r es una medida de rendimiento, que en nuestro caso será el valor de la función objetivo obtenido por la solución actual de cada metaheurística.
- $wm_i \in [0, 1]$ donde $\sum_{i=1}^n wm_i = 1$ y wm_i representa el peso de la metaheurística i (importancia de esta para resolver la instancia actual).
- *suficiente* es un conjunto fuzzy con función de pertenencia trapezoidal con soporte contenido en

$[0, 1]$ definido por la cuadrupla (a, b, c, d) . Su representación matemática es presentada en la ecuación (1):

$$\mu(x, a, b, c, d) = \begin{cases} 0 & x \leq a \text{ or } x \geq d \\ (x - a)/(b - a) & x \in (a, b] \\ (d - x)/(d - c) & x \in [c, d) \\ 1 & x \in [b, c] \end{cases} \quad (1)$$

Esta regla cambia la posición en el espacio de búsqueda de la metaheurística que está mostrando un rendimiento malo por una posición cercana a la de otra metaheurística que está mostrando un comportamiento mejor.

Además de la anterior regla, el coordinador también dispone de un conjunto de reglas de inicialización de parámetros que deciden para cada metaheurística que valores de parámetros deben ser tomados dependiendo de la instancia del problema que se vaya a resolver.

Tanto los wm_i que controlan la regla como los valores de los parámetros de cada metaheurística que deberán tomarse son obtenidos del proceso de aprendizaje supervisado antes mencionado.

5.2 ADAPTACIÓN DEL SISTEMA

Para obtener los pesos, wm_i y los valores de los parámetros que serán usados por las reglas definidas anteriormente, se hace uso de un conjunto de árboles fuzzy obtenidos de la aplicación de un proceso de aprendizaje supervisado. Los modelos necesarios serían los siguientes: un árbol fuzzy para predecir los pesos de todas las metaheurísticas, y un árbol fuzzy para cada parámetro de cada metaheurística que predice el valor que obtendrá el mejor rendimiento.

5.3 ALGUNOS CONFLICTOS

Para obtener estos modelos anteriormente se usaba un proceso de aprendizaje supervisado compuesto de dos fases, preparación de los datos y minería de datos. En la primera fase las diferentes metaheurísticas que componen el sistema son ejecutadas varias veces usando diferentes valores de los parámetros con el propósito de obtener una base de datos que resuma el comportamiento de cada metaheurística. En la segunda fase se aprenden los diferentes árboles fuzzy antes mencionados. Este proceso tiene un importante cuello de botella en la fase de preparación de los datos, puesto que la ejecución de las distintas metaheurísticas tiene un costo muy elevado, por ello es interesante reducir el número de ejemplos que se deben de usar para conseguir los modelos, lo cual se espera conseguir utilizando aprendizaje activo.

5.4 APLICANDO APRENDIZAJE ACTIVO

Al aplicar aprendizaje activo al sistema nos encontramos con dos situaciones distintas, el aprendizaje de los pesos y el de los valores de los parámetros. Para resolver ambos casos se ha utilizado el esquema antes descrito, pero con ciertas peculiaridades.

En el caso de los pesos, al tratarse de una mezcla entre regresión y ranking se debe aplicar un esquema similar al propuesto en [9], en el cual para cada ejemplo de la base de datos se calcula la varianza de los pesos obtenidos por cada miembro del comité, siendo preferida aquella con una mayor varianza.

El caso de los parámetros es un problema de clasificación con la peculiaridad de que sea multiclase, en cuyo caso es interesante usar la aproximación propuesta en [6] que consiste en elegir aquel ejemplo que minimiza la diferencia entre los votos a la clase más popular y la segunda más popular.

6 PRUEBAS SOBRE UN PROBLEMA REAL

En este apartado se realizarán pruebas para comprobar la efectividad de la aproximación propuesta para el problema del modelado del coordinador de un sistema de metaheurísticas.

