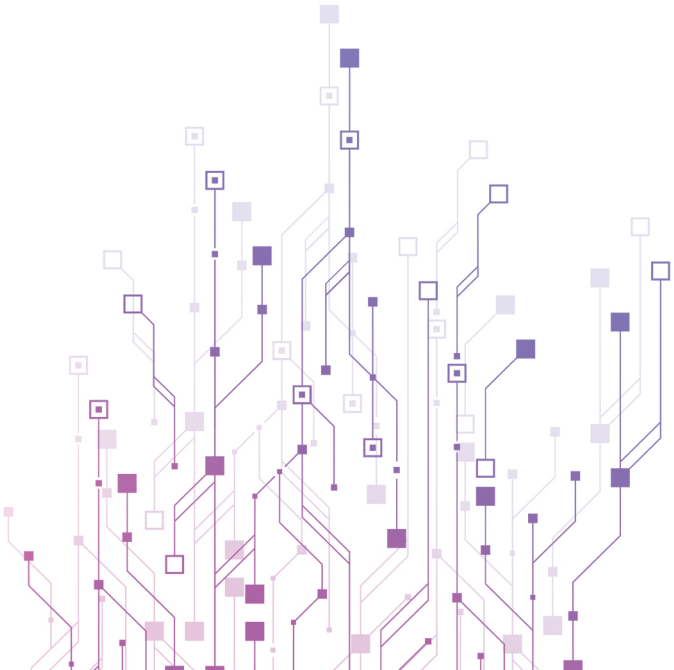
XXIV Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)



XXIV Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)

Sesión general





Integración de LLMs y Computación con Palabras para la Evaluación del Desempeño 360 grados con Información Heterogénea

F. J. Martínez[1 [0000−0001−7969−1737]], Ángela Escobar[1 [0009−0007−8879−2450]], and L. Martínez[1 [0000−0003−4245−8813]]

Universidad de Jaén, Jaén, Spain
fjmimbre@ujaen.es, aem00007@red.ujaen.es, martin@ujaen.es

Resumen La evaluación del desempeño 360 grados integra valoraciones de múltiples colectivos (supervisores, colaboradores, clientes y el propio empleado) sobre un conjunto de criterios. Los marcos basados en computación con palabras (CWW) y modelos como la 2-tupla lingüística gestionan eficazmente la incertidumbre cualitativa de este proceso, pero requieren formularios estructurados que generan fatiga y baja participación. En este trabajo proponemos integrar Modelos de Lenguaje de Gran Escala (LLMs) como evaluadores lingüísticos automáticos capaces de derivar valoraciones por criterio a partir de texto libre. La contribución central no reside en emplear el LLM como clasificador discreto, sino en interpretar su distribución de probabilidad (restringida y renormalizada sobre los términos de la escala lingüística) como una estructura de creencia compatible con el modelo 2-tupla. La traslación simbólica α emerge así directamente de la estructura de creencia, sin heurísticas externas. La representación resultante es integrable en procesos de agregación basados en operadores de 2-tupla lingüística, los cuales fueron empleados con éxito anteriormente en procesos de evaluación del desempeño 360 grados. Se realiza una prueba de concepto sobre un caso de estudio industrial real del sector textil, empleando textos sintéticos alineados con valoraciones humanas de referencia. Los resultados son consistentes con la viabilidad del marco: el método retiene información ordinal que un clasificador convencional descartaría y, en la etapa de agregación analizada, recupera el ranking de referencia. La limitada escala del caso ($N = 7$, dos criterios, un único modelo) impide conclusiones definitivas y motiva una validación más amplia, que se discute como trabajo futuro.

Keywords: Evaluación del desempeño 360 grados · Computación con palabras · Modelo de 2-tupla lingüística · Modelos de lenguaje de gran escala (LLMs) · Agregación de información heterogénea · Interpretabilidad

1. Introducción

La evaluación del desempeño es un proceso estratégico en la gestión del talento humano que permite tomar decisiones sobre promociones, retribución y

desarrollo de empleados [26,13]. La modalidad 360 grados amplía la perspectiva tradicional al incorporar opiniones de supervisores, colaboradores, clientes y el propio empleado [14], introduciendo dos desafíos fundamentales: la *heterogeneidad* de los dominios de expresión y la *incertidumbre* inherente a las valoraciones cualitativas. Los marcos de Computación con Palabras (CWW) [5] y el modelo de 2-tuplas lingüísticas [2] han demostrado ser herramientas idóneas para abordar estos desafíos; Espinilla et al. [1] propusieron un modelo integrado que gestiona información heterogénea y emplea operadores OWA, media ponderada e integral de Choquet para obtener resultados interpretables.

Sin embargo, los sistemas actuales gestionan simultáneamente datos cuantitativos, valoraciones estructuradas y *texto libre* (comentarios, informes, respuestas cualitativas) producido de forma natural en plataformas como Workday, SAP SuccessFactors u Oracle HCM [20,27]. Aspectos como la actitud o el liderazgo son difíciles de capturar numéricamente, y los modelos CWW han ignorado este tipo de información, generando fatiga y baja participación [28].

La aparición de los LLMs [8] abre una vía para superar estas limitaciones. Frente a trabajos previos en RRHH y NLP que típicamente retienen únicamente la etiqueta más probable [23,24,21,22,16,17], este trabajo aporta un puente conceptual entre la distribución de probabilidad interna del LLM y el formalismo de la 2-tupla: dicha distribución se interpreta como una estructura de creencia [19] sobre los términos lingüísticos, lo que la hace compatible con los operadores de agregación existentes y produce resultados interpretables en lenguaje natural en todas las etapas. Esta interpretabilidad nativa de la representación (sin técnicas post-hoc [12,18]) es especialmente relevante dado que el Reglamento Europeo de IA [15] clasifica explícitamente como sistemas de alto riesgo (Anexo III, punto 4) los utilizados en empleo, gestión de trabajadores y acceso al autoempleo, exigiendo transparencia e interpretabilidad para los afectados.

Las contribuciones principales que se presentan son: (1) un marco para derivar valoraciones lingüísticas desde texto libre mediante LLMs; (2) la transformación de la distribución de logprobs a estructura de creencia y 2-tupla [19], donde la traslación simbólica α no nula preserva información ordinal que un clasificador convencional descartaría; (3) la integración transparente con el proceso de agregación de [1]; y (4) un estudio de viabilidad preliminar sobre un caso de estudio industrial real cuyos resultados son consistentes con la viabilidad del marco propuesto.

La contribución metodológica diferencial reside, por tanto, no en emplear un LLM como clasificador discreto, sino en interpretar su distribución de probabilidad restringida a la escala lingüística como una estructura de creencia, conservando mediante la traslación simbólica α información ordinal intermedia que se perdería al retener únicamente la etiqueta más probable.

2. Preliminares

Esta sección revisa los fundamentos formales sobre los que se construye el marco propuesto. En primer lugar, se introduce el modelo de representa-

ción lingüística de 2-tupla [2] y su mecanismo de gestión de información heterogénea [3]. Seguidamente, se presenta el formalismo de las estructuras de creencia y su transformación a 2-tupla [19], que constituye la base matemática del mapeo de salidas del LLM propuesto en la Sección 3. Finalmente, se describe el modelo de evaluación del desempeño 360 grados basado en CWW [1] sobre el que se integra el presente trabajo.

2.1. El modelo de representación lingüística de 2-tuplas

El modelo de 2-tuplas [2] representa la información lingüística mediante pares (s_i, α), donde $s_i \in S = \{s_0, s_1, \ldots, s_g\}$ es un término lingüístico de un conjunto predefinido y $\alpha \in [-0{,}5, 0{,}5)$ es un valor numérico denominado *traslación simbólica* que expresa la diferencia entre el valor cardinal resultante de una operación de agregación y el índice i del término más cercano.

Definition 1 ([2]). *Sea $S = \{s_0, \ldots, s_g\}$ un conjunto de términos lingüísticos. La función $\Delta_S : [0, g] \to \langle S \rangle$ se define como:*

$$\Delta_S(\beta) = (s_i, \alpha), \quad \textit{con } i = \mathrm{round}(\beta),\ \alpha = \beta - i,$$

donde round$(\cdot)$ *asigna a β el entero más próximo en $\{0, 1, \ldots, g\}$.*

La función inversa $\Delta_S^{-1}(s_i, \alpha) = i + \alpha$ permite recuperar el valor numérico equivalente, de manera que las 2-tuplas pueden identificarse con valores en $[0, g]$. Este modelo permite realizar operaciones aritméticas sobre información lingüística sin pérdida de información.

Para gestionar información heterogénea (numérica, en intervalos y lingüística con distintas granularidades) [4] Herrera et al. [3] propusieron un proceso de unificación basado en la transformación de toda la información a conjuntos difusos sobre un *Conjunto Básico de Términos Lingüísticos* (BLTS, del inglés *Basic Linguistic Term Set*) $\bar{S}$, seguida de su conversión a 2-tuplas mediante la función $\chi : \mathcal{F}(\bar{S}) \to \langle \bar{S} \rangle$.

2.2. Estructuras de creencia y transformación a 2-tuplas

Las *estructuras de creencia* (*belief structures*) proporcionan un mecanismo riguroso para representar la incertidumbre en evaluaciones lingüísticas [19]. Dada una escala lingüística $S_q = \{s_0^q, \ldots, s_{g_q}^q\}$, una estructura de creencia sobre S_q es una distribución:

$$\mathcal{B} = \{(s_j^q,\, p_j) : j = 0, \ldots, g_q\},$$

donde $p_j \geq 0$ representa el grado de creencia de que la valoración corresponde al término s_j^q. Si $\sum_{j=0}^{g_q} p_j = 1$ la estructura es *completa*; si $\sum_{j=0}^{g_q} p_j < 1$ es *incompleta* y la diferencia recoge ignorancia residual.

Para transformar una estructura de creencia completa en una 2-tupla, Martínez et al. [19] definen la función $\xi : \mathcal{B} \to [0, g_q]$ como:

$$\xi(\mathcal{B}) = \frac{\sum_{j=0}^{g_q} j \cdot p_j}{\sum_{j=0}^{g_q} p_j} = \sum_{j=0}^{g_q} j \cdot p_j = \beta, \tag{1}$$

donde la última igualdad se cumple en el caso completo ($\sum p_j = 1$). Aplicando la función Δ_{S_q} al valor β se obtiene la 2-tupla equivalente sin pérdida de información:

$$\Delta_{S_q}(\xi(\mathcal{B})) = (s_i^q, \alpha), \quad \alpha = \beta - \text{round}(\beta).$$

La traslación simbólica α captura la posición relativa de la valoración entre los términos adyacentes de la escala: $\alpha > 0$ indica proximidad al término superior, $\alpha < 0$ al inferior, y $\alpha = 0$ coincidencia exacta con un término de la escala.

En el presente trabajo $\mathcal{B}$ se obtiene de los logprobs internos del LLM en lugar de un sistema de reglas FURBER (*Fuzzy Rule-Based Evidential Reasoning*) como en [19]; el objeto matemático y la aplicación de ξ son idénticos.

2.3. Modelo de evaluación del desempeño 360 grados basado en CWW

El modelo propuesto en [1] estructura el proceso de evaluación integral en tres fases principales: (i) *marco de evaluación heterogéneo*, que define los colectivos de evaluadores A (supervisores), B (colaboradores) y C (clientes), el conjunto de empleados $X = \{x_1, \ldots, x_n\}$ y el conjunto de criterios $Y = \{Y_1, \ldots, Y_p\}$; (ii) *recogida de información*, en la que cada evaluador proporciona valoraciones en el dominio que le corresponde según su perfil y el criterio evaluado; y (iii) *proceso de valoración*, que comprende la unificación de la información heterogénea y un proceso de agregación multipaso.

El proceso de agregación opera en tres etapas sucesivas:

1. **Valores colectivos por criterio** $v_A^k(x_j)$: se obtienen aplicando el operador OWA de 2-tuplas G_w [6] —con vector de pesos w derivado de un cuantificador lingüístico— a las valoraciones de cada colectivo sobre cada criterio.
2. **Valores globales por criterio** $v^k(x_j)$: se obtienen mediante la media ponderada de 2-tuplas F_w sobre los valores colectivos, ponderando la relevancia de cada colectivo para el criterio en cuestión.
3. **Valor global del empleado** $v(x_j)$: se obtiene aplicando la integral de Choquet lingüística de 2-tuplas M_ϕ [7] sobre los valores globales por criterio, capturando la interacción entre criterios mediante medidas difusas.

Este modelo proporciona resultados lingüísticos en todas las etapas del proceso, próximos al modelo cognitivo humano y, por tanto, interpretables y comprensibles para el departamento de Recursos Humanos.

3. Marco Propuesto: LLMs como Evaluadores Lingüísticos

Proponemos extender el modelo de [1] añadiendo una etapa previa de *generación automática de valoraciones lingüísticas* a partir de texto libre mediante

LLMs. El LLM actúa como un *evaluador lingüístico proxy*: recibe el texto libre de un evaluador y el criterio a valorar, y emite un juicio lingüístico estructurado que se transforma al espacio de 2-tupla. La Figura 1 ilustra el proceso completo.

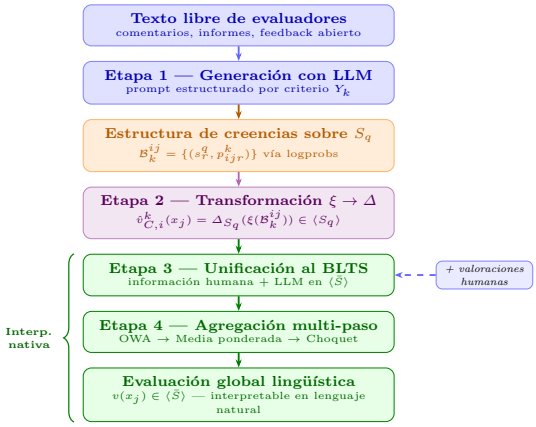


Figura 1. Proceso de evaluación del desempeño 360 grados con LLMs integrados.

3.1. Etapa 1: Generación de la estructura de creencia mediante LLM

Sea t_{ij}^C el texto libre proporcionado por el evaluador $c_i \in C$ sobre el empleado x_j, y $S_q = \{s_0^q, \ldots, s_{g_q}^q\}$ la escala lingüística asociada al criterio Y_k. Para cada par (c_i, x_j), se construye un *prompt* estructurado π que instruye al LLM para emitir una valoración sobre dicho criterio. El prompt $\pi(t_{ij}^C, Y_k, S_q)$ se organiza en tres componentes: (i) *contexto*: descripción del criterio y la escala; (ii) *instrucción*: petición de valoración en la escala indicada; y (iii) *entrada*: el texto libre del evaluador. Un ejemplo para el criterio *Atención al cliente* (Y_7) con escala $S_7 = \{\text{NA}, \text{MBA}, \text{BA}, \text{ME}, \text{AT}, \text{MAT}, \text{PE}\}$ sería:

"Eres un experto en evaluación del desempeño. El criterio a evaluar es ***Atención al cliente****: disponibilidad, saludo, agradecimiento de la compra. Utiliza la escala {NA, MBA, BA, ME, AT, MAT, PE}. A partir del siguiente comentario, indica únicamente la etiqueta más adecuada como respuesta. Comentario: [TEXTO]".*

El diseño sigue los principios de [23,24]: la asignación de rol mejora la coherencia en tareas de dominio [9], la descripción explícita de subcriterios ancla semánticamente la valoración, y restringir la respuesta a una única etiqueta facilita el mapeo directo a $\mathcal{B}$ sin posprocesamiento adicional.

Los modelos de código abierto exponen, para cada token de su vocabulario, el *logprob* $\ell_{ijr}^k = \log P(s_r^q \mid \pi, \theta)$ [25], lo que permite extraer directamente el

logprob de cada término $s_r^q \in S_q$ en la posición de respuesta:

$$\ell_{ijr}^k = \log P_{\text{LLM}}(s_r^q \mid \pi(t_{ij}^C, Y_k, S_q)).$$

Los logprobs se convierten en probabilidades mediante softmax restringida al vocabulario de la escala [10]:

$$p_{ijr}^k = \frac{\exp(\ell_{ijr}^k)}{\sum_{h=0}^{g_q} \exp(\ell_{ijh}^k)}, \quad r = 0, \ldots, g_q.$$

Debe observarse que p_{ijr}^k no representa la probabilidad absoluta asignada por el LLM al término s_r^q sobre todo el vocabulario, sino una probabilidad condicionada y renormalizada sobre el conjunto finito de etiquetas de la escala S_q. Por tanto, la estructura de creencia $\mathcal{B}_k^{ij}$ debe interpretarse como una distribución de creencia relativa sobre los términos lingüísticos admisibles.

El resultado es una **estructura de creencia completa** sobre S_q (véase la Sección 2.2):

$$\mathcal{B}_k^{ij} = \{(s_r^q,\, p_{ijr}^k) : r = 0, \ldots, g_q\}, \quad \sum_{r=0}^{g_q} p_{ijr}^k = 1. \tag{2}$$

La distribución $\mathcal{B}_k^{ij}$ refleja la incertidumbre interna del modelo: si el texto es inequívoco, p_{ijr}^k se concentra en un único índice; si es ambiguo entre dos términos adyacentes, la masa se reparte entre ellos. En el experimento se fijó $T = 0$ y un prompt constante para garantizar reproducibilidad.

Se asume que cada etiqueta de S_q es un único token (verificado para las abreviaturas usadas) y que los logprobs son informativos con $T = 0$ y prompt delimitado [8,25]; $\mathcal{B}_k^{ij}$ debe interpretarse como incertidumbre relativa, no como probabilidad frecuentista.

3.2. Etapa 2: Transformación de la estructura de creencia a 2-tupla

Una vez obtenida la estructura de creencia $\mathcal{B}_k^{ij}$ (Ecuación 2), se aplica la función ξ definida en la Sección 2.2 para obtener el valor numérico equivalente:

$$\beta_k^{ij} = \xi\left(\mathcal{B}_k^{ij}\right) = \sum_{r=0}^{g_q} r \cdot p_{ijr}^k \;\in\; [0,\, g_q]. \tag{3}$$

Aplicando la función Δ_{S_q} se obtiene directamente la 2-tupla lingüística que representa la valoración del evaluador c_i sobre el empleado x_j en el criterio Y_k:

$$\hat{v}_{C,i}^k(x_j) = \Delta_{S_q}\left(\beta_k^{ij}\right) = (s_r^q,\, \alpha_k^{ij}), \tag{4}$$

donde $r = \text{round}(\beta_k^{ij})$ y $\alpha_k^{ij} = \beta_k^{ij} - r \in [-0{,}5,\, 0{,}5)$.

La traslación simbólica α_k^{ij} emerge directamente de la distribución del LLM, sin heurísticas externas: $|\alpha_k^{ij}|$ cercano a 0,5 indica masa repartida entre dos términos adyacentes, y $\alpha_k^{ij} \approx 0$ concentración en un único término.

3.3. Integración en el proceso de agregación

Una vez obtenida la 2-tupla $\hat{v}^k_{C,i}(x_j) \in \langle S_q \rangle$, el proceso continúa de manera idéntica al modelo de [1]: el LLM actúa como un evaluador adicional cuyas valoraciones son formalmente indistinguibles de las humanas en el espacio de 2-tuplas. El estudio de viabilidad de la Sección 4 valida el puente logprobs → 2-tupla y la etapa OWA; las etapas restantes operan sobre 2-tuplas sin adaptación.

4. Estudio de Viabilidad

4.1. Descripción del caso de estudio

Para evaluar la viabilidad del marco propuesto empleamos el caso de estudio industrial real de [1], correspondiente a una empresa multinacional del sector textil con 7 empleados evaluados ($x_1, \ldots, x_7$), 11 criterios ($Y_1, \ldots, Y_{11}$) y 38 evaluadores distribuidos en tres colectivos: supervisores ($|A| = 7$), colaboradores ($|B| = 11$) y clientes ($|C| = 20$).

El experimento se focaliza en el colectivo de **clientes** ($|C| = 20$) y los dos criterios que este colectivo evalúa: Y_7 (*Atención al cliente*: disponibilidad, saludo y agradecimiento de la compra) con escala $S_7 = \{\text{NA}, \text{MBA}, \text{BA}, \text{ME}, \text{AT}, \text{MAT}, \text{PE}\}$, y Y_{11} (*Imagen personal*: forma de vestir y trato) con escala $S_3 = \{\text{NA}, \text{ME}, \text{PE}\}$, donde las abreviaturas denotan, en orden creciente: Nada (NA), Muy Bajo (MBA), Bajo (BA), Medio (ME), Alto (AT), Muy Alto (MAT) y Perfecto (PE) para S_7, y Nada (NA), Medio (ME) y Perfecto (PE) para S_3. Esto supone un total de **280 valoraciones de referencia** (140 por criterio).

La Tabla 1 recoge las valoraciones de referencia del experimento.

Tabla 1. Muestra de valoraciones de referencia (c_1–c_5) sobre Y_7 (escala S_7) e Y_{11} (escala S_3).

	Y_7							Y_{11}						
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_1	x_2	x_3	x_4	x_5	x_6	x_7
c_1	ME	ME	MBA	ME	NA	MAT	PE	PE	ME	ME	PE	ME	PE	PE
c_2	ME	MBA	NA	MBA	MBA	ME	MAT	PE	ME	ME	PE	ME	PE	PE
c_3	ME	MBA	NA	MBA	ME	ME	MAT	ME	ME	NA	PE	ME	PE	PE
c_4	ME	MBA	NA	MBA	ME	MAT	MAT	ME	ME	NA	ME	ME	ME	PE
c_5	ME	MBA	NA	MBA	MBA	MAT	MAT	ME	ME	ME	NA	ME	ME	ME

4.2. Generación de texto libre sintético

Esta prueba de concepto requiere texto libre como entrada al LLM. Dado que el caso de estudio original recogió valoraciones estructuradas pero no comentarios textuales, se ha generado texto sintético alineado con cada valoración de referencia. Para cada trío (c_i, x_j, Y_k) se redactó un comentario de 2–3 frases en

lenguaje coloquial que describiese conductualmente un desempeño consistente con la etiqueta lingüística de referencia, evitando deliberadamente el término de la etiqueta y sus sinónimos directos. Esta restricción busca que el LLM tenga que inferir el nivel a partir de descripciones de comportamiento (el escenario realista de aplicación) y no por simple coincidencia léxica con la escala. La redacción la realizó manualmente una persona con formación en Economía y experiencia en el dominio, sin asistencia de ningún LLM. Como limitación, el redactor conocía la valoración de referencia, lo que podría introducir sesgos sutiles (*demand characteristics*) pese a la restricción léxica; un control completo requeriría anotación ciega o textos reales y queda pendiente para trabajo futuro. La Tabla 2 muestra ejemplos representativos para distintos niveles de valoración.

Tabla 2. Ejemplos de textos sintéticos para distintas valoraciones de referencia.

Crit.	Par	Ref.	Texto sintético generado
Y_7	c_1, x_5	NA	*"No recibí ninguna atención durante toda la visita. Ningún empleado se acercó ni respondió cuando intenté llamarles."*
Y_7	c_3, x_1	ME	*"El servicio fue aceptable. Cuando pregunté me atendieron, aunque no estaban muy pendientes y tardaron en acercarse."*
Y_7	c_1, x_6	MAT	*"La atención fue muy buena. La dependienta estaba muy dispuesta, nos saludó al entrar y se ofreció a ayudarnos en varias ocasiones."*
Y_7	c_1, x_7	PE	*"Atención impecable. Nos recibieron con una sonrisa, resolvieron todas las dudas y nos agradecieron la compra al marcharnos."*
Y_{11}	c_3, x_3	NA	*"La empleada iba bastante desaliñada, su ropa no guardaba relación con el estilo de la tienda y causó una impresión muy negativa."*
Y_{11}	c_1, x_2	ME	*"La empleada vestía de forma correcta, sin que su imagen destacara especialmente ni en positivo ni en negativo."*
Y_{11}	c_1, x_1	PE	*"Imagen impecable. Vestía con mucho estilo, muy acorde a la filosofía de la marca, y transmitía una imagen muy cuidada y profesional."*

Es importante señalar que los textos sintéticos son más directos y menos ambiguos que los comentarios reales, por lo que las métricas obtenidas deben interpretarse como un *límite superior* del rendimiento esperable con texto humano real.

4.3. Configuración del LLM y métricas de evaluación

Se empleó el modelo `Llama-3.1-8B-Instruct` [11] en cuantización Q4_K_M, servido localmente mediante `llama.cpp` sobre hardware Apple Silicon con aceleración Metal. Se eligió por ser open-weight (acceso completo a logprobs y reproducibilidad total) y ejecutable en hardware de consumo, requisitos que los modelos propietarios o de mayor tamaño no satisfacen simultáneamente. Se fijó $T = 0$ (decodificación greedy determinista), `seed=42`, `n_predict=1` y `n_probs=20`, renormalizando por softmax restringida a los términos de S_q para obtener $\{p^k_{ijr}\}$; la baja dispersión de los logprobs de respuesta sustenta que la masa asignada a tokens fuera de la escala es ruido no semántico. Se realizaron 280 inferencias

independientes. Se asume que la cuantización Q4_K_M tiene un impacto despreciable en el orden relativo de los logprobs dado el uso de etiquetas de un único token y escalas de baja granularidad; su verificación cuantitativa queda como trabajo futuro.

Las métricas de evaluación utilizadas fueron: **Tasa de Coincidencia Exacta (TCE)**, definida como la proporción de inferencias en las que $s_r^q = \text{argmax}_r\, p_{ijr}^k$ coincide con la etiqueta de referencia; **Error de Traslación Medio (ETM)**, calculado como $|\beta_k^{ij} - r_{\text{ref}}|/g$ promediado sobre las inferencias (donde r_{ref} es el índice de la etiqueta de referencia y g la granularidad de la escala); y la **correlación de Kendall** τ sobre el ranking global de empleados.

Adicionalmente, para cuantificar la aportación del mapeo probabilístico, se compara con un **baseline** que utiliza únicamente la etiqueta más probable ($s_r = \text{argmax}_r\, p_{ijr}^k$) con $\alpha = 0$, descartando la distribución de probabilidad sobre la escala. Este baseline representa el uso habitual de un LLM como clasificador discreto: se conserva únicamente la etiqueta más probable y se descarta toda la distribución de probabilidad restante, como es habitual en la literatura de LLM-as-a-judge [23,24]. Permite así aislar el efecto específico de utilizar la estructura completa de creencias y la traslación simbólica α.

4.4. Resultados y discusión

Alcance del experimento. El estudio de viabilidad se centra en validar el puente logprobs → 2-tupla y la primera etapa de agregación (OWA intracolectivo). Las dos etapas restantes del pipeline de [1] no se ejecutan en este experimento por dos razones: (i) la media ponderada intercolectivos requeriría incorporar también supervisores y colaboradores, mientras que los textos disponibles proceden únicamente del colectivo de clientes; y (ii) la integral de Choquet intercriterios exigiría una medida difusa sobre $\{Y_7, Y_{11}\}$ cuya elicitación con expertos queda fuera del alcance de esta prueba de concepto. El ranking global se obtiene, por tanto, mediante la media de las valoraciones colectivas en Y_7 e Y_{11}, equivalente a una integral de Choquet con medida aditiva uniforme. La validación de las etapas 2 y 3 con datos completos del caso de estudio queda como trabajo futuro.

La Tabla 3 resume las métricas obtenidas por el método propuesto y el baseline. El método propuesto alcanza una TCE global del 88,9 % (80,0 % en Y_7 y 97,9 % en Y_{11}), con un ETM global de 0,041. El mejor rendimiento en Y_{11} es coherente con la menor granularidad de su escala: con 3 etiquetas la probabilidad de error de clasificación es inherentemente menor que con 7.

Un resultado aparentemente paradójico es que el baseline presenta una TCE global *mejor* (92,1 % vs. 88,9 %). Una posible explicación es que el método propuesto incorpora $\alpha \neq 0$ cuando el modelo distribuye masa entre dos términos adyacentes, lo que incrementa el ETM individual pero retiene información ordinal que el baseline descarta. Cabe señalar que este efecto no se limita a un ajuste de α: cuando la masa de probabilidad cruza el punto medio entre dos términos adyacentes, el redondeo de Δ_{S_q} desplaza también el término base, como ocurre en x_4 sobre Y_7 (Tabla 4), donde el método propuesto agrega a $(MB,\ -,43)$

Tabla 3. Métricas comparativas: método propuesto vs. baseline (argmax, $\alpha = 0$).

Criterio	**Método**	**TCE (%)**	**ETM**
Y_7 (S_7, 7 términos)	Propuesto	80,0	0,052
	Baseline	86,4	0,023
Y_{11} (S_3, 3 términos)	Propuesto	97,9	0,031
	Baseline	97,9	0,021
Global (280 inf.)	Propuesto	88,9	0,041
	Baseline	92,1	0,022

Método	**Ranking global**	$\boldsymbol{\tau}$ (p)
Propuesto	$x_7 \succ x_6 \succ x_1 \succ x_4 \succ x_5 \succ x_2 \succ x_3$	1,000 ($p < 0{,}001$)
Baseline	$x_7 \succ x_6 \succ x_1 \succ x_2 \succ x_4 \succ x_5 \succ x_3$	0,810 ($p = 0{,}011$)
Referencia	$x_7 \succ x_6 \succ x_1 \succ x_4 \succ x_5 \succ x_2 \succ x_3$	—

frente al (CNA, ,33) del baseline. En el caso analizado, esta diferencia se traduce en rankings distintos: el método propuesto recupera el ranking de referencia ($\tau = 1{,}000$, $p < 0{,}001$), mientras que el baseline incurre en dos pares discordantes —x_2 se sitúa por delante de x_4 y de x_5, mientras que en la referencia ocurre lo contrario— ($\tau = 0{,}810$, $p = 0{,}011$). Una causa plausible es que, al colapsar $\alpha = 0$, el baseline produce empates en la agregación OWA (x_2, x_4 y x_5 comparten (CNA, ,33) en Y_7, véase Tabla 4) que limitan la discriminación entre empleados con distribuciones de probabilidad distintas. Conviene insistir en que estos hallazgos son ilustrativos, no concluyentes: con $N = 7$ no es posible descartar que la diferencia observada se atenúe o se invierta en muestras mayores.

La Tabla 4 detalla las valoraciones colectivas (OWA, cuantificador "Most") unificadas al BLTS $\bar{S} = S_9 = \{\text{NA}, \text{CNA}, \text{MB}, \text{BA}, \text{ME}, \text{AT}, \text{LAT}, \text{MA}, \text{PE}\}$. Nótese que las abreviaturas difieren entre escalas: el término *Muy Bajo* se abrevia MBA en S_7 y MB en S_9, al tratarse de conjuntos de términos distintos con convenciones propias. En esta tabla se aprecian los empates del baseline en Y_7.

Tabla 4. Valoraciones colectivas (OWA) en $\bar{S} = S_9$ para cada empleado, por criterio.

		x_1	x_2	x_3	x_4	x_5	x_6	x_7
Y_7	Referencia	(BA, −.07)	(CNA, .07)	(NA, .00)	(CNA, .33)	(CNA, .20)	(AT, −.47)	(LAT, −.13)
	Propuesto	(BA, −.22)	(CNA, .38)	(CNA, −.05)	(MB, −.43)	(CNA, .41)	(ME, .17)	(LAT, −.44)
	Baseline	(BA, −.07)	(CNA, .33)	(CNA, −.07)	(CNA, .33)	(CNA, .33)	(AT, −.47)	(LAT, −.13)
Y_{11}	Referencia	(AT, −.20)	(ME, .00)	(BA, −.20)	(ME, .00)	(ME, .00)	(ME, .00)	(MA, −.20)
	Propuesto	(AT, −.09)	(ME, .00)	(ME, −.02)	(ME, .12)	(ME, .02)	(ME, .11)	(MA, −.17)
	Baseline	(AT, −.20)	(ME, .00)	(ME, .00)	(ME, .00)	(ME, .00)	(ME, .00)	(MA, −.20)

5. Conclusiones y Trabajo Futuro

Hemos propuesto un marco que integra LLMs como evaluadores lingüísticos en el modelo CWW de evaluación 360 grados, mapeando sus logprobs a una es-

tructura de creencia y de ahí a 2-tupla. Compatible con [1] y aplicable a cualquier evaluación lingüística multicriterio con texto libre, el sistema ofrece interpretabilidad nativa en lenguaje natural, alineada con los requisitos del Reglamento Europeo de IA [15] para sistemas de alto riesgo. El estudio de viabilidad arroja resultados consistentes con la hipótesis de partida (TCE del 88,9 % y recuperación del ranking de referencia, $\tau = 1{,}000$ frente a $\tau = 0{,}810$ del baseline), si bien el reducido tamaño muestral, el carácter sintético de los textos y el uso de un único modelo cuantizado limitan el alcance de las conclusiones que cabe extraer. Estos resultados deben entenderse, por tanto, como una primera evidencia de que la traslación simbólica derivada de la distribución del LLM puede aportar información ordinal útil al proceso de agregación.

Las conclusiones deben interpretarse con cautela. El experimento valida la viabilidad técnica del mapeo propuesto, pero no demuestra todavía la validez externa del sistema en entornos reales de evaluación de desempeño. Las limitaciones principales son el uso de textos sintéticos, el tamaño reducido del caso ($N = 7$, dos criterios, un único colectivo), un solo LLM cuantizado sin calibración probabilística y la validación parcial de la agregación.

Como trabajo futuro, las líneas prioritarias son validar con texto libre real en colaboración con empresas, explorar estructuras de creencia incompletas del tipo ($\sum p_{ijr}^k < 1$), extender el análisis a escalas de mayor granularidad (S_9, S_{11}) y comparar distintos modelos open-weight (Qwen,Deepseek etc) para verificar la independencia del marco respecto al LLM subyacente.

Referencias

1. Espinilla, M., de Andrés, R., Martínez, F.J., Martínez, L.: A 360-degree performance appraisal model dealing with heterogeneous information and dependent criteria. Information Sciences **222**, 459–471 (2013)
2. Herrera, F., Martínez, L.: A 2-tuple fuzzy linguistic representation model for computing with words. IEEE Transactions on Fuzzy Systems **8**(6), 746–752 (2000)
3. Herrera, F., Martínez, L., Sánchez, P.J.: Managing non-homogeneous information in group decision making. European Journal of Operational Research **166**(1), 115–132 (2005)
4. Martínez, L., Herrera, F.: An overview on the 2-tuple linguistic model for computing with words in decision making: extensions, applications and challenges. Information Sciences **207**(1), 1–18 (2012)
5. Zadeh, L.A.: From computing with numbers to computing with words: from manipulation of measurements to manipulation of perceptions. IEEE Transactions on Circuits and Systems I **46**(1), 105–119 (1999)
6. Yager, R.R.: On ordered weighted averaging operators in multicriteria decision making. IEEE Transactions on Systems, Man, and Cybernetics **18**, 183–190 (1988)
7. Choquet, G.: Theory of Capacities. Annales de l'Institut Fourier **5**, 131–295 (1953)
8. Brown, T.B., et al.: Language models are few-shot learners. Advances in Neural Information Processing Systems **33**, 1877–1901 (2020)
9. Achiam, J., et al.: GPT-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
10. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
11. Dubey, A., et al.: The Llama 3 Herd of Models. arXiv preprint arXiv:2407.21783 (2024)

12. Arrieta, A.B., et al.: Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion **58**, 82–115 (2020)
13. Fisher, C., Schoenfeldt, L.F., Shaw, J.B.: Human Resources Management. Houghton Mifflin Company, Boston (2006)
14. Lepsinger, R., Lucia, A.D.: The Art and Science of 360 Degree Feedback. John Wiley & Sons (2009)
15. European Parliament and Council of the European Union: Regulation (EU) 2024/1689 of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, L series (2024)
16. Ma, X., Liu, W., Zhao, C., Tukhvatulina, L.R.: Can large language model predict employee attrition? arXiv preprint arXiv:2411.01353 (2024)
17. Dasaklis, T.K., Giannopoulos, P.G., Koutras, D., Malamas, V., Chountalas, P.: Large language models in human resource management: a systematic literature review of applications, open issues and future research directions. SSRN preprint 5314976 (2025)
18. Zhao, H., Chen, H., Yang, F., Liu, N., Deng, H., Cai, H., Wang, S., Yin, D., Du, M.: Explainability for large language models: a survey. ACM Transactions on Intelligent Systems and Technology **15**(2), 1–38 (2024)
19. Martínez, L., Liu, J., Yang, J.B., 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)
20. Aguinis, H.: Performance Management, 4th edn. Chicago Business Press, Chicago (2019)
21. Speer, A.B.: Scoring dimension-level job performance from narrative comments: validity and generalizability when using natural language processing. Organizational Research Methods **24**(3), 572–597 (2021)
22. Speer, A.B., Perrotta, J., Kordsmeyer, T.L.: Taking it easy: off-the-shelf versus fine-tuned supervised modeling of performance appraisal text. Organizational Research Methods **29**(1), 73–91 (2026) https://doi.org/10.1177/10944281241271249
23. Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., et al.: Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. Advances in Neural Information Processing Systems **36**, 46595–46623 (2023)
24. Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., Zhu, C.: G-EVAL: NLG evaluation using GPT-4 with better human alignment. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2511–2522 (2023)
25. Kauf, C., Chersoni, E., Lenci, A., Fedorenko, E., Ivanova, A.A.: Log probabilities are a reliable estimate of semantic plausibility in base and instruction-tuned language models. In: Proceedings of the 7th BlackboxNLP Workshop, pp. 263–277. Association for Computational Linguistics (2024)
26. DeNisi, A.S., Murphy, K.R.: Performance appraisal and performance management: 100 years of progress? Journal of Applied Psychology **102**(3), 421–433 (2017)
27. Pulakos, E.D., Mueller-Hanson, R.A., Arad, S.: The evolution of performance management: searching for value. Annual Review of Organizational Psychology and Organizational Behavior **6**, 249–271 (2019)
28. Pulakos, E.D., Mueller Hanson, R., Arad, S., Moye, N.: Performance management can be fixed: an on-the-job experiential learning approach for complex behavior change. Industrial and Organizational Psychology **8**(1), 51–76 (2015)

Monotonía en Sistemas de Inferencia Fundamentados en la f-inclusión

Carolina Díaz-Montarroso[1,2], Nicolás Madrid[3], y Jorge Elorza[1,2]

[1] Dpto. de Física y Matemática Aplicada, Facultad de Ciencias, Universidad de Navarra, C. Irunlarrea 1, 31008 Pamplona
[2] Instituto de Ciencia de los Datos e Inteligencia Artificial (DATAI), Universidad de Navarra, Edificio Ismael Sánchez Bella, Campus Universitario, 31009 Pamplona
{cdiazmontar, jelorza}@unav.es
[3] Dpto. de Matemáticas, Universidad de Cádiz, C. República Saharaoui, 11510 Puerto Real
nicolas.madrid@uca.es

Resumen Este artículo se centra en los sistemas de inferencia difusos basados en la f-inclusión, que es un operador que mide la inclusión entre conjuntos difusos a través de una aplicación en el intervalo unidad. El objetivo principal de este artículo es el estudio de la propiedad de monotonía, desde una perspectiva lógica, de dicho sistema.

Palabras clave: Sistemas de inferencia difusos · f-índice de inclusión · Conexiones de Galois · Medidas de inclusión

1. Introducción

Los sistemas que hacen uso de reglas difusas para realizar inferencias son una herramienta comúnmente usada en el ámbito del razonamiento aproximado. Estos sistemas se emplean en áreas de la ciencia y la ingeniería en las que es habitual que la información esté sujeta a incertidumbre o vaguedad, y se utilizan para tareas de diagnóstico, predicción o clasificación [2,7].

En trabajos previos se ha demostrado la validez del operador de f-inclusión para modelizar un Modus Ponens Generalizado (GMP, por sus siglas en inglés) [4]. Esta regla de inferencia suele emplearse como motor de inferencia en los Sistemas de Inferencia Difusos (FIS) [1]. Más adelante, en [5] se introduce la estructura de un sistema de inferencia basado en este operador. Este nuevo FIS destaca por estar fundamentado en una Lógica Descriptiva, la cual emplea la inclusión como elemento central en los procesos de inferencia [17].

Los FIS basados en la f-inclusión presentan similaridades con los llamados Sistemas de Inferencia Relacionales (RIS), los cuales hacen uso de la Regla Composicional de Inferencia (CRI) para generar salidas. Un ejemplo conocido de RIS es el propuesto por Mamdani [13], pero existen muchos otros trabajos en la bibliografía sobre estos sistemas [3,15]. La principal diferencia que presenta nuestro sistema con los RIS consiste en el uso de la f-inclusión en lugar de la CRI para modelar el motor de inferencia. No obstante, en [9] se estudia la relación entre

este primer operador con los pares residuados, a través de los cuales suele modelarse la regla composicional de inferencia [14,18], estableciendo una conexión entre esta regla y la f-inclusión.

El propósito de este trabajo consiste en extender el estudio, ya iniciado en [5], de las propiedades del FIS basado en la f-inclusión desde el punto de vista de los sistemas lógico formales. En concreto, se estudia la propiedad de monotonía del sistema, que garantiza la preservación de las consecuencias al incluir información nueva en la base de reglas que lo compone.

A continuación, se presenta la estructura del trabajo. En la Sección 2, recordamos el conjunto de aplicaciones que actúan como índices de inclusión e introducimos la estructura de un sistema de inferencia difuso basado en la f-inclusión. En la Sección 3, introducimos el conjunto de consecuencias en nuestro FIS y proporcionamos algunas propiedades sobre su estructura. Posteriormente, presentamos resultados novedosos sobre el comportamiento del conjunto de consecuencias bajo modificaciones de las condiciones iniciales. Finalmente, en la Sección 4 presentamos las conclusiones de este trabajo y algunos trabajos futuros.

2. Sistemas de Inferencia Difusos Basados en la f-inclusión

Un sistema de inferencia difuso es un mecanismo que permite obtener información nueva en un entorno de incertidumbre a partir de una serie de reglas y de unos datos de entrada. A pesar de las diferencias existentes entre los diversos FIS presentados en la literatura, todos poseen una estructura común que se puede dividir en cuatro pasos.

- En primer lugar, se toman los datos de entrada y, mediante un proceso de *fuzzificación*, se transforman en un conjunto difuso. Generalmente, este proceso utiliza particiones difusas sobre el conjunto de datos de entrada.
- Por otro lado, el sistema posee una *base de conocimiento* compuesta por una serie de reglas difusas, las cuales condensan información proporcionada previamente por un conjunto de expertos.
- En tercer lugar, el *motor de inferencia* hace uso de las reglas de la base de conocimiento y de la entrada *fuzzificada* para obtener un conjunto difuso de salida.
- Por último, se puede aplicar un proceso opcional de *defuzzificación* que transforma el conjunto difuso de salida en un dato interpretable.

En esta sección se presenta el marco en el que se desarrollan los sistemas de inferencia difusos basados en la f-inclusión, así como la base de conocimiento y el motor de inferencia del sistema. Los procesos de *fuzzificación* y *defuzzificación* se omiten en este artículo, dado que están sujetos al contexto práctico en el que se hace uso del FIS.

2.1. f-inclusión entre Conjuntos Difusos

Antes de comenzar con la introducción del sistema, definiremos el concepto principal en torno al cual se desarrolla: la noción de f-inclusión. Este concepto es la base de las reglas difusas que componen la base de conocimiento.

El concepto de f-inclusión aparece por primera vez en [11] con el objetivo de generalizar el concepto de inclusión en el ambiente difuso. Este se diferencia de otras medidas de inclusión en que no representa esta noción a través de un único valor en el intervalo unidad $[0,1]$, sino mediante una aplicación definida de $[0,1]$ en $[0,1]$.

En este trabajo, consideraremos funciones monótonas que formen parte de una conexión de Galois y que sean deflaccionarias. Comencemos recordando el concepto de conexión de Galois.

Definición 1 ([6]). *Sean $f, \overline{f}\colon [0,1] \to [0,1]$ dos funciones. Se dice que $(f, \overline{f})$ forma una* conexión de Galois *si cumple la siguiente relación para todo $a, b \in [0,1]$:*

$$f(a) \leq b \quad \text{si y solo si} \quad a \leq \overline{f}(b).$$

En la siguiente definición concretamos el conjunto de funciones que vamos a considerar para representar la inclusión entre conjuntos difusos.

Definición 2 ([11]). *Se define* conjunto de índices de inclusión*, denotado por $\mathscr{G}$, como el conjunto de funciones $f\colon [0,1] \to [0,1]$ satisfaciendo:*

- *Monotonía creciente: $f(a) \leq f(b)$ para todo $a, b \in [0,1]$ tales que $a \leq b$.*
- *Deflación: $f(a) \leq a$ para todo $a \in [0,1]$.*
- *Continuidad a la izquierda: $f\left(\bigvee A\right) = \bigvee_{a \in A} f(a)$ para todo $A \subseteq [0,1]$.*

Observación. Dado un índice de inclusión $f \in \mathscr{G}$, siempre existe una aplicación $\overline{f}\colon [0,1] \to [0,1]$ tal que $(f, \overline{f})$ es una conexión de Galois (ver [6]).

En [10] se demuestra que el conjunto $\mathscr{G}$ posee estructura de retículo completo, cuyo supremo coincide con el supremo usual de funciones, y que tiene los siguientes elementos máximo y mínimo, respectivamente:

- $\mathrm{id}(a) = a$ para todo $a \in [0,1]$;
- $\mathbf{0}(a) = 0$ para todo $a \in [0,1]$.

A continuación, se introduce la noción de f-inclusión restringida a $\mathscr{G}$[4].

Definición 3 ([9]). *Sean A, B dos conjuntos difusos sobre un universo $\mathcal{U}$ y sea $f \in \mathscr{G}$. Decimos que A* está f-incluido en B*, denotado por $A \subseteq_f B$, si*

$$f(A(u)) \leq B(u)$$

*para todo $u \in \mathcal{U}$. Definimos el f-*índice de inclusión restringido a $\mathscr{G}$ de A en B*, denotado por $\mathrm{Inc}_{\mathscr{G}}(A, B)$, como*

$$\mathrm{Inc}_{\mathscr{G}}(A, B) = \sup\{f \in \mathscr{G} \mid A \subseteq_f B\}.$$

[4] Se utiliza el adjetivo "restringida" siguiendo la nomenclatura empleada en [9], pues en la definición original de f-inclusión no se exige la continuidad izquierda en los índices de inclusión.

Para reforzar la validez de este operador como medida de inclusión, en [8,12] se muestra que está alineado con los axiomas de Sinha-Dougherty [16].

2.2. La Base de Conocimiento

Una vez que hemos introducido la noción de f-inclusión, presentamos el marco y la estructura de las reglas que componen la base de conocimiento. Para conocer más detalles sobre la estructura de este SID, el lector puede consultar [5].

Comencemos definiendo el marco en el que se desarrolla el sistema.

Definición 4. *Un* marco *es una tupla* $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ *donde* X *e* Y *son conjuntos no vacíos, llamados* universos*, y* $\mathcal{A}_X = \{A_1, \ldots, A_n\}$ *y* $\mathcal{B}_Y = \{B_1, \ldots, B_m\}$ *son dos particiones difusas sobre los universos* X *e* Y*.*

Recordemos que una *partición difusa* sobre un universo $\mathcal{U}$ es una familia $\mathcal{A}$ finita de conjuntos difusos tales que para todo $u \in \mathcal{U}$ existe al menos un conjunto difuso $A \in \mathcal{A}$ cumpliendo $u \in \text{supp}(A)$.

La base de conocimiento determina la relación entre los miembros de las dos particiones difusas que componen el marco. Esta relación se determina a través de una serie de reglas difusas.

Definición 5. *Una* base de conocimiento (**KB**) *sobre un marco* $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ *es un conjunto de reglas de la forma*

$$\langle A_i \to B_j \quad ; f_{ij} \rangle,$$

donde $A_i \in \mathcal{A}_X$*,* $B_j \in \mathcal{B}_Y$ *y* $f_{ij} \in \mathscr{G}$*.*

Obsérvese que una **KB** puede interpretarse como una relación $\mathscr{G}$-difusa que asigna a cada par de conjuntos difusos $A_i \in \mathcal{A}_X$ y $B_j \in \mathcal{B}_Y$ la aplicación $f_{ij} \in \mathscr{G}$ de la regla $\langle A_i \to B_j \quad ; f_{ij} \rangle$. De este modo, toda **KB** puede representarse de forma tabular, tal y como se muestra en Tabla 1.

Tabla 1: Representación relacional de una base de conocimiento.

	B_1	$\cdots$	B_j	$\cdots$	B_m
A_1	f_{11}	$\cdots$	f_{1j}	$\cdots$	f_{1m}
$\vdots$	$\vdots$	$\ddots$	$\vdots$		$\vdots$
A_i	f_{i1}	$\cdots$	f_{ij}	$\cdots$	f_{im}
$\vdots$	$\vdots$		$\vdots$	$\ddots$	$\vdots$
A_n	f_{n1}	$\cdots$	f_{nj}	$\cdots$	f_{nm}

La sintaxis de nuestro FIS está determinada directamente por el marco. La siguiente definición introduce la semántica de una base de conocimiento.

Definición 6. *Sea Γ una* **KB** *sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. Se dice que un conjunto $M \subseteq X \times Y$ es* modelo de Γ *si para toda regla $\langle A_i \to B_j \quad ; f_{ij} \rangle \in \Gamma$ se cumple*

$$f_{ij}(A_i(x)) \leq B_j(y) \tag{1}$$

para todo $(x, y) \in M$.

El lector puede observar que los modelos de una base de conocimiento son conjuntos *crisp*, a pesar de que la información se representa mediante conceptos difusos. En [5] se demuestra que el conjunto de modelos de una base de conocimiento tiene estructura de retículo completo con la unión e intersección de conjuntos.

El modelo maximal de una base de conocimiento Γ, denotado por $\mathbf{M_\Gamma}$, es el que más información proporciona sobre la interpretación de las reglas. Cuanto mayores sean los índices de inclusión que aparecen en las reglas de la **KB**, habrá menos puntos en el plano $X \times Y$ que satisfagan la relación expresada en la Ecuación (1). Esto implica que el modelo maximal $\mathbf{M_\Gamma}$ contendrá menos valores. Por el contrario, cuanto menores sean los índices de inclusión de las reglas, mayor será el modelo maximal, representando un menor vínculo entre los elementos del universo X y los del universo Y.

2.3. Entradas y Salidas del Sistema

Tal y como se ha mencionado al inicio esta sección, tanto las entradas como las salidas del FIS vienen dadas en forma de conjunto difuso.

Definición 7. *Una* entrada *sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ es un conjunto difuso en X.*

El concepto de entrada se puede extender a conjuntos difusos en Y. No obstante, debido a la estructura del motor de inferencia, solo se utilizarán entradas en el universo en el que se encuentran los antecedentes de las reglas de la **KB**.

El concepto de modelo de una base de conocimiento (Definición 6) se puede extender a una **KB** y una entrada trasladándolo al entorno difuso.

Definición 8. *Sean Γ una* **KB** *y $A \in \mathcal{F}(X)$ una entrada sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. Un conjunto difuso $M \in \mathcal{F}(X \times Y)$ es* modelo de $\Gamma \cup \{A\}$ *si* $\text{supp}(M)$ *es un modelo de Γ y $M(x, y) \leq A(x)$ para todo $(x, y) \in X \times Y$.*

El conjunto de modelos de una **KB** y de una entrada tiene estructura de retículo con el orden estándar entre conjuntos difusos. En este caso, sus elementos mínimo y máximo son, respectivamente, el conjunto vacío y el conjunto difuso dado, para todo $(x, y) \in X \times Y$, por

$$\mathbf{M_{\Gamma,A}}(x, y) = \begin{cases} A(x) & \text{si } (x, y) \in \mathbf{M_\Gamma}, \\ 0 & \text{en otro caso.} \end{cases}$$

La noción de modelo en nuestro FIS permite introducir el concepto de consecuencia lógica dentro de este, tal y como se hace en los sistemas lógicos formales.

Definición 9. *Sean Γ una* **KB** *y $A \in \mathcal{F}(X)$ una entrada sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. Se dice que un conjunto difuso $B \in \mathcal{F}(Y)$ es* consecuencia *de $\Gamma \cup \{A\}$, denotado por $\Gamma \cup \{A\} \models B$, si todo modelo M de $\Gamma \cup \{A\}$ es modelo de $\Gamma \cup \{B\}$.*

2.4. Motor de Inferencia del Sistema

Finalmente, en esta sección se presenta el mecanismo principal de todo sistema de inferencia difuso: su motor de inferencia. En este caso, se trata de un Modus Ponens Generalizado basado en el f-índice de inclusión. Recordemos que el GMP se trata de una regla de inferencia que, de una regla de implicación y de una instancia similar al antecedente, se infiere una salida similar al consecuente.

En [4] se presentó la primera modelización de esta regla de inferencia a través del f-índice de inclusión. Sea $A \in \mathcal{F}(X)$ una entrada sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ y sea $\langle A_i \to B_j \quad ; f_{ij} \rangle$ una regla asociada a los conjuntos difusos $A_i \in \mathcal{A}_X$ y $B_j \in \mathcal{B}_Y$. EL GMP basado en el f-índice de inclusión aplicado a esta regla y a A genera un conjunto difuso $B_{ij} \in \mathcal{F}(Y)$ a través de la siguiente expresión:

$$\begin{array}{lll} & A_i \to B_j & \equiv f_{ij} \\ & A & \equiv \mathrm{Inc}_{\mathscr{G}}(A, A_i) \\ \hline \therefore & B_{ij} & \overline{\mathrm{Inc}}_{\mathscr{G}}(A, A_i) \circ \overline{f}_{ij}(B_j) \end{array} \tag{2}$$

Nótese que la similaridad entre la entrada y el antecedente de la regla se mide a través del f-índice de inclusión de A en A_i. Esta función se relaciona con la función f_{ij} que aparece en la regla a través de la composición de sus respectivas conexiones de Galois. En [4] se demuestra que esta modelización del GMP está alineada con los axiomas de Baldwin y Pilsworth para reglas de implicación [1].

El próximo resultado demuestra que la salida generada por el GMP presentado en la Ecuación (2) es consecuencia lógica de la instancia y de la regla a las que se aplica.