6.1 EL PROBLEMA DE OPTIMIZACIÓN

El sistema de metaheurísticas que se construirá tendrá como objetivo la resolución de instancias del problema de la mochila, que se puede definir formalmente de la siguiente forma:

$$\begin{aligned} & \text{Max} \quad \sum_{i=1}^n p_i \times x_i \\ & \text{s.a} \quad \sum_{i=1}^n w_i \times x_i \leq C \\ & \quad x_i \in \{0, 1\}, i = 1, \dots, n \end{aligned}$$

donde

- n es el número de items.
- x_i indica si el item i está incluido en la mochila.
- p_i es el beneficio asociado al item i .
- $w_i \in [0, \dots, r]$ es el peso del item i .
- C es la capacidad de la mochila.

Se asume que $w_i < C$, $\forall i$ y $\sum_{i=1}^n w_i > C$.

Este problema es NP-completo y ha sido ampliamente estudiado. Para realizar las pruebas se han utilizado

distintos tipos de instancias generadas siguiendo el método propuesto en [8] y que han demostrado ser difíciles para los mejores algoritmos exactos. Las instancias generadas tendrán un tamaño de 500, 1000, 1500 o 2000 objetos y son de los siguientes tipos:

- **Spanner:** Estas instancias se han construido de forma que sus items son múltiplos de un conjunto pequeño de items llamados key. Esos key fueron generados usando tres distribuciones: Uncorrelated, Weakly correlated, Strongly correlated.
- **Profit ceiling:** En estas instancias todos los beneficios son múltiplos de un parámetro dado d .
- **Circle:** Estas instancias se generan de forma que los beneficios son una función de los pesos teniendo una representación elíptica.

6.2 EL SISTEMA IMPLEMENTADO

Para resolver este problema se ha desarrollado un sistema cooperativo compuesto de tres metaheurísticas, un algoritmo genético, una búsqueda tabú y un temple simulado. Además el conjunto fuzzy *suficiente* se define con los siguientes puntos $[0, 0.1, 1, 1]$ y el α -corte usado es 0.05. Para el aprendizaje previo se usaron 500 instancias de las cuales el aprendizaje pasivo usa todas y la técnica propuesta 20.

6.3 METODOLOGÍA

Para realizar las pruebas se resolvieron 100 instancias distintas, ejecutándose cada estrategia durante 10 segundos parando cada 250 milisegundos para realizar la coordinación. Cada instancia fue resuelta 10 veces, y los resultados muestran los promedios. Los tests se desarrollaron en AMD Opteron Dual Core con 8 GB RAM pertenecientes al cluster UGRGrid¹.

6.4 RESULTADOS

La Tabla 2 muestra los resultados obtenidos por cada estrategia, agrupados por tipo de instancia y tamaño. Como se ha comentado anteriormente los modelos obtenidos por el aprendizaje activo se construyeron usando 20 instancias, mientras que los obtenidos usando aprendizaje pasivo utilizaron 500 instancias. Como se puede observar los resultados obtenidos son muy similares, no pudiéndose observar diferencias significativas con el test estadístico no paramétrico de wilcoxon.

Estos resultados son los que esperábamos. Las técnicas de aprendizaje activo obtienen modelos similares a los obtenidos por el aprendizaje pasivo, ya que obtienen los valores de error muy similares (a veces mejores), pero se debe notar el tiempo usado para generar los

¹<http://ugrgrid.ugr.es/>

Tabla 2: Comparativa aprendizajes.

tipo	tam.	pasivo	semisuperv. arr.
unc span	500	0,01% (0,05%)	0,00% (0,00%)
	1000	0,02% (0,07%)	0,02% (0,09%)
	1500	0,10% (0,22%)	0,08% (0,15%)
	2000	0,08% (0,22%)	0,16% (0,54%)
wea span	500	0,68% (0,71%)	0,69% (0,67%)
	1000	1,22% (1,29%)	1,20% (1,26%)
	1500	1,29% (1,35%)	1,29% (1,33%)
	2000	1,47% (1,44%)	1,49% (1,47%)
str span	500	1,24% (1,35%)	1,27% (1,33%)
	1000	1,48% (2,36%)	1,45% (2,16%)
	1500	1,69% (2,36%)	1,64% (2,31%)
	2000	2,20% (3,07%)	2,20% (3,02%)
pceil	500	0,20% (0,23%)	0,21% (0,26%)
	1000	0,32% (0,45%)	0,31% (0,42%)
	1500	0,52% (0,58%)	0,51% (0,57%)
	2000	0,37% (0,52%)	0,35% (0,49%)
circle	500	3,24% (2,36%)	3,28% (2,35%)
	1000	5,83% (3,46%)	5,62% (3,32%)
	1500	8,62% (5,26%)	8,66% (5,37%)
	2000	11,06% (6,64%)	10,77% (6,42%)