Teorema 1. *Sean Γ una base de conocimiento y $A \in \mathcal{F}(X)$ una entrada sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. Para toda regla $\langle A_i \to B_j \quad ; f_{ij} \rangle \in \Gamma$, se cumple que*

$$\{\langle A_i \to B_j \quad ; f_{ij} \rangle,\ A\} \models B_{ij},$$

siendo $B_{ij} \in \mathcal{F}(Y)$ el conjunto difuso dado, para todo $y \in Y$, por

$$B_{ij}(y) = \overline{\mathrm{Inc}}_{\mathscr{G}}(A, A_i) \circ \overline{f}_{ij}(B_j(y)).$$

El motor de inferencia del FIS es el siguiente: dada una entrada $A \in \mathcal{F}(X)$ y una **KB**, la salida del sistema se obtiene calculando la intersección de todos los conjuntos obtenidos al aplicar el GMP a la entrada A y a cada una de las reglas de la base de conocimiento.

El próximo resultado demuestra la validez de las salidas generadas por el motor de inferencia para una entrada y una base de conocimiento. Es decir, indica la *correctitud* del FIS basado en la f-inclusión.

Teorema 2. *Sean Γ una base de conocimiento y $A \in \mathcal{F}(X)$ una entrada sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. El conjunto difuso $B \in \mathcal{F}(Y)$ dado por*

$$B(y) = \inf_{\substack{A_i \in \mathcal{A}_X, \\ B_j \in \mathcal{B}_Y}} \overline{\mathrm{Inc}}_{\mathscr{G}}(A, A_i) \circ \overline{f}_{ij}(B_j(y))$$

para todo $y \in Y$, es una consecuencia de $\Gamma \cup \{A\}$, es decir, $\Gamma \cup \{A\} \models B$.

3. Propiedad de Monotonía del FIS

En esta sección se introduce el conjunto de consecuencias de una base de conocimiento y de una entrada dentro del FIS basado en la f-inclusión y se demuestra una propiedad relevante sobre la monotonía de este conjunto.

Previamente, se presentan una serie de resultados relacionados con la estructura algebraica de este conjunto de consecuencias, los cuales fueron estudiados inicialmente en [5]. Más adelante, se presenta un método para fusionar la información de dos bases de conocimiento dentro de un marco, y se estudia cómo se modifica el conjunto de consecuencias bajo este fenómeno.

3.1. La Estructura del Conjunto de Consecuencias

Dada una base de conocimiento Γ y una entrada $A \in \mathcal{F}(X)$ sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$, definimos el *conjunto de consecuencias de $\Gamma \cup \{A\}$* como:

$$\mathbb{C}_{\Gamma,A} = \{B \in \mathcal{F}(Y) \mid \Gamma \cup \{A\} \models B\}.$$

En el siguiente resultado se muestra que, cuando un conjunto difuso $B \in \mathcal{F}(Y)$ pertenece a $\mathbb{C}_{\Gamma,A}$, entonces cualquier otro conjunto difuso que contenga a B también pertenece a este conjunto.

Proposición 1 ([5]). *Sea Γ una* **KB** *y sea $A \in \mathcal{F}(X)$ una entrada sobre el marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. Sean dos conjuntos difusos $B, B' \in \mathcal{F}(Y)$ tales que $B \subseteq B'$. Si $\Gamma \cup \{A\} \models B$, entonces $\Gamma \cup \{A\} \models B'$.*

Recordemos que el conjunto de modelos de una entrada A y de una **KB** Γ posee estructura de retículo acotado. El conjunto $\mathbb{C}_{\Gamma,A}$ hereda esta estructura.

Teorema 3 ([5]). *Sea Γ una* **KB** *y sea $A \in \mathcal{F}(X)$ una entrada sobre un marco $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$. Sean $B, B' \in \mathcal{F}(Y)$ dos conjuntos difusos tales que $\Gamma \cup \{A\} \models B$ y $\Gamma \cup \{A\} \models B'$. Entonces, $\Gamma \cup \{A\} \models B \cap B'$ y $\Gamma \cup \{A\} \models B \cup B'$.*

En definitiva, el conjunto de consecuencias de una entrada A y de una base de conocimiento Γ tiene estructura de retículo completo y, según la Proposición 1, su elemento máximo es el universo Y. Además, si se conoce su elemento mínimo, se puede inferir todo el conjunto $\mathbb{C}_{\Gamma,A}$.

3.2. Fusión de Bases de Conocimiento

Dentro del contexto de los sistemas lógicos formales, la propiedad de *monotonía* de un sistema se refiere a que, cuanto mayor es el número de axiomas, mayor es el número de teoremas. En el contexto de nuestro sistema de inferencia, esta propiedad puede reformularse en los siguientes términos: la inclusión de nueva información dentro de la base de conocimiento no invalida las conclusiones previas.

Dentro del marco en el que se desarrolla nuestro FIS, la inclusión de información nueva viene dada en forma de nuevas reglas que están contenidas dentro de una base de conocimiento.

Cuando se unen dos **KB**s, los antecedentes y los consecuentes de sus reglas han de estar definidos respectivamente en los mismos universos; no obstante, pueden estar definidos para diferentes particiones. Por tanto, al fusionarse, también ha de generarse un nuevo marco que incluya todos los conjuntos difusos involucrados en las reglas.

Definición 10. *Sean* $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ *y* $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$ *dos marcos disjuntos, es decir, que sus particiones no poseen conjuntos difusos en común. Sean* Γ *y* Π *dos bases de conocimiento definidas en* $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ *y en* $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$*, respectivamente:*

$$\Gamma = \{\langle A_i \to B_j \quad ; f_{ij}\rangle \mid A_i \in \mathcal{A}_X,\ B_j \in \mathcal{B}_Y,\ f_{ij} \in \mathscr{G}\}\ \ y$$
$$\Pi = \{\langle C_k \to D_l \quad ; g_{kl}\rangle \mid C_k \in \mathcal{C}_X,\ D_l \in \mathcal{D}_Y,\ g_{kl} \in \mathscr{G}\}\,.$$

La unión de Γ y Π*, denotada por* $\Gamma \sqcup \Pi$*, consiste en la* **KB** *definida sobre el marco* $(X, Y, \mathcal{A}_X \cup \mathcal{C}_X, \mathcal{B}_Y \cup \mathcal{D}_Y)$ *donde la asignación de un índice de inclusión para cada par de conjuntos difusos en* $(\mathcal{A}_X \cup \mathcal{C}_X) \times (\mathcal{B}_Y \cup \mathcal{D}_Y)$*, interpretando* **KB** *como una relación* $\mathscr{G}$*-difusa, se realiza bajo los siguientes criterios:*

- *Cuando el par de conjuntos difusos pertenece a* $\mathcal{A}_X \times \mathcal{B}_Y$*, entonces su índice de inclusión asociado es el que tiene originalmente asignado en* Γ*.*
- *Análogamente, cuando el par de conjuntos difusos pertenece a* $\mathcal{C}_X \times \mathcal{D}_Y$*, entonces su índice de inclusión asociado es el que tiene originalmente asignado en* Π*.*
- *Si los conjuntos difusos pertenecen a particiones en marcos diferentes (sea* $\mathcal{A}_X \times \mathcal{D}_Y$ *o* $\mathcal{C}_X \times \mathcal{B}_Y$*), entonces su índice de inclusión asociado es* $\mathbf{0}$*.*

En el último caso de la Definición 10, se asume que el índice de inclusión asociado a la regla es $\mathbf{0}$ pues se desconoce la relación entre los conjuntos difusos involucrados, y este índice de inclusión es equivalente a no imponer ninguna restricción.

Para facilitar la comprensión de esta definición, en la Tabla 2 se muestra la representación relacional de la base de conocimiento resultante al unir dos **KB**s distintas.

En el siguiente resultado se demuestra que, cuando se fusionan dos bases de conocimiento en un mismo marco, no sólo se conservan los conjuntos de consecuencias anteriores, sino que la intersección de consecuencias de los diferentes marcos pertenece al nuevo conjunto de consecuencias.

Tabla 2: Representación relacional de la fusión de bases de conocimiento según los criterios expresados en la Definición 10. A la izquierda, representación de dos **KB**s Γ y Π. A la derecha, representación de la unión $\Gamma \sqcup \Pi$.

Base de conocimiento Γ:

	B_1	$\cdots$	B_j	$\cdots$	B_m
A_1	f_{11}	$\cdots$	f_{1j}	$\cdots$	f_{1m}
$\vdots$	$\vdots$	$\ddots$	$\vdots$		$\vdots$
A_i	f_{i1}	$\cdots$	f_{ij}	$\cdots$	f_{im}
$\vdots$	$\vdots$		$\vdots$	$\ddots$	$\vdots$
A_n	f_{n1}	$\cdots$	f_{nj}	$\cdots$	f_{nm}

Base de conocimiento Π:

	D_1	$\cdots$	D_l	$\cdots$	D_s
C_1	g_{11}	$\cdots$	g_{1l}	$\cdots$	g_{1s}
$\vdots$	$\vdots$	$\ddots$	$\vdots$		$\vdots$
C_k	g_{k1}	$\cdots$	g_{kl}	$\cdots$	g_{ks}
$\vdots$	$\vdots$		$\vdots$	$\ddots$	$\vdots$
C_r	g_{r1}	$\cdots$	g_{rl}	$\cdots$	g_{rs}

Fusión de las KBs Γ y Π ($\Gamma \sqcup \Pi$):

	B_1	$\cdots$	B_m	D_1	$\cdots$	D_s
A_1	f_{11}	$\cdots$	f_{1m}	$\mathbf{0}$	$\cdots$	$\mathbf{0}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$		$\vdots$
A_n	f_{n1}	$\cdots$	f_{nm}	$\mathbf{0}$	$\cdots$	$\mathbf{0}$
C_1	$\mathbf{0}$	$\cdots$	$\mathbf{0}$	g_{11}	$\cdots$	g_{1s}
$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\ddots$	$\vdots$
C_r	$\mathbf{0}$	$\cdots$	$\mathbf{0}$	g_{r1}	$\cdots$	g_{rs}

Teorema 4. *Sean $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ y $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$ dos marcos con sus correspondientes* **KB***s Γ y Π. Sean $A \in \mathcal{F}(X)$ una entrada y $B, D \in \mathcal{F}(Y)$ dos conjuntos difusos. Si*

- $\Gamma \cup \{A\} \models B$ *en* $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ *y*
- $\Pi \cup \{A\} \models D$ *en* $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$,

entonces $\{\Gamma \sqcup \Pi\} \cup \{A\} \models B \cap D$ *en* $(X, Y, \mathcal{A}_X \cup \mathcal{C}_X, \mathcal{B}_Y \cup \mathcal{D}_Y)$.

Esta propiedad del FIS se refleja en las salidas del motor de inferencia tal y como se muestra en el siguiente resultado.

Teorema 5. *Sean $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ y $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$ dos marcos con sus correspondientes* **KB***s Γ y Π, y sea $A \in \mathcal{F}(X)$ una entrada. Sean B y D las salidas obtenidas tras aplicar el motor de inferencia a Γ y A en $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ y a Π y A en $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$, respectivamente. Entonces, la salida obtenida aplicando el motor de inferencia a $\Gamma \sqcup \Pi$ y A es $B \cap D$.*

El siguiente resultado muestra que nuestro FIS es monótono.

Corolario 1. *Sean $(X, Y, \mathcal{A}_X, \mathcal{B}_Y)$ y $(X, Y, \mathcal{C}_X, \mathcal{D}_Y)$ dos marcos con sus correspondientes* **KB***s Γ y Π, y sea $A \in \mathcal{F}(X)$ una entrada. Entonces, $\mathbb{C}_{\Gamma,A} \subseteq \mathbb{C}_{\Gamma \sqcup \Pi,A}$ y $\mathbb{C}_{\Pi,A} \subseteq \mathbb{C}_{\Gamma \sqcup \Pi,A}$.*

4. Conclusiones y Trabajo Futuro

En este trabajo se ha recordado la estructura de los sistemas de inferencia basados en la f-inclusión, prestando especial atención a sus conjuntos de reglas y al mecanismo del motor de inferencia. Entre las novedades presentadas por este FIS, una de las más destacables es su fundamento lógico, que permite incluir la definición de consecuencia lógica dentro de su marco. Esta noción contribuye al FIS en dos aspectos distintos: es un elemento clave para demostrar la correctitud del sistema y, al mismo tiempo, permite analizar el conjunto de consecuencias del FIS independientemente de las salidas obtenidas por el motor de inferencia.

Con respecto a este último aspecto, este artículo se centra en el comportamiento del conjunto de consecuencias bajo modificaciones en el conjunto de reglas del sistema. Hemos concluido que la unión de bases de conocimiento preserva las consecuencias previas, lo cual demuestra la monotonía del FIS como sistema lógico formal.

Con estos resultados, hemos obtenido una herramienta poderosa para la búsqueda de algoritmos orientados a la resolución de problemas de regresión. Como trabajo futuro, pretendemos profundizar en este aspecto y hallar un resultado de aproximación universal de funciones continuas haciendo uso de este sistema de inferencia difuso basado en la f-inclusión.

Si bien el proceso de fusión de bases de conocimiento tiene una utilidad innegable, presenta una clara desventaja, a saber, el incremento del coste computacional del sistema. No obstante, los avances presentados en este trabajo pueden ayudarnos en el futuro a explorar métodos adicionales para modificar los conjuntos de reglas con el fin de optimizar la implementación del motor de inferencia sin requerir recursos computacionales adicionales. También será interesante explorar las cuestiones relacionadas con la gestión y optimización de bases de reglas difusas, una temática de interés recurrente en el ámbito de los sistemas de inferencia.

Agradecimientos. Carolina Díaz Montarroso y Jorge Elorza agradecen el apoyo financiero del proyecto PID2024-155289NB-I00 financiado por MICIU /AEI / 10.13039/501100011033 / FEDER, UE. Carolina Díaz Montarroso agradece el apoyo de la Asociación de Amigos de la Universidad de Navarra.

Nicolás Madrid agradece el apoyo financiero de los proyectos PID2022-137620NB-I00 y PID2022-140630NB-I00 financiados por MICIU/AEI/10.13039/501100011033 y FEDER, UE, de la beca TED2021-129748B-I00 financiada por MCIN/AEI/10.13039/501100011033 y NextGenerationEU/PRTR, y del proyecto FEDER-UCA-2024-A2-04 financiado por el Programa Operativo FEDER Andalucía 2021-2027 y por la Consejería de Universidad, Investigación e Innovación, Junta de Andalucía.

Declaración de intereses. Los autores declaran no tener ningún conflicto de intereses que pudiera haber influido en el trabajo presentado en este artículo.

Referencias

1. Baldwin, J.F., Pilsworth, B.W.: Axiomatic approach to implication for approximate reasoning with fuzzy logic. Fuzzy Sets and Systems **3**(2), 193–219 (1980). https://doi.org/10.1016/0165-0114(80)90054-8
2. Cao, J., Zhou, T., Zhi, S., Lam, S., Ren, G., Zhang, Y., Wang, Y., Dong, Y., Cai, J.: Fuzzy inference system with interpretable fuzzy rules: Advancing explainable artificial intelligence for disease diagnosisA comprehensive review. Information Sciences **662**, 120212 (Mar 2024). https://doi.org/10.1016/j.ins.2024.120212
3. Daňková, M.: On approximate reasoning with graded rules. Fuzzy Sets and Systems **158**(6), 652–673 (2007). https://doi.org/10.1016/j.fss.2006.11.019
4. Díaz-Montarroso, C., Madrid, N., Ramírez-Poussa, E.: Towards a Generalized Modus Ponens Based on the f-Index of Inclusion. In: Conceptual Knowledge Structures: First International Joint Conference, CONCEPTS 2024, Cádiz, Spain, September 9–13, 2024, Proceedings. pp. 36–48. Springer-Verlag, Berlin, Heidelberg (2024). https://doi.org/10.1007/978-3-031-67868-4_3
5. Díaz-Montarroso, C., Madrid, N., Ramírez-Poussa, E.: Correctness of Fuzzy Inference Systems Based on f-Inclusion. Mathematics **13**(11) (2025). https://doi.org/10.3390/math13111897
6. Ganter, B., Wille, R.: Formal concept analysis: mathematical foundations. Springer Science & Business Media, Berlin, Heidelberg, 1 edn. (1999). https://doi.org/10.1007/978-3-642-59830-2
7. Kazeminezhad, M., Etemad-Shahidi, A., Mousavi, S.: Application of fuzzy inference system in the prediction of wave parameters. Ocean Engineering **32**(14-15), 1709–1725 (Oct 2005). https://doi.org/10.1016/j.oceaneng.2005.02.001
8. Madrid, N., Ojeda-Aciego, M.: A view of f-indexes of inclusion under different axiomatic definitions of fuzzy inclusion. In: Moral, S., Pivert, O., Sánchez, D., Marín, N. (eds.) Scalable Uncertainty Management. vol. 10564, pp. 307–318. Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-67582-4_22
9. Madrid, N., Ojeda-Aciego, M.: The f-index of inclusion as optimal adjoint pair for fuzzy modus ponens. Fuzzy Sets and Systems **466**, 108474 (2023). https://doi.org/10.1016/j.fss.2023.01.009
10. Madrid, N., Ojeda-Aciego, M.: Composition as a fuzzy conjunction between indexes of inclusion. Fuzzy Sets and Systems **527**, 109685 (2026). https://doi.org/10.1016/j.fss.2025.109685
11. Madrid, N., Ojeda-Aciego, M., Perfiljeva, I.: f-inclusion indexes between fuzzy sets. In: Proceedings of the 2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology. pp. 1528–1533. Advances in Intelligent Systems Research, Atlantis Press (2015). https://doi.org/10.2991/ifsa-eusflat-15.2015.217
12. Madrid, N., Ramírez-Poussa, E.: Analysis of the φ-index of inclusion restricted to a set of indexes. In: Lecture Notes in Networks and Systems. vol. 1174 LNNS, pp. 3–11. Springer Science and Business Media Deutschland GmbH (2024). https://doi.org/10.1007/978-3-031-74003-9_1
13. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. International Journal of Man-Machine Studies **7**(1), 1–13 (1975). https://doi.org/10.1016/S0020-7373(75)80002-2
14. Novák, V., Perfilieva, I., Močkoř, J.: Mathematical Principles of Fuzzy Logic. Springer US, Boston, MA (1999). https://doi.org/10.1007/978-1-4615-5217-8

15. Pedrycz, W.: Applications of fuzzy relational equations for methods of reasoning in presence of fuzzy data. Fuzzy Sets and Systems **16**(2), 163–175 (Jul 1985). https://doi.org/10.1016/S0165-0114(85)80016-6
16. Sinha, D., Dougherty, E.: Fuzzification of Set Inclusion: Theory and Applications. Fuzzy Sets and Systems **55**(1), 15–42 (1993). https://doi.org/10.1016/0165-0114(93)90299-W
17. Straccia, U.: A fuzzy description logic. In: AAAI/IAAI. pp. 594–599 (1998), https://www.aaai.org/Papers/AAAI/1998/AAAI98-084.pdf
18. Štěpnička, M., Jayaram, B., Su, Y.: A short note on fuzzy relational inference systems. Fuzzy Sets and Systems **338**, 90–96 (2018). https://doi.org/https://doi.org/10.1016/j.fss.2017.08.006

Sustainable Virtual Machine Migration in Cloud Data Centers Using Cuckoo Search Algorithm with Knowledge Acquisition

Francisco J. Maldonado Carrascosa[1*], Sebastián García Galán[1], J. Enrique Muñoz Expósito[1], Juan de la Torre Cruz[1], Francisco J. Cañadas Quesada[1], and Małgorzata Szczerska[2]

[1] University of Jaén, Telecommunication Engineering Department, Linares, Spain
fjmaldon@ujaen.es, sgalan@ujaen.es, jemunoz@ujaen.es, jtorre@ujaen.es, fcanadas@ujaen.es
[2] Gdańsk University of Technology, Gdańsk, Poland
malszcze@pg.edu.pl

Abstract. In the context of optimizing cloud sustainability using virtual machine migration across cloud data centers, this paper explores the integration of knowledge acquisition with the Cuckoo Search Algorithm. Given the increasing demand for efficient and sustainable resource allocation in cloud environments, the study examines how knowledge acquisition mechanisms can enhance the system performance using a bio-inspired algorithm to optimize energy sustainability in this domain. Specifically, the paper compares the effectiveness of the Cuckoo Search Algorithm augmented with knowledge acquisition against the traditional Pittsburgh Algorithm and Knowledge Acquisition with a Swarm Intelligence Approach for Virtual Machine (VM) migration tasks seeking a higher percentage of renewable energy over total energy consumed by the cloud environment. Through a series of experiments conducted in simulated cloud scenarios using a CloudSim-based simulator, the paper evaluates the proposed method's ability to achieve higher renewable energy rates and reduced overall energy consumption. The findings highlight the potential of incorporating knowledge-driven enhancements into an evolutionary algorithm such as Cuckoo Search Algorithm to tackle the complex optimization challenges in cloud computing, particularly regarding VM migration.

Keywords: Knowledge acquisition · VM migration · Cuckoo Search Algorithm · Sustainability · Cloud Computing .

1 Introduction

Cloud computing has revolutionized the way organizations manage and utilize computational resources, offering scalability, flexibility, and cost-effectiveness.

* Corresponding author.

Central to the efficient operation of cloud infrastructures is the dynamic migration of virtual machines across data centers to optimize resource utilization, enhance performance, and minimize operational costs [1]. However, the process of Virtual Machine (VM) migration presents complex optimization challenges, including the need to balance factors such as resource availability, network latency, and energy consumption [2, 3].

In recent years, nature-inspired optimization algorithms have emerged as powerful tools for addressing such challenges [4–6]. Among these, the Cuckoo Search Algorithm (CSA) has garnered attention for its ability to efficiently navigate complex search spaces and find optimal solutions [7]. CSA, inspired by the brood parasitism of some cuckoo species, mimics the reproductive behavior of these birds to iteratively refine candidate solutions through a process of local search and randomization.

To further improve the efficacy of CSA in the context of VM migration, this study proposes the integration of knowledge acquisition mechanisms, aimed at enhancing the algorithm's ability to exploit domain-specific knowledge and adapt to dynamic cloud environments [8]. Knowledge acquisition techniques encompass various methods for acquiring, representing, and utilizing domain knowledge, including expert systems, machine learning, and data mining. By leveraging domain knowledge, CSA can better navigate the search space, identify promising regions, and expedite convergence towards high-quality solutions.

The application of a knowledge acquisition based on an expert system to CSA for optimizing VM migration across cloud data centers is investigated in this paper. Specifically, how leveraging cloud knowledge can augment CSA's performance compared to conventional approaches, such as the Pittsburgh Algorithm or Knowledge Acquisition with a Swarm Intelligence Approach (KASIA) [9], is explored. The Pittsburgh Algorithm, a traditional heuristic-based approach [10], has been widely used in cloud environments but may struggle to adapt to changing conditions and exploit domain-specific knowledge effectively [11, 12].

The effectiveness of the proposed method in optimizing the renewable energy percentage of the cloud environment is assessed through empirical evaluations conducted in simulated cloud-based environments [13, 14]. By comparing the performance of knowledge-aided CSA against traditional approaches, the potential of knowledge-driven optimizations for enhancing the renewable energy percentage of VM migration in cloud data centers is aimed to be demonstrated.

The findings of this research contribute to advancing the state-of-the-art in cloud resource management, offering insights into the potential of knowledge-driven optimizations for addressing the complex optimization challenges inherent in VM migration. By harnessing the power of nature-inspired algorithms and domain-specific knowledge, efforts are made to facilitate more efficient and intelligent VM migration strategies, ultimately enhancing cloud sustainability and performance.

The subsequent section presents the CSA with Knowledge Acquisition, aimed at optimizing virtual machine migration across cloud data centers. In this approach, domain-specific knowledge is encoded through a fuzzy expert system to

enhance performance in addressing the complex optimization inherent in cloud computing environments. In Section 3, the proposed model is compared against a traditional approach, such as the Pittsburgh Algorithm and KASIA, to evaluate its effectiveness in achieving superior migration outcomes regarding sustainability. Finally, conclusions are drawn, and potential avenues for future research in this domain are outlined.

2 Cuckoo Search Algorithm with Knowledge Acquisition

The CSA is a nature-inspired optimization algorithm that simulates the reproductive behavior of cuckoo species to iteratively refine candidate solutions. Developed by Yang and Deb in 2009 [7], CSA has gained popularity for its simplicity, efficiency, and ability to find near-optimal solutions in complex search spaces. The algorithm is based on the obligate brood parasitism behavior exhibited by some cuckoo species, where they lay their eggs in the nests of other host bird species, relying on the hosts to incubate their eggs and rear their offspring [15, 16].

In the context of optimization, CSA operates on a population of candidate solutions represented as nests. Each nest X_i corresponds to a potential solution to the optimization problem, and the quality of a nest is evaluated based on an objective function, in this paper renewable energy percentage over total energy consumed by the cloud environment. During each iteration, the algorithm iteratively updates the nests by performing a combination of local search and randomization. The global exploration is facilitated by the randomization process, which mimics the random selection of host nests by cuckoos, while the local search mechanism enables the algorithm to exploit promising regions of the search space using Lévy flight, which is a key component of the local search in CSA: a random walk characterized by step lengths that follow a heavy-tailed probability distribution. Lévy flights enable the algorithm to make large jumps, which helps in exploring the search space more effectively and escaping local optima. The mathematical representation of the update process for a nest X_i is defined as follows:

In the context of multi-objective optimization, CSA operates over a population matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ representing m candidate nests in an n-dimensional search space:

$$\mathbf{X} = \left[X_1^T \; X_2^T \; \cdots \; X_m^T\right]^T \tag{1}$$

where each row vector $X_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,n}]$ encodes a specific virtual machine placement layout. The evolutionary optimization trajectory balances localized exploitation and randomized global exploration.

Localized search is governed by non-Gaussian random walks whose step lengths follow a heavy-tailed power-law distribution. The position update of the i-th nest at generation $t+1$ is modeled as:

$$X_i^{(t+1)} = X_i^{(t)} + \alpha \cdot \mathcal{L}(\beta) \odot (X_{best} - X_i^{(t)}) \tag{2}$$

where $\alpha > 0$ represents the dynamic step size scaling factor, $\odot$ denotes the Hadamard entry-wise product, and X_{best} is the current global optimum vector. The stochastic step sequence $\mathcal{L}(\beta)$ is drawn from a stable symmetric Lévy distribution via Mantegna's algorithm:

$$\mathcal{L}(\beta) \sim \frac{u}{|v|^{1/\beta}}, \quad u \sim \mathcal{N}(0, \sigma_u^2), \quad v \sim \mathcal{N}(0, 1) \tag{3}$$

where the distribution parameters obey $1 < \beta \leq 2$ (typically set to $\beta = 1.5$), and the standard deviation σ_u is defined through Euler's gamma functions:

$$\sigma_u = \left[\frac{\Gamma(1+\beta) \cdot \sin(\pi\beta/2)}{\Gamma\left(\frac{1+\beta}{2}\right) \cdot \beta \cdot 2^{(\beta-1)/2}} \right]^{1/\beta} \tag{4}$$

This heavy-tailed distribution allows $\mathcal{L}(\beta)$ to yield occasional large jumps, preventing the population from trapping in local minima during host suitability mapping.