modelos que controlan la ejecución de los sistemas. Como se ha comentado anteriormente para clasificar una instancia se deben ejecutar muchas veces todas las metaheurísticas utilizando diferentes valores para sus parámetros, en este caso se ha ejecutado cada instancia durante 5 segundos con 3 metaheurísticas distintas, usando 8 combinaciones diferentes de parámetros y repitiendo cada ejecución 10 veces, esto significa que para cada instancia aprendida se necesitan 1200 segundos, 20 minutos. Por tanto para 20 instancias se han necesitado 400 minutos, aproximadamente 7 horas, mientras que para 500 instancias ha sido necesario aproximadamente 7 días. Se puede ver claramente que, aunque los resultados sean similares es preferible utilizar el aprendizaje activo debido a la reducción de tiempo que supone.

7 CONCLUSIONES

En este trabajo se presenta una estrategia de aprendizaje activo desarrollada para paliar los problemas que presenta cuando se aplica a árboles de decisión. Esta estrategia se basa en la combinación de dos técnicas, aprendizaje semisupervisado que trata de etiquetar aquellos ejemplos para los que se tiene una menor incertidumbre con la etiqueta que propone el modelo de aprendizaje, y aprendizaje con arrepentimiento que se basa en ignorar aquellos ejemplos que han reportado un empeoramiento en el rendimiento del modelo. Los resultados obtenidos sobre determinadas bases de datos del repositorio UCI arrojan resultados prometedores para esta técnica. Asimismo, también se ha

probado la validez de la aproximación con un problema real, el modelado de la toma de decisiones de un coordinador de un sistema metaheurístico cooperativo. Para este caso se han realizado pruebas comparando los resultados que obtiene el sistema cooperativo cuando se ha entrenado usando aprendizaje pasivo con 500 ejemplos y el esquema propuesto con 20 ejemplos. Los resultados obtenidos muestran que estadísticamente ambos sistemas son equivalentes, sin embargo la reducción en tiempo obtenida (de 7 días a 7 horas) hace que la utilización del esquema propuesto sea preferible a la utilización del aprendizaje pasivo.

Agradecimientos

Los autores tercero y cuarto agradecen la financiación proporcionada por el proyecto TIN2008-06872-C04-03 del MICINN, y por el proyecto 04552/GERM/06 y al programa de FPI de la “Agencia de Ciencia y Tecnología” de la Región de Murcia, de España.

Referencias

- [1] A. Asuncion, D.J. Newman. UCI Machine Learning Repository [http://www.ics.uci.edu/~mllearn/MLRepository.html] 2007.
- [2] J.M. Cadenas, M.C. Garrido, E. Muñoz. Using machine learning in a cooperative hybrid parallel strategy of metaheuristics. *Information Sciences* 179: 3255–3267, 2009.
- [3] D. Cohn, L. Atlas, R. Ladner. Improving Generalization with Active Learning. *Machine Learning* 15(2): 201–221, 1994.
- [4] K. Dwyer, R. Holte. Decision Tree Instability and Active Learning. In *Proc. 18th European Conference on Machine Learning*, 4701: 128–139, 2007.
- [5] Y. Freund, H. S. Seung, E. Shamir, N. Tishby. Selective Sampling Using the Query by Committee Algorithm. *Machine Learning* 28: 133–168, 1997.
- [6] J. Huang, S. Ertekin, Y. Song, H. Zha, C. L. Giles. Efficient Multiclass Boosting Classification with Active Learning. In *Proc. 7th SIAM Conference*, 297–308, 2007.
- [7] D.D. Lewis, W.A. Gale. A sequential algorithm for training text classifiers. In *Proc. 17th international ACM SIGIR conference*, 3–12, 1994.
- [8] D. Pisinger. Where are the hard knapsack problems?. *Computer and Operations Research*, 32(9): 2271–2284, 2005.
- [9] M. Saar-Tsechansky, F. Provost. Active Sampling for Class Probability Estimation. *Machine Learning* 54(2): 153–178, 2004.