Conversely, global exploration models the host bird's awareness, where a fraction $p_a \in [0, 1]$ of sub-optimal nests are discovered and abandoned. The replacement mechanism introduces structured diversification:

$$X_i^{(t+1)} = X_i^{(t)} + H(p_a - \epsilon) \cdot r \cdot (X_j^{(t)} - X_k^{(t)}) \tag{5}$$

where $H(\cdot)$ is the Heaviside step function, $\epsilon, r \sim \mathcal{U}(0, 1)$ are uniform random numbers, and $j, k \in \{1, \ldots, m\}$ are mutually exclusive random nest indices ($j \neq k \neq i$).

The fitness assessment maps the algorithmic boundaries to cloud sustainability goals. The objective function evaluates the environmental efficiency by maximizing the renewable energy share over the aggregated consumption profile:

$$\max_{X_i} \text{Fitness}(X_i) = \frac{\sum_\tau E_{\text{ren}}(X_i, \tau)}{\sum_\tau E_{\text{total}}(X_i, \tau)} \tag{6}$$

where E_{ren} and E_{total} represent the renewable energy utilized and total energy consumed by the infrastructure during the scheduling interval τ, respectively.

To enhance the performance of CSA for VM migration in cloud data centers, the proposal is the integration of a knowledge acquisition mechanism: Cuckoo Search Algorithm with Knowledge Acquisition (CSAKA). Knowledge acquisition encompasses various techniques for acquiring, representing, and utilizing domain-specific knowledge to guide the optimization process. In the context of CSA, knowledge acquisition can be employed to provide insights into the characteristics of the optimization problem, such as the features related to cloud data centers, hosts and virtual machines as shown in Table 1.

One approach to incorporating knowledge acquisition into the CSA is through the utilization of expert systems, which encode domain expertise into a set of rules or heuristics [17, 18]. These rules serve as the foundation for guiding the

algorithm's behavior by leveraging insights from expert knowledge about the problem domain. In this sense, in the context of optimizing cloud sustainability through virtual machine migration, a fuzzy expert system can be employed. Following this idea, in this paper, a fuzzy expert system utilizes rule bases to represent each "cuckoo's nest" in CSA, with rules constructed based on features extracted from the cloud system description (Table 1).

Table 1. Features for cloud system description.

Variable	Description
Renewable Availability (CDC-RA)	Renewable energy supplied to the CDC
Host Computational Capacity (HCC)	Maximum computational capacity (CC) in MIPS
Host Computational Availability (HCA)	Remaining CC of the host in MIPS after holding other VMs
VM Maximum Computational Needs (VM-MCN)	Maximum needs of the VM in MIPS
VM Current Computational Needs (VM-CCN)	Remaining computational needs in MIPS for the VM

The features outlined in Table 1 provide essential information about the cloud environment, such as renewable energy availability, host computational capacity, and VM computational needs. By incorporating these features into the fuzzy expert system's knowledge base, CSA can make informed decisions during the optimization process. For example, knowledge about renewable energy availability (CDC-RA) can guide CSA to prioritize VM migration to data centers with higher renewable energy supply, thereby enhancing overall sustainability. Similarly, information about host computational availability (HCA) and VM computational needs (VM-CCN) can inform CSA's decision-making regarding VM placement and migration strategies.

This paper utilizes an expert system whose knowledge bases consist of seven rules, adopting an identical rule structure from a previous work [2] to maintain consistency and leverage its proven effectiveness in addressing the problem. The anatomy of the rules within the KBs is illustrated in Table 2, comprising eight variables including five input variables (cloud features), the output of the Fuzzy Rule-Based System (host suitability for migrating the virtual machine), the weight variable, and the logical operator used in the rule. These variables are constrained by specific upper and lower bounds to ensure stability. During the initialization phase, the knowledge bases are randomly created while adhering to the universe of discourse for both input and output variables. Input variables have three possible values along with a complementary operator, resulting in six distinct values per input, while the output variable encompasses five different values and their complementary counterparts, totaling ten possible output values. The weight remains constant at 1 in all rules, indicating equal impor-

tance, and the logical operators for combining input values include AND and OR, allowing for different rule combinations. As additional information, the expert system uses fuzzy Gaussian membership functions. An example of an expert system rule base extracted from the cloud environment is presented in Table 3.

Table 2. Fuzzy rule anatomy.

Parameter	Value/Description
Number of Variables	8
Upper Bounds	[3, 3, 3, 3, 3, 5, 1, 2]
Lower Bounds	[1, 1, 1, 1, 1, 1, 1, 1]
Rules per Knowledge Base (KB)	7
Rule Anatomy	Inputs (5), Output, Weight, and Logical Operator

Table 3. Rules base of the expert system.

Rule	CDC-RA	HCC	HCA	VM-MCN	VM-CCN	AND/OR	OUTPUT
1		High		Low		AND	Extremely Suitable
2		Low		High		AND	Not Very Suitable
3			High		Low	AND	Extremely Suitable
4			Low		High	AND	Not Very Suitable
5	Low						Not Very Suitable
6	Medium						Suitable
7	High						Extremely Suitable

Incorporating knowledge acquisition into CSA offers several potential benefits for optimizing VM migration in cloud data centers. By leveraging cloud-based information, CSA can make more informed decisions, leading to improved convergence speed, higher-quality solutions, and enhanced overall system performance. Moreover, knowledge-driven optimizations can enable CSA to adapt dynamically to changing conditions and uncertainties in the cloud environment, making it more robust and reliable in future practical deployment scenarios.

3 Experimental Results

To evaluate the performance of CSAKA for VM migration in cloud data centers, a series of experiments in medium-scale scenarios regarding the hosts and virtual machines, and with varying the number of cloudlets were conducted. The experiments were designed to assess CSAKA's effectiveness in optimizing renewable energy compared to the Pittsburgh Algorithm and KASIA, conventional approaches commonly used for VM migration in cloud environments.

For each experiment, a geographically distributed cloud data center environment using an extended version of CloudSim [13] was simulated. The experiments

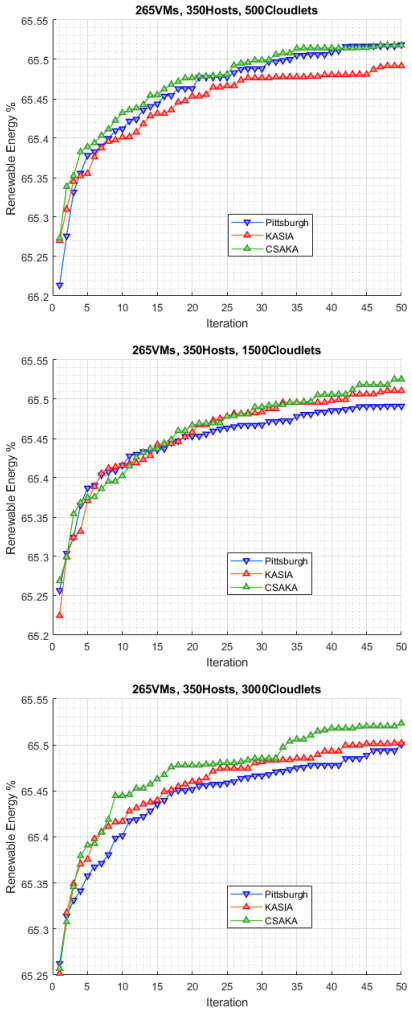


Fig. 1. Performance evolution of Pittsburgh, KASIA and CSAKA in the three scenarios.

were conducted on a server HP ProLiant ML110 G5 (Xeon 3075, dual-core, 2.6 GHz, 4 GB RAM, and 1Gbps of Bandwidth). The simulator enables the simulation of various broker strategies using knowledge bases generated by the knowledge acquisition algorithm. It also accommodates multiple resource architectures and diverse cloud data center builds. The variations encompass different host and VM configurations, as well as various selection and migration policies [3].

For the experiments, three distinct scenarios to assess the performance of the proposed CSAKA were developed in comparison to the Pittsburgh Algorithm and KASIA for renewable energy optimization. Each scenario, counting with 4 geographically distributed cloud data centers, varied in the number of hosts and virtual machines, with a focus on different cloudlet capacities. Specifically, the simulation scenarios encompass 530 hosts and 695 virtual machines, wherein the number of cloudlets varied as follows:

- Scenario 1: Small-scale scenario with 1000 cloudlets.
- Scenario 2: Medium-scale scenario with 2000 cloudlets.
- Scenario 3: Large-scale scenario with 5000 cloudlets.

To evaluate the effectiveness of CSAKA, a comparison is made against the Pittsburgh Algorithm and KASIA in the context of renewable energy optimization. The parameter configuration for both algorithms involved setting 64 individuals, 50 iterations, and 30 repetitions for each simulation run. This setup ensured a comprehensive assessment of the algorithms' performance across multiple iterations and repetitions.

The primary performance metric analyzed in the experiments was renewable energy optimization. By comparing the performance of CSAKA, KASIA and Pittsburgh in terms of their ability to optimize renewable energy usage across the different scenarios, the aim was to assess the efficacy of CSAKA in addressing the complex optimization challenges inherent in cloud computing environments.

From the experimental results shown in Figure 1, it is evident that CSAKA outperforms the Pittsburgh Algorithm and KASIA in all three scenarios. CSAKA demonstrates higher renewable energy rates compared to Pittsburgh Algorithm across different scales of cloud data center environments. This superiority of CSAKA is further corroborated by Table 4, which presents a comparison between Pittsburgh, KASIA and CSAKA in terms of the percentage of renewable energy used. As shown in the table, CSAKA consistently achieves a higher percentage of renewable energy utilization compared to Pittsburgh and KASIA across various configurations of hosts, VMs, and cloudlets. These findings underscore the effectiveness of incorporating knowledge acquisition mechanisms into evolutionary algorithms like CSA for optimizing renewable energy usage in cloud computing environments.

The superior performance of CSAKA can be attributed to its ability to leverage domain-specific knowledge and adapt dynamically to changing conditions in the cloud environment. By incorporating the knowledge acquisition mechanism, CSAKA effectively exploits insights from cloud features to guide the optimization process, resulting in more efficient VM migration strategies.

Table 4. Results comparison between Pittsburgh, KASIA and CSAKA: Percentage of renewable energy used.

Hosts	VMs	Cloudlets	Pittsburgh	KASIA	CSAKA
530	695	1,000	66.70±0.02	66.65±0.02	66.72±0.01
530	695	2,000	66.70±0.02	66.71±0.02	66.72±0.01
530	695	5,000	66.68±0.01	66.67±0.02	66.71±0.01

Overall, the experimental results validate the efficacy of CSAKA for optimizing VM migration in cloud data centers and underscore the importance of knowledge-driven optimizations in enhancing the percentage of renewable energy of cloud computing infrastructures.

3.1 Discussion of Results

The choice of the CSA for the VM migration problem is motivated by its superior ability to maintain a balance between local exploitation and global exploration compared to traditional methods. Unlike the Pittsburgh Algorithm or KASIA, which are often prone to premature convergence in high-dimensional search spaces, Lévy flights are leveraged by CSA to generate a wider, more effective search radius. This mechanism allows the algorithm to escape local optima, a common pitfall encountered in deterministic approaches used in earlier studies. While other metaheuristics, such as Particle Swarm Optimization or Ant Colony Optimization, could theoretically be applied, a lower computational overhead has been demonstrated by CSA in dynamic environments, making it uniquely suited for the energy-intensive demands of cloud data centers.

Regarding the comparison with KASIA and the Pittsburgh Algorithm, a novel fuzzy expert system layer is introduced by the proposed CSAKA approach, which dynamically adjusts the search process based on real-time system states. While numerical improvements over KASIA and Pittsburgh may appear marginal, their significance is underscored by the constraints of cloud sustainability; in large-scale data centers, a marginal increase in renewable energy utilization translates into substantial reductions in carbon emissions and operational costs. These results indicate that a more robust framework for long-term sustainability is provided by CSAKA than by the static structures found in traditional KASIA and Pittsburgh deployments.

The parameters utilized for this study were determined through a sensitivity analysis conducted during the preliminary simulation phases. A population size was selected to ensure diversity across the search space without exceeding the memory limits of the CloudSim environment. Furthermore, these parameters were optimized to ensure that convergence is achieved within a reasonable number of iterations, balancing precision with the execution speed required for real-time VM migration scenarios.

4 Conclusion and Future Work

This paper presented the integration of a knowledge acquisition mechanism based on an expert system with the CSA algorithm to enhance virtual machine migration efficiency and sustainability in cloud data centers. The proposed CSAKA leverages domain-specific knowledge and insights derived from a fuzzy rule-based expert system to improve the optimization process. By incorporating these knowledge-driven elements, CSAKA demonstrated superior performance compared to the traditional Pittsburgh Algorithm and KASIA across various medium-scale scenarios regarding the optimization of renewable energy. The experimental results, conducted in simulated CloudSim environments with different numbers of hosts, virtual machines, and cloudlets, consistently showed that CSAKA outperforms Pittsburgh and KASIA in terms of renewable energy. The significant improvements in this key performance metric validate the efficacy of knowledge acquisition in enhancing the CSA's ability to navigate the complex search space of VM migration problems.

Future work will focus on further refining the knowledge acquisition techniques and exploring their applicability to other optimization problems within cloud computing. Furthermore, the scalability of CSAKA will be investigated in larger and more diverse cloud environments, aiming to ensure its practical applicability and effectiveness in real-world cloud data centers. By continuing to enhance the synergy between evolutionary algorithms and domain-specific knowledge, efforts are made to contribute to the development of more intelligent, sustainable and efficient cloud computing infrastructures.

Acknowledgment

This work has been supported by the ARTERiA Research Project PID2023-146520OB-C21 funded by the Spanish Government. Additionally, this study was performed within the framework of COST Action CA21159 (Understanding interaction light - biological surfaces possibility for new electronic materials and devices - PhoBioS), supported by COST (European Cooperation in Science and Technology; www.cost.eu).

References

1. M. Singh, G. S. Panesar, and S. Taneja, "Optimization, and future prospects of live virtual machine migration in cloud computing: A comprehensive survey on techniques, challenges, and emerging directions," in *Proceedings of the 3rd International Conference on Artificial Intelligence and Machine Learning Applications (AIMLA)*, 2025, pp. 1–6, doi: 10.1109/AIMLA63829.2025.11040408.
2. D. Seddiki, S. G. Galán, E. M. Expósito, M. V. Ibáñez, T. Marciniak, and R. J. P. De Prado, "Sustainability-based framework for virtual machines migration among cloud data centers," in *Proceedings of the 2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS)*, 2021, pp. 1–8.

3. D. Seddiki, S. G. Galán, J. E. M. Expósito, M. V. Ibáñez, T. Marciniak, R. J. P. De Prado, "Sustainable expert virtual machine migration in dynamic clouds," *Computers and Electrical Engineering*, vol. 102, p. 108257, 2022.
4. E. S. M. El-kenawy, N. Khodadadi, S. Mirjalili, A. M. Zaki, A. Ibrahim, A. A. Alhussan, et al., "Glider snake optimizer (GSO): a nature-inspired metaheuristic algorithm for global and engineering optimization problems," *Artificial Intelligence Review*, vol. 59, no. 3, p. 91, 2026, doi: 10.1007/s10462-026-11504-x.
5. F. S. Prity, "Nature-inspired optimization algorithms for enhanced load balancing in cloud computing: a comprehensive review with taxonomy, comparative analysis, and future trends," *Swarm and Evolutionary Computation*, vol. 97, p. 102053, 2025, doi: 10.1016/j.swevo.2025.102053.
6. R. Masadeh, O. Almomani, A. Zaqebah, S. Masadeh, K. Alshqurat, A. Sharieh, and N. Alsharman, "Narwhal optimizer: a nature-inspired optimization algorithm for solving complex optimization problems," *Computers, Materials, & Continua*, vol. 85, no. 2, p. 3709, 2025, doi: 10.32604/cmc.2025.066797.
7. X. S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proceedings of the World Congress on Nature & Biologically Inspired Computing*, 2009, pp. 210–214.
8. S. García-Galán, R. P. Prado, and J. E. M. Expósito, "Swarm fuzzy systems: knowledge acquisition in fuzzy systems and its applications in grid computing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 7, pp. 1791–1804, 2013.
9. R. P. Prado, S. G. Galán, A. J. Yuste, J. M. Expósito, A. S. Santiago, and S. Bruque, "Evolutionary fuzzy scheduler for grid computing," in *Bio-Inspired Systems: Computational and Ambient Intelligence*, 2009, pp. 286–293.
10. S. F. Smith, "A learning system based on genetic adaptive algorithms," Ph.D. dissertation, Dept. Comput. Sci., Univ. Pittsburgh, Pittsburgh, PA, USA, 1980.
11. F. J. M. Carrascosa, S. G. Galán, M. V. Ibáñez, M. Szczerska, T. Marciniak, and N. R. Reyes, "Fuzzy game-theoretic optimization for sustainable cloud computing," *Engineering Applications of Artificial Intelligence*, vol. 162, p. 112725, 2025, doi: 10.1016/j.engappai.2025.112725.
12. F. J. M. Carrascosa, A. J. Sánchez, D. Seddiki, S. G. Galán, M. V. Ibáñez, and T. Marciniak, "Knowledge acquisition approaches for virtual machine migration in cloud computing," *Computing and Informatics*, vol. 43, no. 6, pp. 1285–1319, 2024, doi: 10.31577/cai_2024_6_1285.
13. R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011.
14. A. Verma, G. Verma, S. Goswami, and H. Shukla, "A study on cloudsim framework for cloud simulation environments," *DMPedia Lecture Notes in Multidisciplinary Research*, pp. 1109–1122, 2026.
15. S. N. Makhadmeh, M. A. Awadallah, S. Kassaymeh, M. A. Al-Betar, Y. Sanjalawe, S. Kouka, and A. Al-Redhaei, "Recent advances in multi-objective cuckoo search algorithm, its variants and applications," *Archives of Computational Methods in Engineering*, vol. 32, no. 5, pp. 3213–3240, 2025, doi: 10.1007/s11831-025-10240-9.
16. S. M. Almufti, R. B. Marqas, R. R. Asaad, and A. A. Shaban, "Cuckoo search algorithm: overview, modifications, and applications," *International Journal of Scientific World*, vol. 11, no. 1, pp. 1–9, 2025.
17. Z. E. Mohamed and W. Dabour, "An enhanced adaptive elephant herding optimization based on hybrid cuckoo search algorithm and elite opposition-based learning," *Scientific Reports*, vol. 16, no. 1, p. 15221, 2026, doi: 10.1038/s41598-026-48615-y.

18. M. Cherukuri, K. G. Devi, K. C. Santosh, Y. Farooqui, J. Somlal, and M. M. Kulkarni, "Advancement in next-gen medical intelligence: Fuzzy logic-driven expert systems for clinical decision-making," *International Journal of Drug Delivery Technology*, vol. 16, no. 20s, p. 97, 2026, doi: 10.25258/ijddt.16.20s.97.

A Fuzzy Linguistic Approach to Functional Adaptability Evaluation in Industrial Forecasting Systems

Francisco P. Romero[1,2][0000−0002−6993−2434], Jesus R. Oviedo[1,2][0000−0001−7962−1042], Jared Guerrero-Sosa[1][0000−0001−7999−9870], Moises Rodriguez[1,2][0000−0003−2155−7409], Andres Montoro-Montarroso[1,2][0000−0003−1893−3346], Javier Verdugo[1,2][0000−0002−2526−2918], Jesus Serrano-Guerrero[1,2][0000−0002−6177−8188], Mario Piattini[1,2][0000−0002−7212−8279], and Jose A. Olivas[1,2][0000−0003−1893−3346]

[1] Department of Information Technologies and Systems, University of Castilla La Mancha, Ciudad Real, Spain
franciscop.romero@uclm.es
[2] Intelligent Information Systems Certification, Ciudad Real, Spain
jverdugo@i2sc.es

Abstract. Artificial Intelligence systems deployed in industrial forecasting environments are exposed to temporal drift, seasonality and evolving operational conditions. ISO/IEC 25059 defines functional adaptability as an AI quality subcharacteristic, but it does not specify how to assess whether a deployed system effectively learns from new information and maintains or improves predictive behaviour over time. This proposal addresses that gap through an operational, temporal-error-based assessment, using indicators such as temporal-error trends, degradation ratios, recovery gains and stability measures. Since the interpretation of these indicators is uncertain, context-dependent and often expressed by experts through linguistic judgements such as low degradation, acceptable delay or high stability, the proposal incorporates a fuzzy linguistic model. It defines fuzzy variables, expert-based membership functions and ISO/IEC 25059-aligned aggregation rules. The model provides an interpretable fuzzy linguistic assessment, complemented by a defuzzified score for operational monitoring and comparison.

Keywords: Artificial Intelligence Quality · ISO/IEC 25059 · Functional Adaptability · Fuzzy Logic · Industrial Forecasting

1 Introduction

Artificial Intelligence (AI) systems are increasingly embedded in industrial decision making processes. In these settings, model quality cannot be assessed only through static predictive metrics computed at development time. Once deployed, these systems are exposed to changing data distributions, seasonality, shifts in

user behaviour, and evolving operational constraints. Therefore, their ability to maintain or improve performance over time is a key quality requirement.

ISO/IEC 25059 [1] extends the SQuaRE quality model [2] to AI systems and introduces AI-specific quality subcharacteristics. Among them, functional adaptability refers to the extent to which an AI system can acquire information from data or past actions and use it to inform future predictions or behaviours. This property is especially relevant for forecasting systems that are periodically retrained with updated operational data. However, ISO/IEC 25059 provides a conceptual definition and does not prescribe a concrete measurement procedure for real industrial deployments. Recent work has shown that ISO/IEC 25059 can be operationalised in real AI quality evaluations, highlighting the importance of acceptable specification limits, representative test cases, and context-dependent assessment criteria, especially in safety-critical applications [3].

A previous operational approach evaluates functional adaptability by analysing the temporal evolution of forecasting error across successive operational executions. Under this view, adaptation is evidenced when retraining with new operational information preserves or improves predictive performance [4, 6], consistent with time-adaptive forecasting and concept-drift adaptation methods. Error degradation, recovery gain, and post-adaptation stability are useful indicators, although their interpretation remains context-dependent because acceptable degradation and detection delay vary with the operational environment and forecast horizon [7].

Existing work relevant to this proposal can be organised into three complementary topics: 1) Concept-drift and time-adaptive forecasting studies focused on detecting distributional changes and maintaining predictive performance [8]; 2) Production-oriented machine learning frameworks that emphasise data validation, monitoring, retraining, infrastructure reliability, and technical readiness [9] and; 3) fuzzy linguistic models provide mechanisms for representing gradual and expert dependent judgements [5]. However, these strands do not jointly provide an ISO/IEC 25059-aligned procedure that transforms temporal adaptation evidence into an auditable linguistic quality assessment. Accordingly, this paper reformulates functional adaptability as a fuzzy linguistic assessment problem, by integrating temporal adaptability indicators, expert knowledge, and interpretable linguistic outputs within an ISO/IEC 25059-aligned framework.

The remainder of the paper is organised as follows. Section 2 introduces functional adaptability in ISO/IEC 25059 and its temporal, uncertainty-aware assessment. Section 3 presents the proposed fuzzy linguistic model, including indicators, membership functions, rule-based aggregation, and linguistic output. Section 4 illustrates its application through an industrial predictive maintenance scenario. Section 5 discusses the contributions, limitations, and validation requirements of the proposal. Finally, Section 6 presents the conclusions and future work.

2 Functional Adaptability in Industrial Forecasting Systems

Industrial forecasting systems commonly operate in batch or periodically retrained settings. Unlike online learning systems, these solutions do not update their internal parameters continuously after each new observation. Instead, they are retrained at predefined intervals or when sufficient new operational data becomes available. Each retraining process produces a new operational state of the system, whose predictive behaviour can be evaluated in subsequent forecasting windows. This operational view is consistent with the broader literature on production machine learning, where system quality depends on model accuracy, data validation, monitoring, retraining practices, infrastructure reliability and technical-debt control [9, 10]. In this sense, functional adaptability should be assessed through observable behaviour during operation, rather than inferred only from the existence of a retraining pipeline.

In industrial environments, this issue is particularly relevant because forecasting systems are rarely exposed to stationary data-generating processes. Demand patterns, production regimes, customer behaviour, machine conditions and external factors may evolve over time, leading to changes in the relationship between input variables and target values. This phenomenon is closely related to concept drift, which has been extensively studied in machine learning as a central challenge for models deployed under evolving distributions [8]. In time-dependent prediction problems, concept drift is especially critical because the temporal order of observations determines both the training context and the future operational conditions under which the model is evaluated.

Let $t = 1, \ldots, T$ denote ordered operational executions or evaluation windows. For each execution, the forecasting system generates predictions $\hat{y}_{i,t}$ for observed values $y_{i,t}$. Forecasting error may be measured using application dependent metrics such as RMSE, sMAPE or WAPE. In industrial settings with heterogeneous demand volumes, WAPE is particularly useful because it expresses the aggregate absolute deviation relative to the observed magnitude:

$$WAPE_t = \frac{\sum_i |y_{i,t} - \hat{y}_{i,t}|}{\sum_i |y_{i,t}|}. \tag{1}$$

The ordered sequence $E = \{E_1, E_2, \ldots, E_T\}$, where E_t denotes the error in execution t, provides the empirical basis for analysing functional adaptability. In the original operational proposal, the temporal behaviour of this sequence is examined through trend and stability analysis. A simple linear trend can be estimated as:

$$E_t = \beta_0 + \beta_1 t + \epsilon_t, \tag{2}$$

where $\beta_1 < 0$ suggests improvement, $\beta_1 \approx 0$ suggests stability, and $\beta_1 > 0$ suggests degradation. This analysis can be complemented with hypothesis testing, non-parametric trend tests, bootstrap confidence intervals or segmented analysis by forecasting horizon, region, product family or business context. Such

temporal analysis aligns with the idea that functional adaptability, as introduced in ISO/IEC 25059, should reflect an AI system's ability to acquire information from data or previous actions and use it to inform future predictions.

However, functional adaptability cannot be reduced to a regression coefficient. A deployed AI system may temporarily degrade due to drift, recover after retraining and then remain stable. Therefore, we consider four complementary dimensions: performance degradation, drift detection delay, recovery gain and post-adaptation stability. These dimensions also connect functional adaptability with production-readiness concerns: the system must detect that predictive behaviour has changed and support recovery actions and demonstrate stable post-adaptation behaviour [10].

Let t_c denote the change point at which the operational environment changes, t_d the window in which drift is detected, and t_a the window following an adaptation action, such as retraining. Let E_{base}, E_{drift} and E_{post} represent the mean errors in the baseline, degradation and post-adaptation periods, respectively. The following indicators are then defined:

$$D = \frac{E_{drift} - E_{base}}{E_{base}}, \quad \Delta_{det} = t_d - t_c, \quad G = \frac{E_{drift} - E_{post}}{E_{drift} - E_{base}}. \tag{3}$$

Post-adaptation stability can be estimated as the variability of the error after adaptation:

$$S = \sqrt{\frac{1}{K-1} \sum_{j=0}^{K-1} \left(E_{t_a+j} - \overline{E}_{post}\right)^2}. \tag{4}$$

where $\overline{E}_{post}$ denotes the mean prediction error over the K post-adaptation windows:

$$\overline{E}_{post} = \frac{1}{K} \sum_{j=0}^{K-1} E_{t_a+j}. \tag{5}$$

Lower degradation, shorter detection delay, higher recovery gain, and lower post-adaptation instability indicate better functional adaptability. However, indicators like these cannot be interpreted using universal crisp thresholds, since their acceptability depends on the business context, forecasting horizon, error costs and the volatility of the operational environment. For instance, a small degradation may be acceptable in aggregate demand forecasting but critical in predictive maintenance or safety-sensitive monitoring. Therefore, the proposed fuzzy model preserves the original temporal-error-based assessment and extends it with a linguistic interpretation layer. Degradation, detection delay, recovery gain and post-adaptation stability are still computed from the temporal evolution of prediction error, but they are subsequently interpreted and aggregated under uncertainty, expert judgement and domain-dependent criteria using labels such as *low degradation*, *acceptable delay*, *high recovery* and *stable post-adaptation behaviour*.