ÍNDICE DE AUTORES

G. Acampora.....	643	M. Calderón	97
J.S. Aguilar-Ruiz	459, 465	C. Calvo	631
J. Albusac	261, 375	A. Cano.....	229
R. Alcalá.....	73, 249	C. Cano Gutiérrez	453
J. Alcalá Fernández.....	249, 417	G. Carbone	43
C. Alcalde	151	C.J. Carmona.....	441
J.R. Alique	381	J.V. Carrillo.....	411, 631
J. Alonso	357	M. Casanovas	285, 567
J.M. Alonso.....	7, 399	J. Casanovas	309
S. Alonso	575	R. Castillo-Ortega	267
A. Álvarez	7	J.L. Castro	217
F. Andrés	25	J.J. Castro-Sánchez	261, 375
E. Anguiano Rey	465	M. Cerami.....	145
L. Argüelles	13	J. Chamorro-Martínez.....	337
R. Arjona	393	P. Cordero.....	175,205
J. Aroba	103	O. Cordón.....	501
G. Asencio Cortés.....	459	C. Cornelis.....	555
O. Baños.....	435	M. Correa	381
E. Barrenechea	97, 325	I. Couso	495
A. Barriga	1	S. Damas	501
M. Batet	561	E. de Amo.....	139
I. Baturone	393	B. De Baets	279
G. Beliakov.....	315, 331	J. de la Mata	525
I.J. Blanco.....	223	C. del Val	471, 477
A. Blanco Morón.....	453	J. Derrac	417
D. Boixader	127	S. Díaz.....	513
P. Bonissone	637	F. Díaz-Hermida	255
J. Bragard	133	R.A. Díaz-Valladares	637
J. Brito	625	J.M. Doña	607
P. Brox	393	J. Elorza	133
A. Bugarín	255	M. Enciso	205
P. Burillo.....	133	M. Espinilla	567, 601
A. Burusco.....	151	F. Esteva	145, 187
H. Bustince.....	97, 115, 279, 315, 331	E. Falcó Díaz de Cerio	581
I.P. Cabrera	175	G. Fenza	643
F.J. Cabrerizo.....	575	A. Fernández	97, 115, 417
J.M. Cadenas	411, 631, 637	J. Fernández	279, 315
S. Cala Cordero.....	423	S. Fernández	363

G. Fernández-Escribano	387	L.J. Herrera.....	447
J. Fernández-Sánchez	139	V. Herrera Tirado	375
E. Fernández-Viñas.....	525, 531	E. Herrera-Viedma.....	549, 555, 575
M. Ferreira-Satler	537, 543	O. Ibáñez.....	501
V.M. Flores	381	C. Ivorra	631
R. Fuentes-González.....	133, 151	J. Jacas.....	67
M.J. Gacto.....	73	L. Jiménez-Linares	25, 387
A. Gajate	429	P. Julián-Iranzo.....	211, 543
M. Galar	331	A. Jurio	325
M. Galende	79, 91	L.M. Laita	169
J. Galindo	199, 405	M.T. Lamata.....	619
J. Gallardo-Casero	211, 543	E. Lerma.....	343
A.J. García	607	T. León.....	31
S. García	417	E. León González	405
F. García Alcalde	453	V. Liern	31, 631
M.C. García-Alegre.....	349	B. Llamazares	587
M.S. García-Cascales.....	619	P.L. López	441
I. García-Honrado	163	V. López	217
J.L. García Lapresta	581, 587	V. López	303, 321
L. Garmendia	321	M.A. López-Carmona	363
M.C. Garrido	411, 631, 637	F.J. López Domingo	453
P. Garrido.....	61	D.A. López García	43
R. Ghinea.....	447	C. López-Molina	325
K. Gibert.....	561	I. López Quintero	43
A.M. Gil-Lafuente	291	J. Luengo.....	235, 417
L. Godo	187	J.M. Luna.....	241
D. Gómez	303	N. Madrid.....	157
F. Gómez Bravo	43	L. Magdalena	399
R. González del Campo.....	321	N. Marín	267
C. González Morcillo.....	375	A. Márquez.....	85
S. Guadarrama.....	49, 163	F.A. Márquez	85
D. Guinea.....	349	A.E. Márquez Chamorro.....	465
D.M. Guinea.....	349	D. Martín	349
G. Gutiérrez.....	175	J. Martín.....	297
C. Gutiérrez-Soto	193	P. Martín	121
R. Haber	429	J.M. Martín Ramos	43
J. Hernández.....	181	D. Martinetti.....	181, 513
N. Hernández	7	F.J. Martínez	61
A. Hernando.....	169	F.J. Martínez	625
F. Herranz	7	J. Martínez.....	175
F. Herrera.....	73, 97, 115, 235, 249, 417, 593	L. Martínez.....	567, 593, 601