3 Fuzzy Linguistic Functional Adaptability Model

The proposed model interprets functional adaptability as a fuzzy quality property derived from observable temporal behaviour. Instead of imposing crisp thresholds, the model represents each adaptability dimension through linguistic variables and expert-defined membership functions. This design reflects the fact that AI quality evaluation often involves gradual transitions between acceptable and unacceptable behaviour.

3.1 Input and Output Linguistic Variables

The fuzzy assessment model uses four input variables: degradation D, detection delay Δ_{det}, recovery gain G, and post-adaptation instability S. The output variable represents functional adaptability, denoted as FA, through the linguistic labels *Very Low*, *Low*, *Moderate*, *High*, and *Very High*. A scalar value, denoted as FA_{score}, is obtained only after defuzzification as a complementary representation for operational monitoring and comparison. Table 1 summarises the proposed linguistic structure.

Table 1. Linguistic variables for fuzzy functional adaptability assessment.

Variable	**Type**	**Linguistic labels**
Performance degradation D	Input	Low, Medium, High
Detection delay Δ_{det}	Input	Fast, Acceptable, Slow
Recovery gain G	Input	Poor, Partial, Good
Post-adaptation instability S	Input	Stable, Moderate, Unstable
Functional adaptability FA	Output	Very Low, Low, Moderate, High, Very High

The proposed labels are intentionally simple in order to provide an more interpretable description. Additional labels may be introduced in domains requiring finer granularity.

3.2 Expert-defined Membership Functions

Each linguistic label is represented through a membership function. Triangular and trapezoidal functions are suitable for an initial industrial deployment because they are interpretable, easy to calibrate and compatible with expert elicitation. For a generic variable x, a triangular membership function can be defined as:

Table 2. Illustrative membership function parameters for normalised indicators.

Variable	Labels	Illustrative membership functions
D	Low; Medium; High	Low: $(0, 0, 0.25)$; Medium: $(0.15, 0.40, 0.65)$; High: $(0.50, 1, 1)$
Δ_{det}	Fast; Acceptable; Slow	Fast: $(0, 0, 0.30)$; Acceptable: $(0.20, 0.45, 0.70)$; Slow: $(0.60, 1, 1)$
G	Poor; Partial; Good	Poor: $(0, 0, 0.35)$; Partial: $(0.25, 0.50, 0.75)$; Good: $(0.65, 1, 1)$
S	Stable; Moderate; Unstable	Stable: $(0, 0, 0.30)$; Moderate: $(0.20, 0.45, 0.70)$; Unstable: $(0.60, 1, 1)$
FA	Very Low; Low; Moderate; High; Very High	Five overlapping fuzzy sets uniformly distributed in $[0, 1]$

$$\mu_A(x; a, b, c) = \begin{cases} 0, & x \leq a, \\ \frac{x-a}{b-a}, & a < x \leq b, \\ \frac{c-x}{c-b}, & b < x < c, \\ 0, & x \geq c. \end{cases} \tag{6}$$

For example, the linguistic variable degradation may be calibrated using domain-specific parameters. In a relatively stable forecasting domain, low degradation may correspond to values close to 0, medium degradation to moderate increases in error, and high degradation to values exceeding an agreed-upon operational tolerance. The precise parameters should be elicited from domain experts, historical performance distributions or service-level requirements.

A possible normalised configuration is shown in Table 2. The values are illustrative and should be adapted to each domain. In this configuration, all input indicators are normalised to $[0, 1]$, except recovery gain, where higher values are better.

This fuzzy formalisation aligns with ISO/IEC 25059 because it assesses the system effectively uses new information to inform future predictions, while avoiding the assumption that adaptability can be judged through universal crisp thresholds.

3.3 Fuzzy Rule Base

The rule base encodes expert knowledge about how the different indicators interact. A Mamdani-type fuzzy inference system [11] is appropriate because it provides transparent linguistic rules and facilitates communication with technical and non-technical stakeholders. The general model can be expressed as:

$$FA = \mathcal{F}(D, \Delta_{det}, G, S), \tag{7}$$

where $\mathcal{F}$ denotes the fuzzy inference process. Table 3 presents examples of rules.

The rule base can be extended or weighted based on the application's criticality. For instance, in short-term operational forecasting, detection delay may be more relevant than long-term stability. In strategic forecasting, post-adaptation stability may receive greater importance.

Table 3. Example fuzzy rules for functional adaptability evaluation.

Rule	**Antecedent**	**Conseq.**
R1	If degradation is Low and recovery is Good and instability is Stable	Very High
R2	If degradation is Low and recovery is Partial and delay is Acceptable	High
R3	If degradation is Medium and recovery is Good and delay is Fast	High
R4	If degradation is Medium and instability is Moderate and recovery is Partial	Moderate
R5	If degradation is High and recovery is Good and instability is Stable	Moderate
R6	If degradation is High and recovery is Poor	Low
R7	If delay is Slow and recovery is Poor	Very Low
R8	If instability is Unstable and degradation is Medium or High	Low

3.4 Aggregation and Defuzzification

The fuzzy inference process produces an aggregated fuzzy output over the linguistic terms defined for functional adaptability, such as *Low*, *Medium*, and *High*. Therefore, the primary result of the model is a linguistic assessment represented by the activation degrees of the output labels.

Let $\mathcal{L}_{FA}$ denote the set of output linguistic terms. The linguistic assessment is represented as

$$\mathcal{A}_{FA} = \{(L_k, \gamma_k) \mid L_k \in \mathcal{L}_{FA}\}, \tag{8}$$

where $\gamma_k \in [0, 1]$ is the activation degree of linguistic label L_k. The dominant assessment is given by

$$L^* = \arg \max_{L_k \in \mathcal{L}_{FA}} \gamma_k. \tag{9}$$

Since several labels may be partially activated simultaneously, the complete linguistic profile is retained. These activation degrees are not probabilities and are therefore not required to sum to one.

For operational monitoring and comparison, the aggregated fuzzy output is also transformed into a scalar value using centroid defuzzification:

$$FA_{\text{score}} = \frac{\int_0^1 z\, \mu_{\widetilde{FA}}(z)\, dz}{\int_0^1 \mu_{\widetilde{FA}}(z)\, dz}. \tag{10}$$

The model reports both a linguistic assessment and a complementary numerical score. For example, a result may be expressed as *Medium functional adaptability, with partial activation of High*, together with its corresponding defuzzified value. The complete linguistic profile constitutes the primary interpretable output of the model, whereas the scalar score facilitates longitudinal monitoring, comparison across models and forecasting horizons, and integration into industrial dashboards. The inference process operates on fuzzy sets and preserves their activation profile, while defuzzification provides only an auxiliary operational representation. Accordingly, the scalar value does not replace the underlying fuzzy linguistic assessment.

Interpretability is operationalised by retaining an assessment trace composed of the normalised indicator values, their membership degrees, the activated rules and their corresponding firing strengths, the aggregated linguistic output, and the complementary defuzzified score. Thus, an evaluator can trace a final assessment back to the observed degradation, detection delay, recovery, and post-adaptation stability values, and identify which expert rules determined the result. The explanation therefore concerns the final linguistic label and the complete path from temporal operational evidence to the resulting quality assessment.

4 Illustrative Industrial Predictive Maintenance Scenario

To illustrate how the proposed framework can be instantiated, we consider a predictive maintenance scenario based on the AI4I 2020 Predictive Maintenance Dataset [12]. The purpose of this section is to define the operational windows, adaptation events, temporal indicators, and fuzzy assessment process that would be used in an empirical application. AI4I is a synthetic benchmark designed to reproduce relevant characteristics of an industrial manufacturing process. It includes air temperature, process temperature, rotational speed, torque, tool wear, machine-failure labels, and specific failure modes, thereby providing a controlled setting in which different operational changes and adaptation situations can be defined.

In this scenario, a maintenance system is periodically retrained to estimate the probability of machine failure in future production windows. Let $W_1, \ldots, W_T$ denote successive operational evaluation windows. For each window W_t, the system receives new sensor observations and produces a failure-risk prediction for each machine instance. Once the actual maintenance outcome is observed, the predictive error is computed using a window-level metric such as the false negative rate, the balanced error rate, or a cost-sensitive failure prediction error. This generates a temporal error sequence:

$$E = \{E_1, E_2, \ldots, E_T\},$$

where E_t represents the predictive error observed in window W_t. In this setting, functional adaptability is the observable capacity of the predictive maintenance system to maintain or recover its predictive behaviour after changes in the operational regime.

The AI4I variables allow the simulation of several industrial adaptation situations. For example, a gradual increase in tool wear may represent progressive machine ageing; changes in torque and rotational speed may indicate modified production regimes; and variations in process temperature may represent changes in operating conditions. These changes may alter the distribution of the input variables and lead to temporal degradation of the predictive maintenance model. Therefore, the system should be evaluated in terms of static classification performance and its temporal behaviour before, during and after adaptation.

Following the original temporal-error-based assessment, four quantitative indicators are computed. First, degradation D measures the relative increase in prediction error after the operational change:

$$D = \frac{E_{drift} - E_{base}}{E_{base}},$$

where E_{base} is the mean error before the change and E_{drift} is the mean error during the degradation phase. Second, detection delay Δ_{det} measures the number of evaluation windows required to detect a relevant change in predictive behaviour:

$$\Delta_{det} = t_d - t_c,$$

where t_c is the estimated change point and t_d is the drift detection point. Third, recovery gain G measures the extent to which the system recovers its previous predictive performance after retraining:

$$G = \frac{E_{drift} - E_{post}}{E_{drift} - E_{base}},$$

where E_{post} is the mean error after the adaptation action. Finally, post-adaptation instability S measures the variability of the error after adaptation:

$$S = \sqrt{\frac{1}{K-1} \sum_{j=0}^{K-1} (E_{t_a+j} - \overline{E}_{post})^2},$$

where t_a is the adaptation point and K is the number of post-adaptation windows considered.

Interpreting these indicators through crisp thresholds is problematic. For instance, an 8% degradation may be considered acceptable in a low-risk production context but critical in a safety-sensitive maintenance scenario. Similarly, a detection delay of three windows may be acceptable when failures evolve slowly, but inadequate when failures are abrupt and costly.

Therefore, the proposed model maps these numerical indicators into fuzzy linguistic variables. The degradation indicator is represented by the linguistic labels *Low*, *Medium*, and *High*. Detection delay is represented as *Fast*, *Acceptable*, and *Slow*; recovery gain as *Poor*, *Partial*, and *Good*; and post-adaptation instability as *Stable*, *Moderate*, and *Unstable*. These labels are defined through expert-based triangular or trapezoidal membership functions, allowing gradual transitions between quality levels.

The fuzzy inference system aggregates these four dimensions into a fuzzy linguistic assessment of functional adaptability, complemented by a defuzzified operational score. The output variable is defined using five linguistic labels: *very low*, *low*, *moderate*, *high* and *very high*. A simplified fragment of the rule base is shown below:

R_1 : IF degradation is low AND recovery is good
THEN functional adaptability is very high.

R_2 : IF degradation is high AND recovery is poor
THEN functional adaptability is low.

R_3 : IF degradation is medium AND detection delay is acceptable
AND recovery is partial THEN functional adaptability is moderate.

R_4 : IF detection delay is slow AND post-adaptation instability is unstable
THEN functional adaptability is very low.

This formulation allows the predictive maintenance system to be assessed in a way that is both quantitative and interpretable. The temporal indicators preserve the original operational assessment of functional adaptability, while the fuzzy layer reflects the uncertainty and context-dependence involved in judging whether the system has adapted adequately. As a result, the model provides an ISO/IEC 25059-aligned mechanism for evaluating whether the predictive maintenance system effectively incorporates new operational information and maintains reliable behaviour under changing industrial conditions.

In this scenario, functional adaptability requires limited degradation under changes in operating conditions, timely detection, substantial recovery after retraining, and stable post-adaptation performance. By contrast, a persistent increase in error, delayed detection, poor recovery, or unstable performance indicates insufficient adaptability. The model reports the dominant linguistic assessment, the activation degrees of the remaining categories, and a complementary defuzzified score. The linguistic profile supports expert interpretation, while the numerical score enables comparisons across models, forecasting horizons, and evaluation periods.

5 Discussion and Conclusions

This paper proposes an ISO/IEC 25059-aligned fuzzy linguistic framework for evaluating functional adaptability in industrial forecasting systems. The ap-

proach preserves the original operational view that adaptability should be assessed through observable temporal predictive behaviour, using degradation, detection delay, recovery gain, and post-adaptation stability. Its main contribution is a fuzzy inference architecture that integrates several temporal indicators into an interpretable assessment framework that combines quantitative evidence with expert knowledge.

The fuzzy model extends crisp threshold-based evaluation by representing the gradual and context-dependent nature of adaptability criteria. Membership functions and linguistic rules make explicit how domain experts interpret acceptable degradation, timely detection, effective recovery, and stable post-adaptation behaviour. The primary output is therefore a fuzzy linguistic assessment, while the defuzzified score is retained only as a complementary representation for longitudinal monitoring, comparison across models and forecasting horizons, and integration into industrial dashboards.

The expected benefit of the proposal is representational and decision-oriented due to it avoids abrupt changes around fixed thresholds, preserves partial satisfaction of quality criteria, and provides a traceable link between temporal indicators, activated rules, and the final assessment. Nevertheless, membership functions may introduce subjectivity, the rule base may contain inconsistencies if it is not carefully validated, and the reliability of the assessment depends on the quality of the underlying drift-detection and temporal-error estimation procedures.

Future work will empirically compare the fuzzy model with a crisp threshold-based baseline using the same temporal indicators. The evaluation will consider agreement with expert assessments, stability near decision boundaries, sensitivity to membership-function parameters, and explanation traceability. Further studies will also examine the transferability of the calibrated model across forecasting horizons and industrial domains, as well as its possible extension to other AI quality subcharacteristics defined in ISO/IEC 25059.

References

1. ISO/IEC: ISO/IEC 25059:2023. Software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Quality model for AI systems. International Organization for Standardization, Geneva (2023)
2. ISO/IEC: ISO/IEC 25010:2023. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Product quality model. International Organization for Standardization, Geneva (2023)
3. Albertini, W., Oviedo, J., Motta, E., Araujo, J., Araujo, M., Rodriguez, M., Lopes, H., Kalinowski, M., Piattini, M.: Practical insights into AI system product quality evaluation. IT Professional **28**(1), 81–87 (2026). https://doi.org/10.1109/MITP.2025.3646118
4. Lang, M.N., Lerch, S., Mayr, G.J., Simon, T., Stauffer, R., Zeileis, A.: Remember the past: A comparison of time-adaptive training schemes for non-homogeneous regression. Nonlinear Processes in Geophysics **27**(1), 23–34 (2020). https://doi.org/10.5194/npg-27-23-2020

5. Herrera, F., Herrera-Viedma, E.: Linguistic decision analysis: Steps for solving decision problems under linguistic information. Fuzzy Sets and Systems **115**(1), 67–82 (2000). https://doi.org/10.1016/S0165-0114(99)00024-X
6. Oliveira, G.H.F.M., Cavalcante, R.C., Cabral, G.G., Minku, L.L., Oliveira, A.L.I.: Time series forecasting in the presence of concept drift: A PSO-based approach. In: 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 239–246. IEEE (2017). https://doi.org/10.1109/ICTAI.2017.00042
7. Zhang, Y., Wen, Q., Chen, W., Sun, L., Zhang, Z., Wang, L., Wang, X., Jin, R., Tan, T.P.: OneNet: Enhancing time series forecasting models under concept drift by online ensembling. In: Advances in Neural Information Processing Systems, vol. 36, pp. 69949–69980 (2023)
8. Gama, J., Zliobaite, I., Bifet, A., Pechenizkiy, M., Bouchachia, A.: A survey on concept drift adaptation. ACM Computing Surveys **46**(4), 1–37 (2014). https://doi.org/10.1145/2523813
9. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.F., Dennison, D.: Hidden technical debt in machine learning systems. In: NeuroIPS 2015, pp. 2503–2511 (2015)
10. Breck, E., Cai, S., Nielsen, E., Salib, M., Sculley, D.: The ML test score: A rubric for ML production readiness and technical debt reduction. In: IEEE International Conference on Big Data, pp. 1123–1132. IEEE (2017). https://doi.org/10.1109/BigData.2017.8258038
11. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. International Journal of Man-Machine Studies **7**(1), 1–13 (1975). https://doi.org/10.1016/S0020-7373(75)80002-2
12. Matzka, S.: Explainable artificial intelligence for predictive maintenance applications. In: 2020 Third International Conference on Artificial Intelligence for Industries (AI4I), pp. 69–74. IEEE (2020). https://doi.org/10.1109/AI4I49448.2020.00023

Modelado del Consenso y el Conflicto en Sistemas Agénticos Basados en Roles para la Verificación de Afirmaciones

Juan Barea-Rojo[1], José Ramón Trillo[2], Juan Carlos González-Quesada[1,3], Francisco Javier Cabrerizo[1,3], and Ignacio Javier Pérez[1,3]

[1] Departamento de Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada, Granada, 18071, España
barearojojuan@gmail.com
[2] Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, Zaragoza, 50018, España
jtrillo@unizar.es
[3] Departamento de Ciencias de la Computación e Inteligencia Artificial, Instituto Andaluz Interuniversitario en Ciencia de Datos e Inteligencia Computacional, DaSCI, Universidad de Granada, Granada, 18071, España
juancarlosgq@ugr.es, cabrerizo@decsai.ugr.es, ijperez@decsai.ugr.es

Resumen Los sistemas agénticos basados en grandes modelos de lenguaje se utilizan cada vez más en tareas de verificación, pero sus predicciones finales suelen obtenerse mediante reglas simples de agregación que no siempre reflejan la fiabilidad de la decisión colectiva. Cuando las salidas de los agentes están muy correlacionadas, el acuerdo puede ser engañoso y la fusión directa puede ocultar información relevante sobre el comportamiento interno del conjunto. Este trabajo propone una arquitectura en dos etapas para la verificación de afirmaciones. Primero, varios agentes especializados por rol evalúan de forma independiente el mismo par afirmación-evidencia y generan juicios estructurados. Después, el comportamiento colectivo se representa mediante señales de consenso, desacuerdo, conflicto, dominancia y confianza, que son procesadas por una capa de decisión supervisada. Los resultados muestran que la agregación directa aporta beneficios limitados, mientras que la arquitectura propuesta produce decisiones finales más eficaces y equilibradas que las alternativas evaluadas. La contribución principal consiste en modelar explícitamente cómo se forman las decisiones colectivas, en lugar de asumir que el consenso es fiable por sí mismo.

Keywords: Sistemas agénticos · Grandes modelos de lenguaje · Verificación de afirmaciones · Toma de decisiones en grupo · Consenso

1. Introducción

Los grandes modelos de lenguaje, o *Large Language Models* (LLM), han impulsado el desarrollo de sistemas agénticos para tareas que requieren razonamiento complejo y coordinación. En estos sistemas, varios agentes pueden asumir roles

distintos para abordar un mismo problema desde perspectivas complementarias. Marcos como AutoGen han mostrado el potencial de los flujos de trabajo agénticos orquestados [19], mientras que CAMEL ha ilustrado cómo la especialización por roles puede inducir comportamientos diferenciados en sociedades basadas en LLM [10]. Sin embargo, aumentar el número de agentes no garantiza una decisión final más fiable. Trabajos recientes han cuestionado que los *prompts* basados en roles o el debate multiagente sean suficientes para mejorar de forma consistente el razonamiento [14], y han mostrado que la colaboración efectiva requiere mecanismos adicionales de coordinación o aprendizaje [12].

En verificación de afirmaciones, esta limitación es especialmente relevante. Muchos sistemas agénticos siguen utilizando reglas simples de agregación, como votación o fusión directa de puntuaciones. No obstante, cuando los agentes comparten sesgos o patrones de error, el consenso puede ser engañoso: varios agentes pueden coincidir no porque la respuesta sea correcta, sino porque fallan de forma correlacionada. Por ello, la decisión final no debería depender solo de qué etiqueta recibe más apoyo, sino también de cómo se forma ese apoyo dentro del conjunto.

Esta idea conecta la verificación agéntica con la toma de decisiones en grupo. Estrategias recientes de agregación han mostrado que explotar información de orden superior puede superar a la votación mayoritaria [1], y que la coordinación estructurada puede mejorar frente a combinaciones agénticas planas [18]. Desde esta perspectiva, la calidad de una decisión colectiva depende no solo del acuerdo, sino también de la estructura del consenso, el conflicto, el desacuerdo, la dominancia y la confianza [15,8].

Este trabajo propone una arquitectura en dos etapas que separa la generación de juicios agénticos de la estimación supervisada de su fiabilidad colectiva. Primero, varios agentes especializados evalúan de forma independiente el mismo par afirmación-evidencia. Después, su comportamiento colectivo se transforma en una representación explícita basada en señales de consenso, desacuerdo, conflicto, dominancia y confianza, que alimenta una capa supervisada de meta-agregación. A diferencia de un esquema clásico de *stacked ensemble learning*, la segunda etapa no se plantea como una simple capa adicional de clasificación, sino como un mecanismo para caracterizar patrones de acuerdo, conflicto y confianza entre agentes. Esta representación puede extenderse a escenarios de verificación continua, en los que la evolución del desacuerdo, la incertidumbre y el conflicto aporte señales adicionales sobre la estabilidad de la decisión.

El resto del artículo se organiza de la siguiente forma. La Sección 2 presenta los fundamentos sobre toma de decisiones en grupo y sistemas agénticos basados en LLM. La Sección 3 describe el marco propuesto. La Sección 4 presenta la evaluación experimental sobre Climate-FEVER. La Sección 5 discute las implicaciones y limitaciones del enfoque propuesto. Finalmente, la Sección 6 recoge las conclusiones y plantea líneas de trabajo futuro.

2. Fundamentos

Esta sección presenta los dos fundamentos del enfoque propuesto: la toma de decisiones en grupo y los sistemas agénticos basados en LLM. La conexión entre ambos permite interpretar las salidas de varios agentes no solo como predicciones independientes, sino como juicios colectivos cuya fiabilidad depende del consenso, el desacuerdo, el conflicto y la confianza.

2.1. Toma de Decisiones en Grupo

La toma de decisiones en grupo, o *Group Decision Making* (GDM), estudia cómo varios decisores pueden alcanzar una decisión colectiva gestionando el desacuerdo. En los modelos basados en consenso, el objetivo no es solo agregar preferencias, sino guiar al grupo hacia un nivel aceptable de acuerdo mediante un proceso de alcance de consenso, incluso cuando las preferencias son heterogéneas [9]. En entornos difusos, los modelos de *soft consensus* sustituyen la unanimidad estricta por grados de acuerdo, permitiendo medir la coherencia colectiva de forma más flexible [8].

Un operador habitual en este contexto es *Ordered Weighted Averaging* (OWA), que agrega valores ordenados asignando distinto peso a cada posición. Dado un conjunto de entradas $x_1, \ldots, x_n$, el operador OWA se define como:

$$\mathrm{OWA}(x_1, \ldots, x_n) = \sum_{i=1}^{n} w_i \, x_{(i)}, \tag{1}$$

donde $x_{(i)}$ representa el i-ésimo mayor valor del vector reordenado, y w_i son los pesos asociados, cumpliendo $w_i \geq 0$ y $\sum_{i=1}^{n} w_i = 1$ [6]. La GDM también se ha extendido a escenarios de gran escala, donde aparecen retos como la escalabilidad, la heterogeneidad, la robustez y la influencia social [3,7]. Modelos recientes han incorporado señales de confianza y conflicto para mejorar los procesos de consenso [4]. Por ello, resulta necesario analizar no solo el acuerdo final, sino también cómo se forma dicho acuerdo y qué dinámicas sostienen la decisión colectiva [15].

2.2. Sistemas Agénticos Basados en LLM

Los sistemas agénticos basados en LLM tratan los modelos fundacionales como entidades capaces de procesar contexto, generar decisiones intermedias e interactuar con otros agentes. Estos sistemas se apoyan en arquitecturas *transformer* [16] y en taxonomías recientes que describen componentes como roles, memoria, herramientas y mecanismos de coordinación [17]. En este marco, varios agentes pueden analizar una misma tarea siguiendo roles diferentes. AutoGen muestra cómo los protocolos de interacción estructurados permiten flujos de razonamiento colaborativo [19], mientras que CAMEL ilustra cómo la especialización por roles puede inducir comportamientos diferenciados en sociedades basadas en LLM [10].

Sin embargo, la colaboración agéntica no garantiza por sí misma una decisión colectiva fiable. Cuando los agentes comparten el mismo modelo base, *prompts* similares o sesgos cercanos, sus salidas pueden estar fuertemente correlacionadas y producir consenso alrededor de una respuesta incorrecta. Por ello, distintos trabajos han explorado mecanismos para reforzar el razonamiento, como la combinación de razonamiento y recuperación de evidencia [20] o estrategias de debate y desacuerdo estructurado para mejorar robustez y factualidad [5].

Desde esta perspectiva, los sistemas agénticos basados en LLM pueden entenderse como realizaciones computacionales de procesos de decisión en grupo. Cada agente produce un juicio individual, pero la calidad de la salida final depende de cómo esos juicios se combinan y de si el consenso observado refleja robustez real o errores correlacionados. La innovación de este trabajo se sitúa precisamente ahí: en lugar de aplicar únicamente una regla fija de agregación, proponemos representar explícitamente el comportamiento colectivo mediante señales de consenso, desacuerdo, conflicto, dominancia y confianza, y utilizar dicha representación para aprender la fiabilidad de la decisión final.

3. Arquitectura

La arquitectura propuesta sigue un marco de verificación de hechos en dos etapas, ilustrado en la Figura 1. Para mayor claridad, el diagrama muestra una representación simplificada del flujo de trabajo, destacando los componentes principales que intervienen en el proceso de verificación.

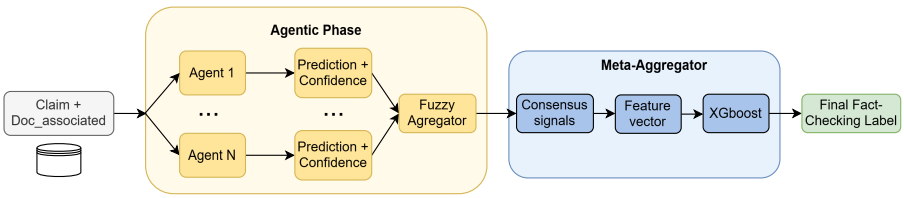


Figura 1. Visión general de la arquitectura agéntica propuesta para la verificación de hechos.

En la primera etapa, varios agentes de razonamiento analizan de forma independiente la misma afirmación y su evidencia asociada. En la segunda etapa, el comportamiento colectivo de los agentes se codifica en una representación estructurada, que posteriormente es procesada por un modelo de meta-aprendizaje para obtener la decisión final de verificación. La separación entre ambas etapas permite distinguir la generación de juicios individuales de la estimación de su fiabilidad colectiva. La fase agéntica produce varios juicios sobre el mismo par afirmación-evidencia; la fase de meta-agregación transforma esos juicios en variables que describen acuerdo, desacuerdo, dominancia, conflicto y confianza, y aprende su relación con la etiqueta final.

3.1. Fase Agéntica

La primera etapa corresponde a la *fase agéntica*, en la que todos los agentes analizan en paralelo el mismo par afirmación-evidencia (c, e).

Sea $A = \{a_1, a_2, \ldots, a_n\}$ el conjunto de agentes. Cada agente a_i produce una salida

$$y_i = (l_i, s_i, r_i),$$

donde $l_i \in \{SUPPORT, REFUTE, NOT_ENOUGH_INFO\}$ es la etiqueta predicha, s_i es la puntuación de confianza y r_i es una justificación textual. Esta justificación no se utiliza directamente por el modelo de meta-agregación, pero permite depurar el sistema y analizar cualitativamente los patrones de razonamiento de los agentes. Una vez generadas las salidas de todos los agentes, se aplica una agregación difusa para obtener una representación suave del apoyo colectivo. Para cada etiqueta, las puntuaciones de confianza se combinan mediante el operador OWA definido en la Ecuación (1), produciendo un grado de pertenencia agregado. Los pesos se fijan antes del entrenamiento mediante un perfil geométrico decreciente, evitando su ajuste sobre el conjunto de evaluación. Para n agentes, se definen como

$$\tilde{w}_j = 0{,}65^{j-1}, \qquad w_j = \frac{\tilde{w}_j}{\sum_{k=1}^{n} \tilde{w}_k}, \qquad j = 1, \ldots, n. \tag{2}$$

Se usa una configuración que asigna mayor influencia a los valores más altos, pero mantiene la contribución del resto de agentes. De este modo, el operador representa una agregación orientada a la dominancia, situada entre la media uniforme y el máximo estricto. El parámetro se mantiene fijo en todos los experimentos para garantizar una comparación reproducible y evitar que las mejoras procedan del ajuste específico de los pesos.

3.2. Meta-Agregador

La segunda etapa corresponde al *meta-agregador*. En lugar de utilizar directamente la salida del agregador difuso como decisión final, el sistema transforma el comportamiento colectivo de los agentes en una representación estructurada. Para cada instancia afirmación-evidencia, se calculan *señales de consenso* sin utilizar la etiqueta real durante la construcción de características. Estas señales se derivan principalmente de las salidas de la fase agéntica y se complementan con algunos descriptores simples de la entrada.

La representación incluye valores de pertenencia fusionados para las tres etiquetas de verificación, *SUPPORT*, *REFUTE* y *NOT_ENOUGH_INFO*; indicadores de incertidumbre y dominancia, como la entropía y el margen entre las dos clases principales; una puntuación de conflicto entre etiquetas competidoras; y descriptores basados en votos y confianza, como proporciones de voto, desacuerdo y estadísticos resumen de confianza.

Estas señales se organizan en un vector de características

$$x = (x_1, x_2, \ldots, x_m),$$

donde cada componente captura un aspecto del comportamiento colectivo del conjunto. Así, la representación no solo codifica qué etiqueta parece dominante, sino también cuán fuerte, estable o fragmentada es esa dominancia entre agentes.

Finalmente, *XGBoost* actúa como modelo de decisión final. En lugar de aprender a partir del texto crudo, aprende a partir del vector derivado de las señales de consenso. Su función es modelar qué patrones de acuerdo, desacuerdo, confianza y conflicto se asocian con resultados de verificación correctos o incorrectos. Así, la decisión final no depende solo de una regla fija de fusión, sino de una representación aprendida del comportamiento colectivo observado entre los agentes.

4. Experimentos

Para evaluar la arquitectura propuesta en un escenario realista de verificación de afirmaciones, utilizamos el benchmark Climate-FEVER [2]. Los experimentos se realizan sobre la partición de entrenamiento procesada, compuesta por 1.535 instancias afirmación-evidencia. La fase de generación agéntica se ejecuta sobre todas las instancias sin utilizar las etiquetas reales; posteriormente, las salidas de los agentes se transforman en características de meta-nivel y el protocolo de entrenamiento y *holdout* se aplica únicamente en la etapa supervisada de meta-aprendizaje.

Se construye una única partición estratificada 80/20, con 1.228 instancias de entrenamiento y 307 de *holdout*, reutilizada en todas las ejecuciones para evitar solapamientos entre entrenamiento y evaluación. El conjunto presenta un desbalance moderado: 654 instancias *SUPPORT* (42.61 %), 474 *REFUTE* (30.88 %) y 407 *NOT_ENOUGH_INFO* (26.51 %). No se utiliza información del *holdout* para el diseño de características, la selección de modelos ni el ajuste de hiperparámetros. La evaluación se reporta mediante *Accuracy*, *Macro F1* y matrices de confusión normalizadas.

La capa agéntica se instancia utilizando DeepSeek V3.2 [11]. Los experimentos preliminares con *backbones* alternativos y configuraciones mixtas mostraron tendencias similares, pero no mejoras consistentes frente al uso de un *backbone* compartido. Por ello, la evaluación principal utiliza una configuración controlada en la que todos los agentes comparten modelo base y la diversidad se induce mediante especialización por roles y *prompting*. Las cinco ejecuciones reportadas corresponden a generaciones agénticas independientes bajo la misma configuración, evaluadas siempre sobre el mismo *holdout*. Las etiquetas reales se usan únicamente como objetivos supervisados del meta-agregador, entrenado en la partición de entrenamiento mediante validación cruzada estratificada de cinco pliegues.

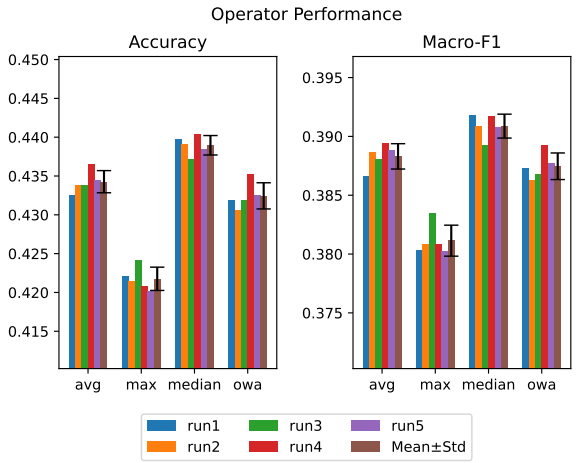


Figura 2. Efecto del operador de agregación sobre el rendimiento del conjunto.

4.1. Agregación Directa y Comportamiento de los Agentes

En primer lugar, examinamos el comportamiento de la capa agéntica antes de introducir el meta-agregador supervisado. En particular, analizamos tanto el efecto de la agregación directa como el comportamiento por clase de los cinco agentes sobre el conjunto global de *holdout*. Los operadores de agregación evaluados incluyen media, máximo, mediana y *Ordered Weighted Averaging* (OWA).

La Figura 2 muestra la *Accuracy* y el *Macro F1* a lo largo de las cinco ejecuciones. Los cuatro operadores presentan un rendimiento global similar, lo que indica que modificar la regla de fusión produce una variación limitada en el resultado final. Entre ellos, la mediana muestra el comportamiento más estable entre ejecuciones y se sitúa ligeramente por encima en las dos métricas reportadas, mientras que la media, el máximo y OWA siguen un patrón muy parecido.

La Figura 3 ofrece una visión más detallada de los agentes individuales mediante sus matrices de confusión normalizadas sobre el conjunto global de *holdout*. Se observa un patrón común en los cinco agentes. La diagonal correspondiente a *SUPPORT* suele ser más pronunciada, lo que sugiere que esta clase se maneja de forma más consistente. En cambio, la principal fuente de error se concentra en la separación entre *REFUTE* y *NOT_ENOUGH_INFO*, cuyas filas muestran una distribución más dispersa fuera de la diagonal.

4.2. Resultados de Meta-Agregación

A continuación, evaluamos si las señales colectivas producidas por la capa agéntica pueden explotarse mejor mediante meta-agregación supervisada que mediante fusión difusa directa. La Tabla 1 compara cuatro sistemas: agregación difusa directa de los cinco agentes (*Fuzzy*), el meta-agregador propuesto (*Fuzzy +*

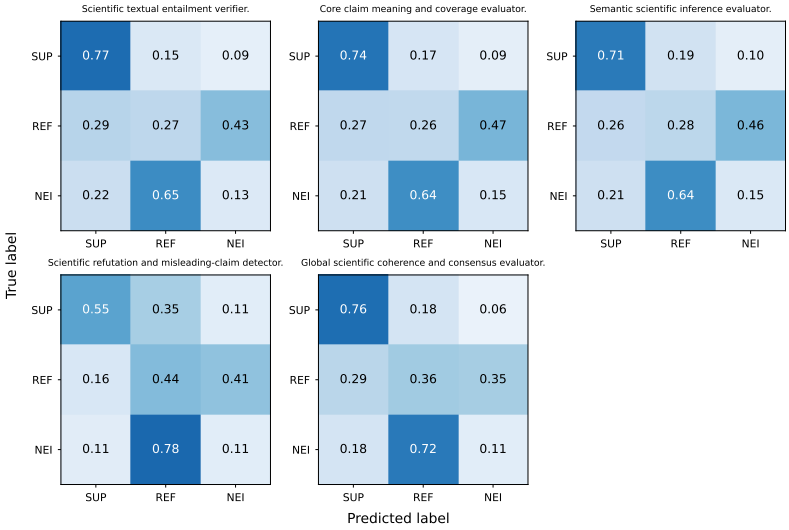


Figura 3. Matrices de confusión normalizadas de los cinco agentes especializados por rol sobre el conjunto global de *holdout*. El eje x corresponde a la etiqueta predicha, mientras que el eje y corresponde a la etiqueta real.

XGBoost), un verificador basado en un único LLM (*Single LLM*) y una línea base de *Embeddings* basada en representaciones Sentence-BERT del par afirmación-evidencia [13]. Esta última utiliza también XGBoost, lo que permite comparar la representación propuesta frente a una representación semántica estándar.

Tabla 1. Comparación de rendimiento sobre el conjunto global de *holdout*. Las métricas se presentan como media y desviación estándar sobre el número de ejecuciones indicado.

Método	Accuracy	Macro-F1
Fuzzy + XGBoost	$0{,}5557 \pm 0{,}0157$	$0{,}5328 \pm 0{,}0136$
Embeddings	$0{,}4397 \pm 0{,}0003$	$0{,}3950 \pm 0{,}0046$
Single LLM	$0{,}3941 \pm 0{,}0040$	$0{,}3587 \pm 0{,}0047$
Fuzzy	$0{,}3902 \pm 0{,}0027$	$0{,}3564 \pm 0{,}0024$

Fuzzy + XGBoost obtiene los mejores valores de *accuracy* y *macro-F1* entre los sistemas evaluados. La diferencia respecto a *Fuzzy* y *Single LLM* sugiere que la mejora no procede del número de agentes por sí mismo, sino de explotar explícitamente la estructura de sus acuerdos, desacuerdos y niveles de confianza. Además, al superar a *Embeddings* usando el mismo clasificador XGBoost, el resultado sugiere que las señales de consenso, desacuerdo, dominancia y confianza proporcionan una representación más informativa que una línea base semántica estándar. La proximidad entre *Fuzzy* y *Single LLM* confirma que la agregación

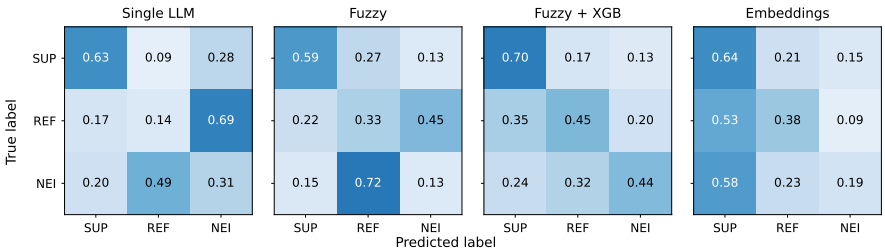


Figura 4. Matrices de confusión normalizadas de los cuatro sistemas evaluados. El eje x corresponde a la etiqueta predicha y el eje y a la etiqueta real.

directa de agentes especializados no basta para mejorar el resultado final. En este caso, la especialización por roles produce salidas adicionales, pero no suficiente diversidad epistémica para que una regla fija de fusión las explote de forma efectiva.

La Figura 4 muestra que *Fuzzy* y *Single LLM* mantienen una separación débil entre clases, especialmente entre *REFUTE* y *NOT_ENOUGH_INFO*. Por el contrario, *Fuzzy + XGBoost* presenta un comportamiento más equilibrado, lo que confirma que el meta-agregador aprovecha mejor las señales colectivas generadas por el conjunto.

Con el objetivo de identificar qué señales contribuyen en mayor medida a la decisión del meta-agregador, se realizó un análisis global de importancia de características agregado sobre las distintas ejecuciones. La Figura 5 muestra la importancia media de cada variable junto con su desviación estándar. Las características más influyentes son las proporciones de votos asignadas a las clases SUPPORT, REFUTE y NEI, especialmente la proporción de votos SUPPORT. A continuación aparecen los grados de pertenencia difusos asociados a estas mismas clases, lo que indica que el modelo combina tanto la distribución discreta de votos como el apoyo agregado obtenido mediante la fusión OWA. El desacuerdo entre agentes también presenta una contribución apreciable, aunque claramente inferior a la de las proporciones de voto y las pertenencias difusas. Por el contrario, las variables relacionadas con la longitud de la afirmación, la dispersión de la confianza y el margen entre las clases principales muestran una importancia menor. En conjunto, estos resultados sugieren que la decisión del meta-agregador depende principalmente de cómo se distribuye el apoyo entre las tres clases, mientras que las señales de conflicto, confianza e incertidumbre actúan como información complementaria.

5. Discusión

Los resultados muestran que la agregación directa presenta una capacidad limitada en el escenario evaluado. Aunque los agentes están especializados por rol, sus predicciones permanecen correlacionadas en numerosas instancias y los

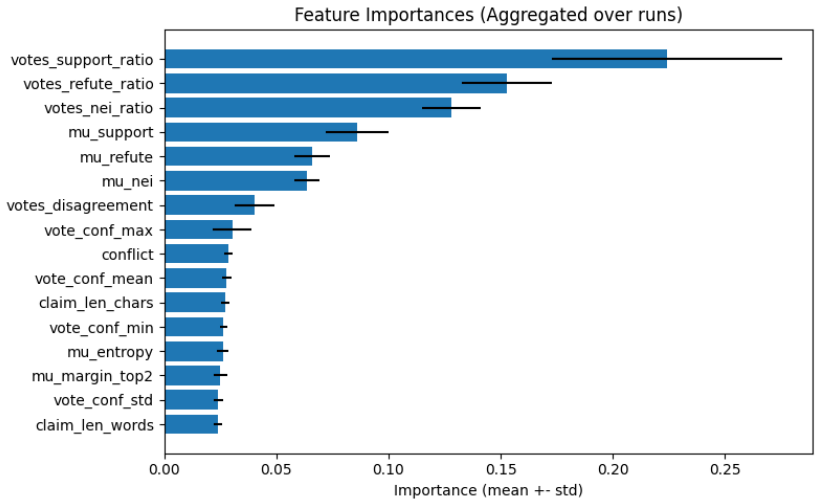


Figura 5. Importancia global de las características del meta-agregador, expresada como media y desviación estándar sobre las distintas ejecuciones.

distintos operadores producen diferencias reducidas. Por tanto, la especialización mediante *prompting* no garantiza por sí sola la diversidad necesaria para que las reglas fijas de fusión mejoren sustancialmente el rendimiento.

Este resultado matiza la interpretación del consenso en sistemas agénticos. El acuerdo entre agentes no implica necesariamente corrección, ya que puede reflejar sesgos compartidos o errores correlacionados. Desde la perspectiva de GDM, la fiabilidad de la decisión colectiva depende no solo del nivel de acuerdo, sino también de su estructura interna, incluyendo desacuerdo, conflicto, dominancia y confianza [15,8].

La aportación principal del trabajo consiste en representar explícitamente ese comportamiento colectivo. La capa de meta-agregación transforma las salidas de los agentes en señales de voto, confianza, incertidumbre, dominancia y conflicto. La comparación experimental respalda esta estrategia: la agregación difusa directa obtiene un rendimiento similar al verificador basado en un único LLM, mientras que *Fuzzy + XGBoost* alcanza los mejores resultados en *Accuracy* y *Macro-F1*. Este comportamiento es coherente con trabajos que muestran que las estrategias de agregación estructurada pueden superar a la votación simple o a las combinaciones agénticas planas [1,18].

No obstante, la propuesta requiere datos etiquetados para entrenar el meta-agregador y su coste está dominado por la ejecución de varios agentes, aunque estos pueden operar en paralelo. Además, la evaluación se limita a Climate-FEVER y a una configuración con un *backbone* compartido, DeepSeek V3.2. Estas decisiones permiten aislar el efecto de la representación colectiva, pero limitan la generalización de las conclusiones. Como trabajo futuro, será necesario

evaluar conjuntos de datos adicionales, modelos base heterogéneos y estrategias adaptativas que activen varios agentes únicamente en casos inciertos o conflictivos.

6. Conclusiones

Este trabajo ha propuesto una arquitectura de meta-agregación en dos etapas para la verificación agéntica de afirmaciones. Frente a enfoques basados únicamente en votación o fusión directa, el método representa explícitamente señales de acuerdo, desacuerdo, conflicto, dominancia y confianza, y las utiliza para estimar la decisión final.

Los resultados sobre Climate-FEVER muestran que la agregación directa de agentes especializados aporta mejoras limitadas cuando sus salidas permanecen correlacionadas. En cambio, la capa de meta-agregación obtiene un rendimiento superior al explotar la estructura interna de la decisión colectiva. La principal aportación consiste, por tanto, en trasladar el aprendizaje supervisado desde el texto original hacia una representación del comportamiento conjunto de los agentes, conectando la verificación agéntica con los principios de GDM [1].

Como trabajo futuro, resulta prioritario evaluar la propuesta en otros conjuntos de datos para determinar su capacidad de generalización y analizar en qué escenarios las señales colectivas mantienen su utilidad. Asimismo, será necesario estudiar la sensibilidad del sistema frente a diferentes configuraciones de los pesos OWA, con el fin de cuantificar hasta qué punto la orientación hacia la dominancia afecta al rendimiento final. También se plantea incorporar modelos base heterogéneos y comparar distintos meta-modelos para separar con mayor precisión el efecto de la representación colectiva del clasificador empleado.

Acknowledgments. Esta publicación es parte del proyecto de I+D+i PID2022-139297OB-I00, financiado por MICIU/AEI/10.13039/501100011033 y por FEDER/UE

Referencias

1. Ai, R., Pan, Y., Simchi-Levi, D., Tambe, M., Xu, H.: Beyond majority voting: Llm aggregation by leveraging higher-order information. arXiv preprint arXiv:2510.01499 (2025)
2. Diggelmann, T., Boyd-Graber, J., Bulian, J., Ciaramita, M., Leippold, M.: Climate-fever: A dataset for verification of real-world climate claims. arXiv preprint arXiv:2012.00614 (2020)
3. Ding, R.X., Palomares, I., Wang, X., Yang, G.R., Liu, B., Dong, Y., Herrera-Viedma, E., Herrera, F.: Large-scale decision-making: Characterization, taxonomy, challenges and future directions from an artificial intelligence and applications perspective. Information fusion **59**, 84–102 (2020)
4. Ding, R.X., Yang, B., Yang, G.R., Li, M.N., Wang, X., Chiclana, F.: A confidence and conflict-based consensus reaching process for large-scale group decision-making problems with intuitionistic fuzzy representations. IEEE Transactions on Fuzzy Systems **32**(6), 3420–3432 (2024)

5. Du, Y., Li, S., Torralba, A., Tenenbaum, J.B., Mordatch, I.: Improving factuality and reasoning in language models through multiagent debate. In: Forty-first international conference on machine learning (2024)
6. Fedrizzi, M., Kacprzyk, J., Nurmi, H.: Consensus degrees under fuzzy majorities and fuzzy preferences using owa (ordered weighted average) operators. Control and Cybernetics **22**(4), 71–80 (1993)
7. García-Zamora, D., Labella, Á., Ding, W., Rodríguez, R.M., Martínez, L.: Large-scale group decision making: A systematic review and a critical analysis. IEEE/CAA Journal of Automatica Sinica **9**(6), 949–966 (2022)
8. Herrera-Viedma, E., Cabrerizo, F.J., Kacprzyk, J., Pedrycz, W.: A review of soft consensus models in a fuzzy environment. Information Fusion **17**, 4–13 (2014)
9. Herrera-Viedma, E., Herrera, F., Chiclana, F.: A consensus model for multiperson decision making with different preference structures. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans **32**(3), 394–402 (2002)
10. Li, G., Hammoud, H., Itani, H., Khizbullin, D., Ghanem, B.: Camel: Communicative agents for"mind.^{ex}ploration of large language model society. Advances in neural information processing systems **36**, 51991–52008 (2023)
11. Liu, A., Mei, A., Lin, B., Xue, B., Wang, B., Xu, B., Wu, B., Zhang, B., Lin, C., Dong, C., et al.: Deepseek-v3. 2: Pushing the frontier of open large language models. arXiv preprint arXiv:2512.02556 (2025)
12. Park, C., Han, S., Guo, X., Ozdaglar, A.E., Zhang, K., Kim, J.K.: Maporl: Multi-agent post-co-training for collaborative large language models with reinforcement learning. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 30215–30248 (2025)
13. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP). pp. 3982–3992 (2019)
14. Smit, A., Duckworth, P., Grinsztajn, N., Barrett, T.D., Pretorius, A.: Should we be going mad? a look at multi-agent debate strategies for llms (2024). URL https://arxiv. org/abs/2311.17371
15. Trillo, J.R., Herrera-Viedma, E., Morente-Molinera, J.A., Cabrerizo, F.J.: A group decision-making method based on the experts' behavior during the debate. IEEE Transactions on Systems, Man, and Cybernetics: Systems **53**(9), 5796–5808 (2023)
16. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
17. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., et al.: A survey on large language model based autonomous agents. Frontiers of Computer Science **18**(6), 186345 (2024)
18. Wang, Z., Moriyama, S., Wang, W.Y., Gangopadhyay, B., Takamatsu, S.: Talk structurally, act hierarchically: A collaborative framework for llm multi-agent systems. arXiv preprint arXiv:2502.11098 (2025)
19. Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., et al.: Autogen: Enabling next-gen llm applications via multi-agent conversations. In: First conference on language modeling (2024)
20. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K.R., Cao, Y.: React: Synergizing reasoning and acting in language models. In: The eleventh international conference on learning representations (2022)

Aprendizaje incremental de Sistemas Difusos Lingüísticos para Flujos de Datos

Javier Martín-Moreno[1], Francisco A. Márquez[1], Ana M. Roldán[3] y Antonio Peregrín[1,2]

[1] Centro de Estudios Avanzados en Física, Matemática y Computación (CEAFMC)
[2] Instituto Andaluz Interuniversitario en Ciencia de Datos e Inteligencia Computacional (DaSCI)
[3] Departamento de Tecnologías de la Información
Universidad de Huelva, Huelva, España
javier.martin@dti.uhu.es

Resumen. Este trabajo presenta una estrategia de aprendizaje en línea para Sistemas Difusos Lingüísticos Mamdani orientada a la predicción sobre flujos de datos continuos en contextos donde la interpretabilidad es prioritaria. El sistema propuesto adopta un enfoque de modelado difuso descriptivo, fundamentado en una única partición difusa global que garantiza la coherencia semántica de toda la Base de Reglas, junto con un mecanismo incremental de generación de reglas basado en cubrimiento. Para favorecer la cooperación entre reglas y limitar el crecimiento estructural, se incorpora un esquema de ponderación que regula su influencia y actúa como poda implícita. El ajuste en línea de etiquetas y pesos se realiza mediante un algoritmo genético sobre un búfer de aprendizaje reducido. Se evalúan tres estrategias de actualización del búfer: *First-In First-Out* (FIFO), *Reservoir Sampling* (RS), y *Sliding Window with Weight Decay* (SWWD). Esta propuesta se valida sobre dos conjuntos de datos de referencia, mostrando que el modelo de aprendizaje incremental en línea funciona, y que las políticas de búfer tienen efecto. A pesar de priorizar la interpretabilidad mediante una arquitectura Mamdani descriptiva, el método propuesto alcanza una precisión competitiva, constituyendo una alternativa relevante frente a modelos evolutivos más opacos.

Palabras Clave: Sistemas Difusos Evolutivos, Aprendizaje Evolutivo en Línea, Sistema Difuso Lingüístico, Aprendizaje Incremental.

1 Introducción

El análisis de flujos de datos en tiempo real es clave en aplicaciones modernas que requieren monitorización continua, como las infraestructuras del Internet de las Cosas (IoT) y las plataformas de gestión energética. La característica principal de este tipo de datos es su naturaleza no estacionaria y temporalmente cambiante, manifestada como *concept drift*, es decir, variaciones en las propiedades estadísticas de la función objetivo a lo largo del tiempo. Los modelos predictivos implementados en estos contextos deben ser capaces de adaptarse de manera continua en tiempo real; de lo contrario, su rendimiento puede degradarse. Además, el elevado volumen y la velocidad de la generación de datos exigen estrategias de aprendizaje en línea eficientes de baja latencia.

Si bien los avances recientes en aprendizaje automático, especialmente los basados en arquitecturas profundas, han logrado una capacidad predictiva sobresaliente, lo hacen habitualmente a expensas de la explicabilidad. En numerosas aplicaciones reales, particularmente aquellas en las que la seguridad, la confianza y la transparencia son requisitos fundamentales, resulta imprescindible disponer de modelos que preserven su interpretabilidad y sean comprensibles para los expertos a lo largo de su ciclo de vida.

En este contexto, los Sistemas Difusos que Evolucionan (*Evolving Fuzzy System* - EFS) [1] ofrecen un equilibrio prometedor: combinan capacidades de aprendizaje incremental con representaciones lingüísticas inherentemente interpretables. Los modelos EFS clásicos aprenden y ajustan dinámicamente reglas difusas a medida que llegan nuevos datos, usando mecanismos de generación, fusión y eliminación de reglas con el objetivo de mantener la compacidad y la relevancia de la Base de Conocimiento (BC). Sin embargo, a pesar de ser más interpretables que los modelos de caja negra, muchos EFS se basan en arquitecturas de tipo Takagi-Sugeno (TS) o Neuro-Difusos (ND), donde los consecuentes son habitualmente funciones lineales y los conjuntos difusos difieren entre reglas. Aunque este tipo de diseño ofrece una mayor capacidad predictiva, reduce la interpretabilidad semántica, considerándose modelos de caja gris.

En el paradigma actual de la Inteligencia Artificial Explicable (*Explainable Artificial Intelligence* - XAI) [2], la interpretabilidad se concibe como una propiedad fundamental de los sistemas desplegados en entornos de alta sensibilidad. Así, las arquitecturas difusas lingüísticas de tipo Mamdani [3] resultan especialmente atractivas, dado que el empleo de etiquetas lingüísticas en los consecuentes produce estructuras de reglas significativamente más accesibles a la interpretación humana. En particular, los modelos descriptivos, en los que todas las reglas comparten una única partición difusa global en la Base de Datos (BD), garantizan la consistencia semántica en toda la Base de Reglas (BR), propiedad ausente en las alternativas aproximativas, propias de los EFS.