M. Martínez-Ballesteros	109, 471	M.M. Pérez	447
C. Martínez-Cruz	223	P. Pérez Asurmendi	587
P. Martínez-Jiménez	337	I. Pérez de Guzmán	205
G. Mayor	297	M.D. Pérez-Godoy	441
B. Melián-Batista	37	J. Pijuan	369
J. Méndez	193	I. Plaza	61
V.H. Menéndez	537	A. Pradera	273
J.M. Merigó	285, 291, 567, 613	M.E. Prieto	537
R. Mesiar	279	M. Polo	19
V. Milanés	357	H. Pomares	435, 447
A. Monreal	67	C. Porcel	549
J. Montero	169, 279, 303	C. Puente	519
I. Montes	181, 513	R. Pulgar	447
S. Montes	181, 513	J. Recasens	67, 127
E. Montseny	343	M.I. Rey	91
A. Mora	205	J.V. Riera	309
J.A. Moreno	625	V.M. Rivas	109
J. Moreno-García	25, 261, 387	A.J. Rivera	441
J.M. Moreno-Vega	37	E. Roanes-Lozano	169
F.J. Moreno-Velo	121, 423	R.M. Rodríguez	593
A. Moreo	217	R.O. Rodríguez	187
A. Muñoz	637	L. Rodríguez Benítez	25, 261, 387
E. Muñoz	411, 643	I. Rojas	435, 447
Nashaat M.H.	1	S. Romaní	343
M. Ocaña	7	B. Romera	643
M. Ojeda-Aciego	157, 175	F.P. Romero	19, 537, 543
J.A. Olivas	19, 519, 525, 531	J.R. Romero	241
J.L. Olmo	241	R. Romero-Zaliz	471, 477
E. Onieva	357	C. Rubio-Manzano	211
J. Otero	507	J. Ruiz-Morilla	531
M. Pagola	325	G.I. Sainz	79, 91
A.M. Palacios	489	D. Sánchez	267
A. Pasuello	369	L. Sánchez	417, 483
D. Paternain	315, 331	J. Santamaría	501
J.I. Peláez	607	J. Santana	447
D.A. Pelta	357	J.A. Sanz	115
A. Peralta	19	M. Schuhmacher	369
A. Peregrín	85, 103	A. Sepúlveda	199
D. Pérez	49	J. Serrano-Guerrero	525, 531
I. J. Pérez	575	P. Sobrevilla	343
J. Pérez	357	A. Sobrino	519

C.J. Solana-Ciprés.....	261, 387
A. Soto	525
J.M. Soto-Hidalgo	337
J. Tinguaro-Rodríguez	303
R. Toscano	103
J. Tramullas	61
E. Trillas	163
G. Triviño	7, 55
A. Urrutia.....	193, 199
A. Valls.....	369, 561
A. van der Heide	55
P. Vega	429
F.J. Vela.....	61
J.R. Velasco	363
S. Ventura	229, 241
J.L. Verdegay	37, 625
T. Vetterlein.....	187
P. Victor.....	555
B. Victoriano	303
M.A. Vila	223
E. Villanueva	349
A. Zafra	229
A. Zapata	537
J.M. Zurita	217
I. Zwir	471, 477