No obstante, adoptar una arquitectura Mamdani en un entorno dinámico plantea dos retos centrales: el crecimiento de la BR, puesto que la creación incremental de reglas puede generar fácilmente redundancias y solapamientos; y la cooperación entre reglas, ya que las nuevas reglas incorporadas pueden interferir de forma no deseada con las previamente existentes, deteriorando el rendimiento predictivo global [4].

Para abordar estos desafíos, se propone un sistema difuso Mamdani descriptivo que combina un método incremental de generación de reglas por cubrimiento, un mecanismo de ponderación de reglas que favorece su cooperación y proporciona una poda implícita, y una técnica de ajuste en línea de las etiquetas difusas y de los pesos de las reglas mediante un algoritmo evolutivo que opera sobre un búfer de memoria limitado.

El objetivo de este trabajo no es superar la precisión de los EFS de tipo TS y ND, sino ofrecer una alternativa transparente y lingüísticamente coherente con capacidad de adaptación en línea. Para fortalecer el mecanismo de adaptación, se evalúan tres estrategias de actualización del búfer, analizando el impacto de cada política sobre el rendimiento del modelo en entornos dinámicos, en respuesta a una cuestión abierta [5].

El resto del artículo se organiza de la siguiente manera: la Sección II revisa los antecedentes más relevantes. La Sección III formula el problema. La Sección IV describe el modelo propuesto. La Sección V detalla el estudio experimental. La Sección VI discute los resultados obtenidos. La Sección VII recoge las conclusiones del trabajo.

2 Antecedentes

Esta sección revisa las contribuciones más relevantes del ámbito EFS y sitúa la propuesta presentada dentro del marco científico.

Desde el trabajo seminal de Angelov [6] a principios de la década de los 2000, los EFS han sido reconocidos por su capacidad de adaptar la estructura de los modelos difusos en línea mediante aprendizaje en una sola pasada, considerándose adecuados para aplicaciones en tiempo real. Gran parte de los modelos EFS propuestos hacen uso de arquitecturas TS [7]-[12] o ND [13]-[16], compartiendo dos características comunes:

- Generación de reglas mediante agrupamiento basado en densidad, donde los conjuntos difusos son habitualmente de tipo gaussiano independientes para cada regla.
- Adaptación de los consecuentes por Mínimos Cuadrados Recursivos (*Recursive Least Squares* - RLS), primando la precisión en detrimento de la interpretabilidad.

La definición independiente de los conjuntos difusos de los antecedentes en cada regla genera particiones semánticas heterogéneas, de modo que cada regla gestiona términos lingüísticos distintos. Si bien esta flexibilidad mejora la precisión, debilita la interpretabilidad y la transparencia del modelo resultante. Asimismo, el ajuste de los parámetros de los consecuentes mediante variantes del algoritmo RLS dificulta en gran medida la comprensión del proceso de aprendizaje interno, haciendo que el proceso de inferencia del modelo sea poco transparente y dificultando la explicabilidad a posteriori.

Por el contrario, las arquitecturas Mamdani emplean consecuentes lingüísticos y, cuando se combinan con una partición difusa global, preservan la consistencia semántica en toda la BR, simplificando tanto la explicación como la inspección humana. Estos modelos están típicamente asociados a BCs descriptivas, en las que todas las reglas de la BR comparten la misma partición difusa global definida en una BD común. Esta estructura compartida garantiza que los términos lingüísticos mantengan una coherencia semántica uniforme en toda la BR, reforzando así su interpretabilidad global.

Dado que estos modelos presentan un menor número de parámetros ajustables y una estructura con menor grado de libertad, habitualmente se emplean técnicas de optimización evolutiva para su ajuste. Los algoritmos genéticos (AG) han sido ampliamente aplicados en este contexto, gracias a su capacidad de explorar espacios de búsqueda complejos y no lineales sin necesidad de información sobre el gradiente, proporcionando un buen equilibrio entre exploración del espacio de búsqueda y explotación de regiones prometedoras, ofreciendo buenos resultados con un alto coste computacional.

Sin embargo, el ajuste evolutivo se ha llevado a cabo tradicionalmente de forma *offline*, sobre conjuntos de datos estáticos mediante procesamiento iterativo. En consecuencia, muchos sistemas difusos de tipo Mamdani carecen de la capacidad de adaptarse a los cambios del entorno en tiempo real. En entornos dinámicos es imprescindible el uso de estrategias de ajuste en línea, siendo éste el principal desafío: los algoritmos evolutivos requieren múltiples evaluaciones de la función de evaluación, lo que generalmente implica la necesidad de más de una observación en cada instante de tiempo. Esto motiva el uso de un búfer de aprendizaje, cuya gestión se convierte en un aspecto clave de la adaptación en línea.

3 Formulación del Problema

El objetivo es desarrollar un método capaz de aprender y ajustar en tiempo real modelos de predicción sobre flujos de datos, donde la información llega de forma secuencial y puede sufrir *concept drift*.

En numerosos escenarios de regresión y predicción de series temporales, los modelos de aprendizaje se apoyan en técnicas de memoria explícita, de modo que una secuencia de observaciones pasadas se utiliza como entrada en lugar de depender exclusivamente de la observación actual. Siendo x_i el elemento de entrada en el instante t_i, resulta necesario mantener un pequeño búfer de retardos $\hat{x}_i$ de tamaño λ que almacena las observaciones más recientes. El vector de entrada se define como:

$$\hat{x}_i = [x_{i-\lambda}, \dots, x_i]$$

y el modelo trata de predecir el siguiente valor, típicamente:

$$y_i \approx x_{i+1}$$

Esta configuración define un sistema de memoria explícita, en el que la información pasada se proporciona directamente al modelo, permitiéndole capturar las dependencias temporales a corto plazo esenciales para una predicción precisa en entornos dinámicos. Por otro lado, los sistemas de memoria implícita almacenan la información en estados o parámetros internos y pueden realizar predicciones empleando únicamente la observación actual.

El rendimiento de los modelos predictivos empleados en tareas de regresión y predicción de series temporales se evalúa mediante dos métricas fundamentales. El Error Cuadrático Medio (*Root Mean Squared Error*, RMSE), definido en la ecuación (1), cuantifica el error absoluto de predicción en las mismas unidades que la variable objetivo y. Por su parte, el Índice de Error No Dimensional (*Non-Dimensional Error Index*, NDEI), dado en la ecuación (2), proporciona una medida de error normalizada y adimensional, que facilita la comparación justa del rendimiento de los modelos entre conjuntos de datos con diferentes escalas y varianzas.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f(\hat{x}_i) - y_i)^2} \tag{1}$$

donde N es el número de instancias evaluadas, $f(\cdot)$ representa la predicción inferida por el modelo, $\hat{x}_i$ es el valor del búfer de retardos en el instante i, e y_i es el valor objetivo (típicamente x_{i+1}).

$$NDEI = \frac{RMSE}{\sigma(\hat{x})} \tag{2}$$

donde $\sigma(\hat{x})$ es la desviación típica del conjunto de búferes de retardos evaluado.

4 Modelo Propuesto

Esta sección ofrece una descripción detallada de los principios de diseño y los aspectos de implementación que definen la estructura y el funcionamiento del modelo propuesto.

4.1 Arquitectura del Modelo

El sistema propuesto adopta un Sistema Basado en Reglas Difusas (SBRD) de tipo Mamdani descriptivo, compuesto por:

- Una BD difusa global: cada variable de entrada y de salida se particiona mediante funciones de pertenencia triangulares que cubren el dominio conocido de la variable. Estas funciones de pertenencia se corresponden con conceptos lingüísticos fijos (p. ej., Bajo, Medio, Alto) comunes a todas las reglas. Las posiciones y anchuras de estas etiquetas se actualizan en línea a medida que evoluciona la distribución de los datos, partiendo de una distribución equitativa de todo el dominio.
- Una BR lingüística: las reglas emplean etiquetas lingüísticas tanto en antecedentes como en consecuentes, siguiendo la estructura canónica Mamdani:
  $$SI\ x_1\ es\ \mu_a\ Y\ \ldots Y\ x_n\ es\ \mu_b\ ENTONCES\ y'\ es\ \mu_c$$
  Dado que todas las reglas hacen referencia a la misma BD compartida, la BC resultante mantiene su coherencia semántica.
- Un mecanismo de ponderación de reglas: cada regla incorpora un peso (*Rule Weight* - RW ∈ [0,1]) que modula su contribución durante la inferencia. Las reglas con poco peso se vuelven inactivas en la práctica, funcionando como poda implícita para controlar la redundancia y reforzar la cooperación entre las reglas con mayor peso, mejorando la precisión global del modelo.

Tradicionalmente, este tipo de BC lingüística se combina con un mecanismo de inferencia que emplea fuzzificación puntual, la T-norma mínimo para el emparejamiento e inferencia de reglas, y la estrategia FITA (*First Infer, Then Aggregate*) con defuzzificación mediante la media ponderada de máximos.

Por regla general, la generación de la BR por cubrimiento y el ajuste evolutivo de la BD se llevan a cabo mediante técnicas de procesamiento por lotes, iterando sobre todos los datos de entrenamiento disponibles. Sin embargo, el enfoque propuesto en este trabajo busca adaptar dichas técnicas a entornos dinámicos y continuamente cambiantes, donde tanto la generación de reglas como el ajuste de parámetros deben ejecutarse en línea, sin depender de datos estáticos obtenidos de forma *offline*.

4.2 Aprendizaje y Adaptación en Línea

A medida que los datos llegan de forma secuencial, el SBRD se actualiza de forma continua:

1. Generación de reglas por cubrimiento: Si el nuevo elemento observado no está suficientemente cubierto por ninguna regla existente, se genera una nueva regla utilizando la BD compartida.
2. Ajuste evolutivo en línea: El AG elegido, de tipo CHC, se emplea para ajustar la distribución de las etiquetas lingüísticas y los pesos de las reglas. Otros elementos son:
   a. Codificación: el cromosoma usa la codificación 3-tuplas (posición, anchura, etiqueta) para las etiquetas de la BD codificadas como valores reales, y los pesos de las reglas codificados como reales en el intervalo [0,1], inicializados a 1.
   b. Selección elitista para preservar los mejores cromosomas.
   c. Prevención de incesto mediante umbral de distancia de Hamming sobre una representación de punto fijo, para evitar la convergencia prematura.
   d. Operador de cruce BLX-α.
   e. Técnica de reinicio de la población para prevenir el estancamiento.
   f. Función de evaluación NDEI.

De este modo, el algoritmo optimiza la redistribución de las etiquetas lingüísticas teniendo en cuenta su influencia colectiva sobre todas las reglas, favoreciendo así la cooperación entre ellas, un aspecto clave para alcanzar una mayor precisión del modelo.

Los AGs necesitan un conjunto de muestras de tamaño considerable para evaluar los cromosomas. En entornos dinámicos, no es eficaz iterar sobre una única observación de forma significativa. En consecuencia, no resulta factible llevar a cabo el ajuste evolutivo en tiempo real bajo un enfoque estricto de pasada única. Para solucionar esta limitación, se mantiene un búfer de aprendizaje de tamaño limitado que conserva un historial reducido de datos recientes.

Este búfer permite ejecutar el AG en línea cada vez que llega una nueva observación, no evaluándola de forma aislada, sino apoyándose en un conjunto compacto de elementos almacenados previamente de manera temporal. El búfer contiene así un número reducido de muestras pasadas sobre las que los cromosomas pueden evaluarse de forma iterativa. Para poblar y mantener este búfer, se exploran tres políticas de actualización:

- *First In, First Out* (FIFO): Al recibir un nuevo elemento, se descarta el más antiguo del búfer. Conserva únicamente las observaciones más recientes. Rápida pero potencialmente limitada ante ciertas variaciones en los patrones.
- *Reservoir Sampling* (RS): Este enfoque preserva una muestra estadísticamente imparcial en el tiempo de todos los datos pasados. Cada vez que llega un nuevo elemento, cada elemento existente tiene la misma probabilidad uniforme de ser reemplazado. Ofrece una mayor robustez frente a cambios a largo plazo y cambios de patrones recurrentes.
- *Sliding Window with Weight Decay* (SWWD): Se asigna un peso decreciente a las instancias más antiguas, de modo que los elementos con mayor antigüedad tienen mayor probabilidad de ser reemplazados. Además, este peso influye en la evaluación de los cromosomas, reduciendo la contribución de las instancias más antiguas en el proceso de validación. Como resultado, las observaciones de mayor antigüedad se vuelven progresivamente más susceptibles de ser reemplazadas y menos influyentes en el proceso de ajuste.

La Figura 1 ilustra el flujo general de la metodología de ajuste en línea propuesta. Para cada nuevo búfer de retardos de entrada $\hat{x}_{i-2}$, el búfer de aprendizaje se actualiza de acuerdo con la política seleccionada. Dicho búfer sustenta a continuación el ajuste incremental de la BC del SBRD: se generan nuevas reglas Mamdani según sea necesario y, posteriormente, los términos lingüísticos y los pesos de las reglas se ajustan evaluando cada cromosoma sobre el contenido actual del búfer de aprendizaje.

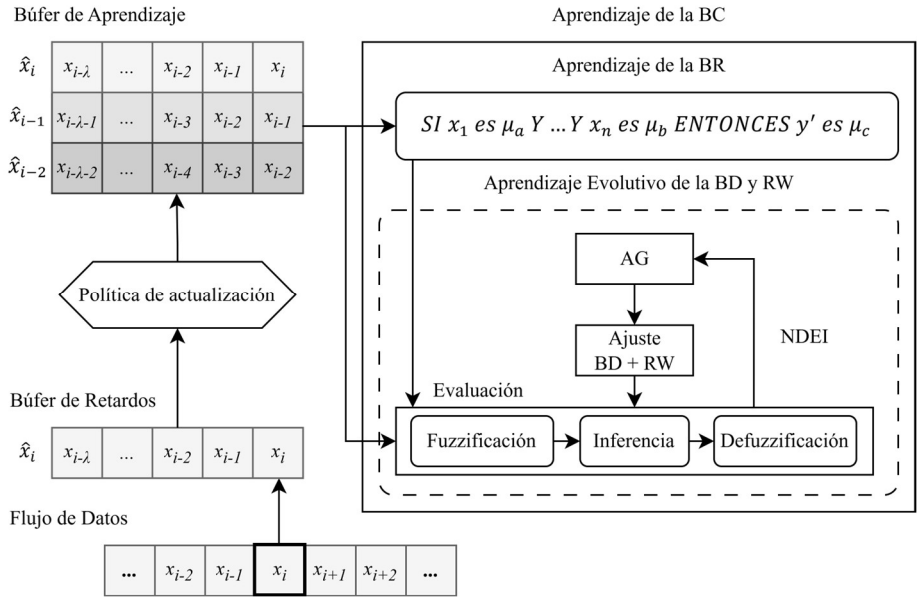


Fig. 1. Flujo del proceso de ajuste en línea del sistema difuso lingüístico.

5 Estudio Experimental

En esta sección se presenta el estudio experimental llevado a cabo para evaluar el rendimiento del modelo propuesto.

5.1 Configuración Experimental

Para evaluar la precisión del modelo se han seleccionado dos conjuntos de datos de referencia de uso extendido, considerados representativos en la literatura sobre EFS.

- Mackey-Glass: este conjunto de datos corresponde a una serie temporal no lineal generada a partir de la ecuación diferencial con retardo mostrada en (3).

$$\frac{dx(t)}{dt} = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \tag{3}$$

Se emplearon los parámetros habituales de esta serie: condición inicial $x(0) = 1.2$ y parámetro de retardo $\tau = 17$, generando un total de 6.000 muestras. El búfer de retardos se configuró con el vector de entrada $[x_{(t-18)}, x_{(t-12)}, x_{(t-6)}, x_{(t)}]$, para predecir el valor futuro $y = x_{(t+85)}$. Desde $t = 201$ a $t = 3.200$ se usaron para entrenamiento, y desde $t = 5.001$ a $t = 5.500$ para test.

- S&P500: este conjunto de datos corresponde a la serie temporal definida por los precios de cierre diarios del índice S&P500, obtenidos del repositorio de *Yahoo! Finance*, cubriendo el periodo desde el 3 de Enero de 1950 al 12 de Marzo de 2009, compuesto por 14.893 instancias, concatenado con su versión homóloga inversa en el tiempo, generando un conjunto de datos con 29.786 observaciones en total. El búfer de retardos se configuró con el vector de entrada $[x_{(t-4)}, x_{(t-3)}, x_{(t-2)}, x_{(t-1)}, x_{(t)}]$ para predecir el valor futuro $y = x_{(t+1)}$. Desde $t = 0$ a $t = 14.892$ se usaron para entrenamiento, y desde $t = 14.893$ a $t = 29.892$ para test.

Los hiperparámetros del sistema difuso evolutivo propuesto han sido los siguientes: todas las variables de entrada y salida se particionaron en tres etiquetas lingüísticas triangulares simétricas, definiendo la granularidad de la partición difusa global de la BD. Para el ajuste evolutivo, el tamaño de la población se fijó en 10 cromosomas en el modelo incremental propuesto, y 50 cromosomas en el modelo entrenado con todos los datos (modelo iterativo *offline*), que se utilizó como referencia para la comparación. El número máximo de evaluaciones por paso de aprendizaje evolutivo se limitó a 10 en la configuración en línea y 1.000 en el modelo *offline* de referencia. Se empleó el operador de cruce BLX-α con $\alpha = 0.5$. Para preservar la diversidad genética, se aplicó la prevención de incesto permitiendo el cruce únicamente entre cromosomas cuya distancia de Hamming superara ($Longitud \times 64 / 4$). Adicionalmente, se aplicó una estrategia de elitismo que conserva el 10% de los mejores cromosomas al reiniciar la población.

Con el fin de caracterizar el comportamiento del modelo propuesto, se diseña un estudio comparativo estructurado en dos etapas. A modo de ablación interna, se analiza en primer lugar el impacto de las políticas de actualización del búfer (FIFO, RS y SWWD) sobre el rendimiento predictivo. Se contrastan los resultados para distintos tamaños de búfer (100, 150, 300 y 600 instancias) con dos referencias de los extremos del espectro de aprendizaje incremental: el modelo sin búfer, entrenado en pasada única sobre cada observación sin posibilidad de reevaluación, y el modelo iterativo *offline*, entrenado con la totalidad de los datos de entrenamiento disponibles mediante procesamiento por lotes, que actúa como cota superior de referencia. En la segunda etapa se compara el mejor resultado del modelo propuesto frente a enfoques EFS del estado del arte reportados en la literatura.

5.2 Resultados

Para evaluar la influencia de las distintas políticas de búfer de aprendizaje, las Tablas 1 y 2 recogen los resultados obtenidos por el modelo difuso Mamdani descriptivo propuesto cuando es ajustado en línea mediante las diferentes políticas de búfer con diversos tamaños de búfer sobre la serie temporal Mackey-Glass y S&P500 respectivamente.

La Tabla 3 contextualiza los resultados frente a enfoques EFS del estado del arte.

Tabla 1. Comparativa de Políticas de Búfer, evaluadas mediante NDEI sobre Mackey-Glass

Tamaño de Búfer	Política de Búfer			Sin Política
	FIFO	RS	SWWD	
1 (pasada única)	-	-	-	1.324
100	1.040	1.469	1.212	-
150	1.224	1.215	1.125	-
300	1.614	0.824	1.205	-
600	1.451	0.698	0.836	-
3.200 (*offline*)	-	-	-	0.875

Tabla 2. Comparativa de Políticas de Búfer, evaluadas mediante NDEI sobre S&P500

Tamaño de Búfer	Política de Búfer			Sin Política
	FIFO	RS	SWWD	
1 (pasada única)	-	-	-	1.071
100	0.465	0.934	0.189	-
150	1.349	0.161	0.184	-
300	1.221	0.070	0.341	-
600	0.291	0.059	0.256	-
14.893 (*offline*)	-	-	-	0.067

Se observa que las políticas de búfer que retienen información a más largo plazo tienden a obtener mejores resultados, siempre que el tamaño del búfer sea suficientemente grande para que el AG pueda ajustar eficazmente los parámetros de la BC. En ocasiones, esta ventaja resulta lo suficientemente significativa como para que la configuración en línea supere la precisión obtenida mediante aprendizaje iterativo offline.

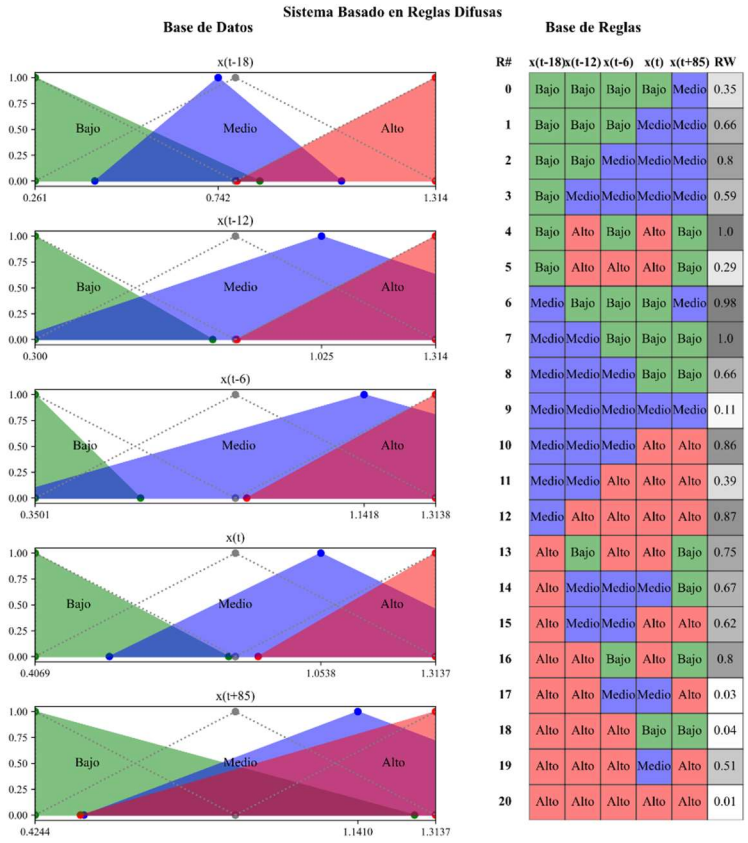
Tabla 3. Comparativa de precisión frente a EFS del Estado del Arte

Model	Mackey-Glass (NDEI)	S&P 500 (NDEI)
eTS [7]	0.347	0.015
Simpl_eTS [8]	0.279	0.045
LEOA [9]	0.248	0.123
ALMMo [10]	0.402	0.015
SEFS [11]	0.375	0.018
EFS-SLAT [12]	0.139	0.016
EFuNN [13]	0.139	0.154
DENFIS [14]	0.278	0.020
PANFIS [15]	0.301	0.090
GENEFIS [16]	0.280	0.070
RS (600)	0.698	0.059

A pesar de su diseño orientado a la interpretabilidad, el modelo propuesto alcanza resultados aceptables, especialmente teniendo en cuenta las limitaciones inherentes, como el empleo de una BD descriptiva en lugar de aproximativa y una BR de tipo Mamdani en lugar de TS o ND.

Cabe destacar que el objetivo de este trabajo no es maximizar la precisión predictiva, sino ofrecer una alternativa más interpretable que preserve la consistencia lingüística manteniendo al mismo tiempo un nivel de exactitud razonable.

La Figura 2 ilustra la BC obtenida utilizando la política RS con un tamaño de 600 sobre Mackey-Glass. Las etiquetas difusas iniciales se representan como funciones triangulares punteadas en gris, distribuidas simétricamente sobre el dominio completo de cada variable. Tras el proceso de ajuste, las etiquetas difusas adaptadas, mostradas en verde, azul y rojo, presentan posiciones y anchuras actualizadas, reflejando los cambios introducidos por el proceso evolutivo. Finalmente, los pesos de las reglas, inicializados todos a 1, han sido adaptados durante el aprendizaje, obteniendo nuevos valores que modulan la contribución de cada regla en la inferencia final.



R#	x(t-18)	x(t-12)	x(t-6)	x(t)	x(t+85)	RW
0	Bajo	Bajo	Bajo	Bajo	Medio	0.35
1	Bajo	Bajo	Bajo	Medio	Medio	0.66
2	Bajo	Bajo	Medio	Medio	Medio	0.8
3	Bajo	Medio	Medio	Medio	Medio	0.59
4	Bajo	Alto	Bajo	Alto	Bajo	1.0
5	Bajo	Alto	Alto	Alto	Bajo	0.29
6	Medio	Bajo	Bajo	Bajo	Medio	0.98
7	Medio	Medio	Bajo	Bajo	Bajo	1.0
8	Medio	Medio	Medio	Bajo	Bajo	0.66
9	Medio	Medio	Medio	Medio	Medio	0.11
10	Medio	Medio	Medio	Alto	Alto	0.86
11	Medio	Medio	Alto	Alto	Alto	0.39
12	Medio	Alto	Alto	Alto	Alto	0.87
13	Alto	Bajo	Alto	Alto	Bajo	0.75
14	Alto	Medio	Medio	Medio	Bajo	0.67
15	Alto	Medio	Medio	Alto	Alto	0.62
16	Alto	Alto	Bajo	Alto	Bajo	0.8
17	Alto	Alto	Medio	Medio	Alto	0.03
18	Alto	Alto	Alto	Bajo	Bajo	0.04
19	Alto	Alto	Alto	Medio	Alto	0.51
20	Alto	Alto	Alto	Alto	Alto	0.01

Fig. 2. BC resultante del modelo propuesto tras el ajuste de la BD y los pesos de las reglas.

Cabe destacar que el resultado permanece siempre auditable. A diferencia de los consecuentes de las arquitecturas TS o ND, la BC conserva en todo momento una representación lingüística legible, permitiendo inspeccionar cómo evolucionan los conceptos y las reglas a lo largo del tiempo. Esta trazabilidad permite la auditoría de los modelos.

6 Discusión

En términos generales, los resultados indican que las dos políticas de actualización del búfer de aprendizaje más avanzadas comienzan a mostrar su eficacia a partir de un determinado tamaño. Este comportamiento era previsible, dado que el búfer tiene sentido precisamente como mecanismo para preservar una memoria de experiencia (ejemplos históricos) que orienta el aprendizaje, incorporando lo más relevante de la dinámica pasada del sistema y consolidando progresivamente su BC.

Sin embargo, uno de los hallazgos más destacados es que el modelo propuesto, cuando se ajusta mediante una política de actualización sofisticada sobre un búfer de tamaño fijo, alcanza una precisión predictiva en ocasiones superior a la obtenida entrenando el mismo modelo sobre el conjunto de datos completo. Esto evidencia que mantener un búfer acotado pero representativo puede resultar más beneficioso que utilizar la totalidad del historial de datos, mejorando además la eficiencia del AG. Es decir, la representatividad del búfer importa más que la exhaustividad del registro histórico.

A la vista de estos resultados, conviene señalar algunas limitaciones del enfoque propuesto: la naturaleza descriptiva del modelo, junto con las restricciones inherentes a las reglas de tipo Mamdani, conduce inevitablemente a una precisión predictiva inferior a la de las arquitecturas TS o ND. Además, la optimización evolutiva es computacionalmente más costosa que los métodos basados en gradiente, un aspecto relevante que debe valorarse cuidadosamente en entornos de flujo de datos en tiempo real.

7 Conclusiones

En este trabajo se ha presentado un enfoque de aprendizaje en línea para sistemas difusos lingüísticos Mamdani de tipo descriptivo, que combina el aprendizaje incremental de reglas con el ajuste evolutivo de las particiones difusas globales y los pesos de las reglas, sobre un búfer de aprendizaje de tamaño limitado. Se han comparado tres políticas distintas de actualización del búfer (FIFO, RS y SWWD) con el fin de analizar su impacto en el aprendizaje evolutivo en línea.

El estudio experimental realizado sobre los conjuntos de datos de referencia pone de manifiesto que la composición y la dinámica de actualización del búfer de aprendizaje desempeñan un papel determinante en la capacidad de adaptación en línea del modelo. En particular, las políticas que preservan una representación más rica de la información pasada obtienen un mejor rendimiento, logrando una precisión competitiva con respecto a los modelos difusos evolutivos orientados a la exactitud, sin renunciar a la alta interpretabilidad inherente a los modelos Mamdani descriptivos. Este equilibrio adquiere una importancia creciente en despliegues sobre flujos de datos, donde los modelos deben adaptarse con el tiempo y mantenerse auditables y comprensibles.

Como líneas de trabajo futuro, se contempla explorar un abanico más amplio de estrategias de gestión del búfer, evaluar la aplicabilidad del modelo propuesto en otros escenarios reales de flujo de datos y añadir métricas de error que evalúen mejor el ajuste del modelo.

Agradecimientos. Este trabajo ha sido financiado por el Ministerio de Ciencia, Innovación y Universidades, bajo el proyecto PID2023-150070NB-I00.

Declaración de Intereses. Los autores no tienen conflicto de interés que declarar.

Referencias

1. Angelov, P.: Evolving fuzzy systems. In: Granular, Fuzzy, and Soft Computing, pp. 725–741. Springer US, New York (2023)
2. Herrera, F.: Reflections and attentiveness on explainable artificial intelligence (XAI): The journey ahead from criticisms to human-AI collaboration. Inf. Fusion 121, 103133 (2025)
3. Fernandez, A., Herrera, F., Cordon, O., del Jesus, M.J., Marcelloni, F.: Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to? IEEE Comput. Intell. Mag. 14(1), 69–81 (2019)
4. Márquez, F.A., Peregrín, A., Herrera, F.: Cooperative evolutionary learning of fuzzy rules and parametric aggregation connectors for Mamdani linguistic fuzzy systems. IEEE Trans. Fuzzy Syst. 15(6), 1162–1178 (2007)
5. Martín-Moreno, J., Márquez, F.A., Roldán, A.M., Peregrín, A.: Online on-device adaptation of linguistic fuzzy models for TinyML systems. AI 6(12), 325 (2025)
6. Angelov, P.P., Filev, D.P.: An approach to online identification of Takagi-Sugeno fuzzy models. IEEE Trans. Syst., Man, Cybern. B, Cybern. 34(1), 484–498 (2004)
7. Angelov, P., Buswell, R.: Evolving rule-based models: a tool for intelligent adaption. In: Proc. IFSA World Congr. 20th NAFIPS Int. Conf., vol. 3, pp. 1062–1067 (2001)
8. Angelov, P.: Fuzzily connected multimodel systems evolving autonomously from data streams. IEEE Trans. Syst., Man, Cybern. B, Cybern. 41(4), 898–910 (2011)
9. Ge, D., Zeng, X.J.: Learning evolving T–S fuzzy systems with both local and global accuracy – a local online optimization approach. Applied Soft Computing 68, 795–810 (2018).
10. Angelov, P.P., Gu, X., Principe, J.C.: Autonomous learning multimodel systems from data streams. IEEE Trans. Fuzzy Syst. 26(4), 2213–2224 (2018)
11. Ge, D., Zeng, X.J.: A self-evolving fuzzy system which learns dynamic threshold parameter by itself. IEEE Trans. Fuzzy Syst. 27(8), 1625–1637 (2019)
12. Ge, D., Zeng, X.J.: Learning data streams online - an evolving fuzzy system approach with self-learning/adaptive thresholds. Inf. Sci. 507, 172–184 (2020)
13. Kasabov, N.: Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 31(6), 902–918 (2001).
14. Kasabov, N.K., Song, Q.: DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. IEEE Trans. Fuzzy Syst. 10(2), 144–154 (2002)
15. Pratama, M., Anavatti, S.G., Angelov, P.P., Lughofer, E.: PANFIS: A novel incremental learning machine. IEEE Transactions on Neural Networks and Learning Systems 25(1), 55–68 (2013).
16. Pratama, M., Anavatti, S.G., Lughofer, E.: GENEFIS: Toward an effective localist network. IEEE Transactions on Fuzzy Systems 22(3), 547–562 (2013).

Clasificación explicable de EEG mediante descriptores compactos y reglas difusas lingüísticas

Javier Aguirre-Eraso[1][0009−0001−0382−0800], Carlos Lopez-Molina[1,2][0000−0002−0904−9834] y Carmen Vidaurre[3,4,5][0000−0003−3740−049X]

[1] Universidad Pública de Navarra (UPNA), Pamplona, España
javier.aguirreeraso@unavarra.es
[2] IdiSNA, Navarra Institute for Health Research, NavarraBiomed, Pamplona, España
[3] Basque Center on Cognition, Brain and Language (BCBL), Donostia-San Sebastián, España
[4] Ikerbasque, Bilbao, España
[5] TU-Berlin / BIFOLD Center, Berlín, Alemania

Resumen El aprendizaje profundo se ha posicionado como la solución predeterminada en la clasificación de señales biomédicas. Sin embargo, su opacidad limita su aplicabilidad en escenarios donde las predicciones requieren auditoría o justificación. Este estudio evalúa el equilibrio entre rendimiento y explicabilidad en una tarea de clasificación de edad mediante EEG de reposo (conjunto LEMON, N=203). Comparamos el rendimiento de una red neuronal convolucional (CNN) sobre matrices espectrales complejas frente a un framework difuso basado en descriptores compactos de conectividad biológicamente interpretables. La validación cruzada muestra una diferencia predictiva marginal: la CNN obtiene un accuracy de 0.729 (AUC: 0.800) frente al 0.709 (AUC: 0.776) del modelo difuso. A cambio de esta mínima diferencia, el modelo transparente justifica sus decisiones explícitamente utilizando una base de solo cinco reglas lingüísticas legibles. Estos resultados sugieren que, en ciertos problemas, compensa no perseguir el máximo rendimiento predictivo absoluto si a cambio se prioriza la interpretabilidad y la trazabilidad del modelo.

Palabras clave: Inteligencia artificial explicable · EEG · reglas difusas · conectividad funcional · aprendizaje profundo

1. Introducción

En los últimos años, el deep learning se ha consolidado como la primera opción para abordar muchos problemas de clasificación. La razón es comprensible: cuando hay suficientes datos, las redes neuronales pueden aprender representaciones complejas sin exigir un diseño manual de variables. En EEG, por ejemplo, las arquitecturas convolucionales se han convertido en una de las familias más empleadas por su capacidad para explotar estructura espacial y espectral, y por su flexibilidad frente a distintos paradigmas experimentales [10,7]. Esta tendencia,

sin embargo, ha creado una inercia metodológica peligrosa: asumir que un modelo más profundo o complejo es necesariamente la mejor solución.

La interpretabilidad no es un requisito estético. En salud, una predicción puede influir en decisiones clínicas o en la generación de hipótesis sobre mecanismos fisiológicos. En finanzas, un modelo puede afectar a la concesión de crédito, la detección de fraude o la gestión de riesgo, escenarios donde la trazabilidad es inseparable de la responsabilidad. En neurotecnología, además, el modelo no solo clasifica: también puede orientar qué patrones cerebrales merecen ser estudiados. En estos dominios, explicar los resultados de una caja negra puede quedarse corto si el objetivo no es solo acertar, sino justificar y discutir la decisión. Rudin ha defendido que, cuando hay consecuencias relevantes o necesidad de auditoría, conviene considerar modelos intrínsecamente interpretables antes de recurrir a cajas negras explicadas *a posteriori* [11]. En la misma línea, la literatura sobre inteligencia artificial explicable (XAI) subraya que transparencia, responsabilidad y confianza dependen de que las explicaciones sean comprensibles para los usuarios del sistema, no solo para quien lo programó [1]. El punto central, por tanto, no es negar la utilidad del deep learning, sino preguntarse cuándo compensa usarlo. Si un modelo interpretable obtiene resultados muy próximos a los de una red neuronal, el pequeño margen predictivo de la red puede no justificar la pérdida de legibilidad. Esta cuestión es especialmente relevante en muestras moderadas, donde las arquitecturas flexibles pueden aprender patrones útiles, pero también explotar regularidades difíciles de auditar. Frente a ello, un modelo basado en descriptores comprensibles y reglas lingüísticas ofrece una alternativa mucho más clara en su razonamiento.

El EEG de reposo ofrece un escenario adecuado para estudiar este equilibrio. Las señales registradas reflejan interacciones distribuidas entre regiones cerebrales y bandas de frecuencia, y los cambios asociados al envejecimiento sano pueden manifestarse como variaciones sutiles de potencia, sincronía o estructura de conectividad [6]. Además, las matrices espectrales complejas contienen información que permite construir descriptores biológicos orientados a interacciones no instantáneas entre diferentes regiones cerebrales. En particular, la parte imaginaria de la coherencia se propuso para reducir la influencia del efecto de conducción de volumen [9].

En este trabajo se comparan de forma controlada dos aproximaciones sobre la misma tarea binaria, distinguir participantes jóvenes y mayores en el conjunto LEMON [2]. La primera es una CNN que opera directamente sobre las matrices espectrales complejas. La segunda es ExFuzzy, un framework que combina selección de descriptores compactos, inspirados en propiedades biológicas de la conectividad, con reglas difusas lingüísticas optimizadas evolutivamente [3]. La contribución principal no es reclamar un nuevo estado del arte, sino mostrar que una solución interpretable puede acercarse mucho al rendimiento de una CNN y, al mismo tiempo, devolver reglas comprensibles sobre las variables que sostienen la clasificación.

2. Materiales y métodos

2.1. Datos y representación de entrada

Se empleó la condición de ojos abiertos (EO) del conjunto Leipzig Mind-Brain-Body, una base multimodal diseñada para estudiar relaciones entre cerebro, cognición, emoción y fisiología periférica en adultos jóvenes y mayores [2]. El conjunto de datos contiene 203 participantes: 138 etiquetados como *Young* y 65 como *Old*. Para cada participante se dispone de matrices espectrales complejas en cinco bandas, Delta, Theta, Alpha, Beta y Gamma, con 61 canales por banda.

La representación de partida es la misma para ambos enfoques. Para la CNN se construyó un tensor concatenando las partes real e imaginaria de las cinco matrices espectrales, con tamaño final $10 \times 61 \times 61$. Para el modelo difuso, cada matriz compleja se resumió mediante descriptores compactos de conectividad. En cada banda se calculó fuerza media, entropía de fase ponderada por magnitud y heterogeneidad de centralidad; además, se añadió una medida de interferencia entre Alpha y Delta. Esta compresión reduce la dimensionalidad y facilita la interpretación, pero también elimina información espacial fina que la CNN sí conserva.

2.2. Clasificador convolucional

La CNN se diseñó como un baseline deliberadamente pequeño, ajustado al tamaño de muestra disponible. La arquitectura recibe 10 canales de entrada y contiene dos bloques convolucionales: una convolución 5×5 con 16 filtros, activación ReLU y *max pooling*, seguida de una convolución 3×3 con 32 filtros, activación ReLU, otro *max pooling* y *adaptive average pooling*. La cabeza predictora aplana la representación y aplica una capa densa de 16 unidades con ReLU antes de la salida binaria. El entrenamiento utilizó *BCEWithLogitsLoss* ponderada por la proporción de clases del pliegue de entrenamiento, Adam con tasa de aprendizaje 10^{-3}, *weight decay* 10^{-4}, lotes de 16 muestras, un máximo de 60 épocas y *early stopping* con paciencia de 8 épocas. En cada pliegue externo, el 20 % del entrenamiento se reservó como validación interna estratificada y la normalización por canal se estimó solo con las muestras de entrenamiento.

El umbral de decisión de la CNN se mantuvo fijo en 0.5. No se optimizó por pliegue porque la CNN se usa como referencia compacta, no como techo de rendimiento exhaustivamente ajustado. Esta distinción es importante porque el objetivo experimental es comparar rendimiento y transparencia bajo un presupuesto razonable, no agotar la búsqueda de arquitecturas neuronales.

2.3. Clasificador difuso explicable

El modelo transparente se implementó con ExFuzzy, una biblioteca orientada a inteligencia artificial explicable simbólica mediante programación difusa [3]. En cada fold, la selección de variables se ajustó únicamente con las muestras de entrenamiento mediante *SelectKBest* y estadístico ANOVA F, manteniendo

$k = 5$ descriptores. Las variables seleccionadas se escalaron al intervalo $[0, 1]$ y se aplicó Edited Nearest Neighbours sobre el entrenamiento como limpieza local de muestras. El clasificador final usó particiones difusas de tipo 1 con tres términos lingüísticos, *Low*, *Medium* y *High*, y una base de reglas con un máximo de seis reglas y hasta dos antecedentes por regla.

Las reglas se optimizaron con el backend evolutivo de ExFuzzy basado en *pymoo*. Este backend realiza una búsqueda poblacional de bases de reglas candidatas, variando antecedentes y consecuentes lingüísticos y seleccionando las soluciones de acuerdo con la función de ajuste interna de ExFuzzy, siguiendo el enfoque general de los sistemas difusos evolutivos [5]. La configuración empleó 40 generaciones, población de 22 individuos, probabilidad de variación de 0.20, η de cruce SBX igual a 5.0, η de mutación igual a 10.0 y torneo de tamaño 2. En cada fold se entrenaron tres modelos con semillas 42, 43 y 44 y se promediaron sus probabilidades. El umbral para la clase *Old* se fijó en 0.77 a partir de la exploración previa y se mantuvo constante durante la validación externa. Las reglas finales se reajustaron con todos los datos y la configuración seleccionada, únicamente para describir el modelo lingüístico de referencia; las métricas reportadas proceden de predicciones fuera de muestra en los folds externos.

2.4. Evaluación

Ambos modelos se evaluaron mediante validación cruzada estratificada de cinco particiones. Se reportaron accuracy, área bajo la curva ROC (AUC), coeficiente de correlación de Matthews (MCC), balanced accuracy, F1 macro y recall por clase. Estas métricas permiten resumir tanto el rendimiento global como el comportamiento por clase en una muestra con tamaños de grupo distintos. La Tabla 1 resume las diferencias principales entre las dos configuraciones antes de presentar los resultados.

Tabla 1. Configuración resumida de los dos modelos comparados.

Componente	CNN compacta	ExFuzzy
Entrada	Tensor real/imaginario $10 \times 61 \times 61$	16 descriptores iniciales de conectividad
Selección de variables	No aplicada	ANOVA, $k = 5$
Preprocesamiento	Normalización por canal en entrenamiento	Escalado MinMax y ENN
Modelo	Dos bloques convolucionales y cabeza densa	Reglas difusas tipo 1 con 3 particiones
Decisión	Umbral 0.5	Umbral 0.77 para *Old*

3. Resultados

La comparación de rendimiento se muestra en la Fig. 1. La CNN compacta alcanzó el mejor resultado en todas las métricas, pero la diferencia con ExFuzzy fue pequeña en las medidas globales: 0.020 puntos en accuracy agrupado y 0.024 puntos en AUC. Esta cercanía es relevante porque el modelo difuso trabaja con una representación mucho más comprimida y, a cambio, conserva una lectura directa de sus decisiones.

La diferencia entre ambos modelos debe leerse junto con la naturaleza de sus entradas. La CNN recibe la matriz espectral completa, mientras que ExFuzzy opera sobre un número reducido de descriptores agregados. Que el margen en accuracy y AUC sea limitado sugiere que parte de la información relevante para la tarea queda capturada por variables compactas de conectividad, sin necesidad de utilizar una representación neuronal de mayor capacidad.

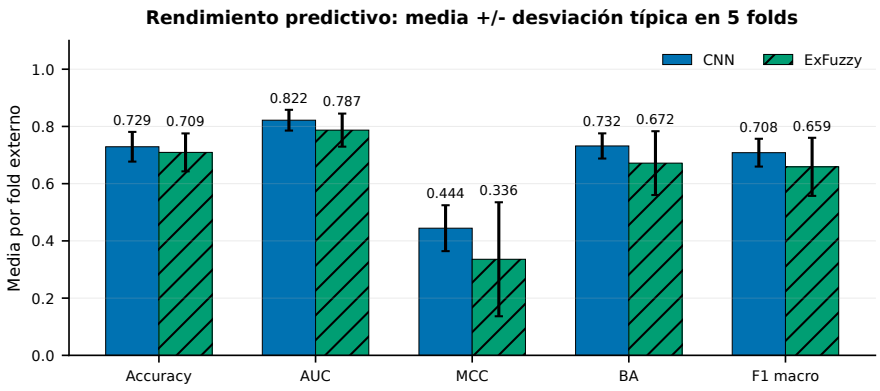


Figura 1. Comparación de rendimiento entre la CNN compacta y ExFuzzy. Las barras muestran media y desviación típica en los cinco pliegues externos. La CNN conserva ventaja predictiva, pero ExFuzzy se mantiene cerca con una estructura mucho más interpretable. BA denota balanced accuracy.

Más allá de las métricas agregadas, el interés del modelo difuso está en que permite inspeccionar el criterio de clasificación. La predicción no queda reducida a una puntuación final, sino que puede relacionarse con un conjunto pequeño de variables fisiológicamente interpretables. Esta propiedad facilita discutir si los patrones utilizados por el modelo son coherentes con el conocimiento disponible sobre conectividad y envejecimiento.

La diferencia cualitativa entre ambos enfoques aparece en la base final de reglas. La CNN devuelve una probabilidad, pero no una descripción directa de qué propiedades neurofisiológicas han pesado en la decisión. ExFuzzy, en cambio, utiliza cinco variables seleccionadas: *Theta_Strength*, *Theta_Centrality*,

Beta_ Entropy, *Gamma_ Strength* y *Gamma_ Entropy*. Las reglas asociadas a *Young* incluyen baja entropía beta, alta fuerza theta y una combinación de centralidad theta media con fuerza gamma media. Las reglas asociadas a *Old* incluyen baja fuerza theta y alta entropía gamma. La Fig. 2 resume estos antecedentes y muestra para cada regla los valores de *dominance score* (DS) y accuracy interna (ACC) proporcionados por ExFuzzy.

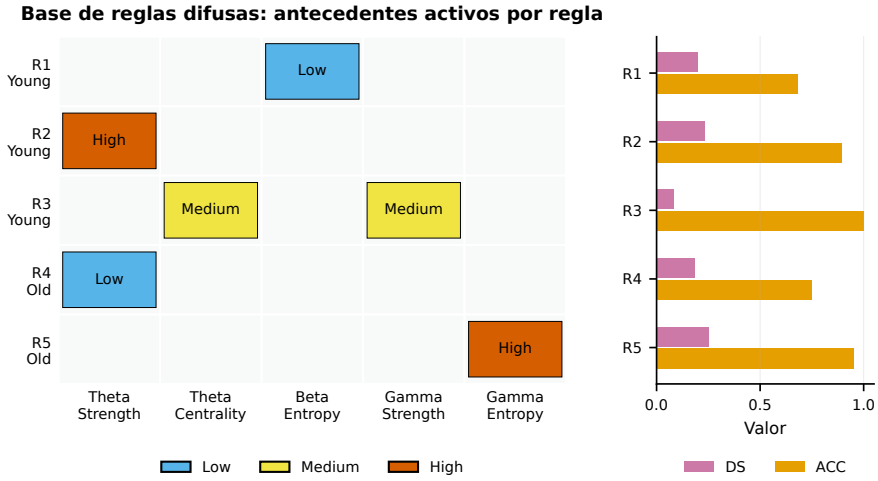


Figura 2. Visualización de la base de reglas final de ExFuzzy. La matriz izquierda indica qué término lingüístico participa en cada antecedente; el panel derecho muestra DS y ACC de cada regla. Esta representación permite inspeccionar de forma directa qué descriptores conducen a cada clase.

4. Discusión

Los resultados apuntan a una conclusión práctica: no todos los problemas de clasificación requieren necesariamente deep learning como primera respuesta. La CNN logra mejores métricas agregadas, algo esperable porque recibe una representación mucho más rica de la matriz espectral completa. El sistema difuso, en cambio, trabaja con una compresión drástica de la señal y aun así queda solo 0.020 puntos por debajo de la CNN en accuracy agrupado y 0.024 puntos por debajo en AUC. En un contexto donde solo importara exprimir la métrica, esa diferencia favorecería a la CNN. En un contexto donde la decisión debe ser examinada, discutida y relacionada con conocimiento del dominio, la cercanía del rendimiento cambia la lectura: un modelo interpretable puede ser una opción más adecuada aunque no alcance el máximo absoluto. Esta lectura es coherente con la defensa de modelos intrínsecamente interpretables en dominios de alto

impacto [11] y con la idea de que una explicación útil debe poder discutirse desde el conocimiento del usuario, no solo desde el funcionamiento interno del algoritmo [8].

La interpretación de las reglas exige cautela, pero también muestra por qué este tipo de framework es valioso. La presencia de fuerza theta, entropía beta y entropía gamma encaja con la idea general de que el envejecimiento sano modifica ritmos de reposo y patrones de conectividad [6]. Un modelo basado en reglas permite formular esa relación en términos que pueden revisarse: si baja la fuerza theta, si aumenta la entropía gamma, si una combinación de centralidad y fuerza entra en una zona media. Esa lectura no está disponible de forma inmediata en la CNN. Sin embargo, este estudio es predictivo y observacional; no permite atribuir causalidad ni establecer biomarcadores clínicos.

Desde el punto de vista metodológico, la comparación debe leerse como una demostración de trazabilidad, no como una evaluación definitiva de todas las arquitecturas posibles. Ni la CNN ni el sistema fuzzy fueron sometidos a una búsqueda extensa de hiperparámetros, por lo que las métricas reportadas describen estas configuraciones concretas y no el techo de rendimiento de cada familia metodológica. En trabajos futuros sería conveniente reforzar esta comparación con semillas múltiples, validación externa independiente, calibración del umbral dentro de cada fold y almacenamiento de predicciones por muestra, de modo que puedan analizarse las diferencias entre modelos con mayor precisión.

La literatura de sistemas difusos lingüísticos subraya que la interpretabilidad depende tanto del número de reglas como de la semántica de las variables utilizadas [12,4]. En este sentido, el modelo presentado cumple dos condiciones favorables: usa pocos descriptores y expresa las decisiones con términos lingüísticos simples. Esto no convierte automáticamente sus reglas en verdad neurobiológica, ni demuestra que constituyan la representación mínima, única o causalmente necesaria del fenómeno. La ventaja es otra: el modelo deja ver sus supuestos. Esa visibilidad permite discutirlos, contrastarlos con conocimiento experto y refutarlos si no se sostienen en nuevos datos. En campos como salud, finanzas o neurotecnología, esa posibilidad de auditoría puede ser tan importante como unas décimas de mejora en una métrica.

5. Conclusiones

Este estudio compara una CNN compacta y ExFuzzy en una tarea de clasificación de edad con EEG de reposo del conjunto LEMON. La CNN alcanza mejores métricas predictivas, pero ExFuzzy logra resultados muy próximos y expone su lógica mediante una base de reglas breve y visualizable. El mensaje principal es que deep learning no tiene por qué ser siempre la solución por defecto: cuando un framework interpretable sobre características biológicas descriptivas ofrece un rendimiento similar, la capacidad de explicar la decisión puede inclinar la balanza a su favor. La evidencia actual debe considerarse preliminar: las reglas son útiles para formular hipótesis sobre conectividad en bandas theta, beta y

gamma, pero requieren validación externa y análisis estadístico más amplio antes de interpretarse como marcadores neurofisiológicos estables.

Acknowledgments. Este trabajo ha sido financiado por la Agencia Estatal de Investigación (Gobierno de España, ayuda PID2022-136627NB-I00). Carmen Vidaurre cuenta con apoyo de las ayudas PID2020-118829RB-I00 y PID2024-161502OB-I00, y de IKERBASQUE.

Disclosure of Interests. Los autores declaran que no tienen intereses en conflicto relevantes para el contenido de este artículo.

Referencias

1. Arrieta, A.B., Diaz-Rodriguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., Herrera, F.: Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion **58**, 82–115 (2020). https://doi.org/10.1016/j.inffus.2019.12.012
2. Babayan, A., Erbey, M., Kumral, D., Reinelt, J.D., Reiter, A.M.F., Roebbig, J., Schaare, H.L., Uhlig, M., Anwander, A., Bazin, P.L., et al.: A mind-brain-body dataset of MRI, EEG, cognition, emotion, and peripheral physiology in young and old adults. Scientific Data **6**, 180308 (2019). https://doi.org/10.1038/sdata.2018.308
3. Fumanal-Idocin, J., Andreu-Perez, J.: Ex-Fuzzy: a library for symbolic explainable AI through fuzzy logic programming. Neurocomputing **599**, 128048 (2024). https://doi.org/10.1016/j.neucom.2024.128048
4. Gacto, M.J., Alcala, R., Herrera, F.: Interpretability of linguistic fuzzy rule-based systems: an overview of interpretability measures. Information Sciences **181**(20), 4340–4360 (2011). https://doi.org/10.1016/j.ins.2011.02.021
5. Herrera, F.: Genetic fuzzy systems: taxonomy, current research trends and prospects. Evolutionary Intelligence **1**, 27–46 (2008). https://doi.org/10.1007/s12065-007-0001-5
6. Kang, J.H., Bae, J.H., Jeon, Y.J.: Age-related characteristics of resting-state electroencephalographic signals and the corresponding analytic approaches: a review. Bioengineering **11**(5), 418 (2024). https://doi.org/10.3390/bioengineering11050418
7. Lawhern, V.J., Solon, A.J., Waytowich, N.R., Gordon, S.M., Hung, C.P., Lance, B.J.: EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces. Journal of Neural Engineering **15**(5), 056013 (2018). https://doi.org/10.1088/1741-2552/aace8c
8. Miller, T.: Explanation in artificial intelligence: insights from the social sciences. Artificial Intelligence **267**, 1–38 (2019). https://doi.org/10.1016/j.artint.2018.07.007
9. Nolte, G., Bai, O., Wheaton, L., Mari, Z., Vorbach, S., Hallett, M.: Identifying true brain interaction from EEG data using the imaginary part of coherency. Clinical Neurophysiology **115**(10), 2292–2307 (2004). https://doi.org/10.1016/j.clinph.2004.04.029
10. Roy, Y., Banville, H., Albuquerque, I., Gramfort, A., Falk, T.H., Faubert, J.: Deep learning-based electroencephalography analysis: a systematic review. Journal of Neural Engineering **16**(5), 051001 (2019). https://doi.org/10.1088/1741-2552/ab260c
11. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x

12. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—I. Information Sciences **8**(3), 199–249 (1975). https://doi.org/10.1016/0020-0255(75)90036-5