Web Usage Mining: users' navigational patterns extraction from web logs using Ant-based Clustering Method

Kobra Etminani [1] Mohammad-R. Akbarzadeh-T. [2] Noorali Raeeji Yanehsari [3]

1 Dept. of Computer Engineering, Ferdowsi University of Mashhad, Iran
2 Depts of Electrical Engineering and Computer Engineering, Ferdowsi University of Mashhad, Iran
3 Iran Khodro, Khorasan, Iran
Email: etminani@wali.um.ac.ir, akbarzadeh@ieee.org, raeeji@ikkco.ir

Abstract— *Web Usage Mining is the process of applying data mining techniques to the discovery of usage patterns from data extracted from Web Log files. It mines the secondary data (web logs) derived from the users' interaction with the web pages during certain period of Web sessions. Web usage mining consists of three phases, namely preprocessing, pattern discovery, and pattern analysis. In this paper, web logs of our university web server logs (http://www.um.ac.ir/) are pre-processed. Then, ant-based clustering is applied to pre-processed logs to extract frequent patterns for pattern discovery and then it is displayed in an interpretable format. Result of this paper would be useful for our university web site owner.*

Keywords— web usage mining, ant-based clustering, frequent pattern extraction, web mining.

1 Introduction

Web usage information mining could help to engage new customers, maintain current customers, track customers who are leaving web site, and so on [1]. Usage information can be extracted to increase web server efficiency by pre-fetching and caching strategies [2].Based on several researches done in the area of web mining, we can broadly classify it into three domains: web content mining, web structure mining, and web usage mining.

Web content mining is the process of extracting knowledge from web documents such as text and multimedia. Knowledge extraction from the structure of web and hyperlink references is called web structure mining. Web usage mining is the process of knowledge exploitation from the secondary data [3]. By secondary data, we mean browser logs, user profiles, web server access logs, registration data, user sessions or transactions, cookies, user queries, mouse clicks and any other data that is the result of interaction with the Web.

In the following section we give an overview over the related work. Section 3 explains the problem and ant-based clustering in more detail. Section 4 goes into detail how we implement the proposed method i.e. the experimental procedure of the proposed method and results are shown. We concluded our work in Section 5.

2 Related Work

There are several methods for pattern extraction from the secondary data (web logs) such as Masseglia et al. [2] [7], Spiliopoulou et al. [8], Bonchi et al. [9], Hay et al. [10], Zhu et al. [11], Nakagawa and Mobasher [12]. In [2], Masseglia et al. invented a new method named PSP (Prefix-tree for Sequential Patterns) that follows the basic principles of GSP (Generalized Sequential Patterns algorithm) in [13]. The tree structure in PSP is similar to the *prefix-tree* used in [14]. At the *k*th step, the tree has a depth of *k*. Each branch from the root to a leaf stands for a candidate sequence.

In [8], Spiliopoulou et al. proposed the WUM (Web utilization miner) tool that determines patterns which are noticeable from the statistical view. So, it emphasizes the frequency (minimum support) of the patterns.

Hay et al. in [10] applied the notion of time embedded in the navigations to cluster user sessions. They used an alignment algorithm to compute the distance between sessions.

Zhu et al. in [11] considered navigating between web sites as a Markov chain and mainly discussed about Markov model problems.

In [12], Nakagawa and Mobasher show that depending on whether the propositions are based on frequent itemsets or frequent sequences, the features of the site have an impact on the quality of the refinement shown to users.

In this paper, we propose a new method for extracting patterns from web logs based on ant clustering algorithm. We apply ant-based clustering for pattern discovery, other similar methods applied ant colony clustering to segregate visitors [15]. Some methods applied Markov models for modeling user web navigation behavior. But the proposed method has the similarity and speed of ant-based clustering algorithm rather than other clustering algorithms.

3 Problem description

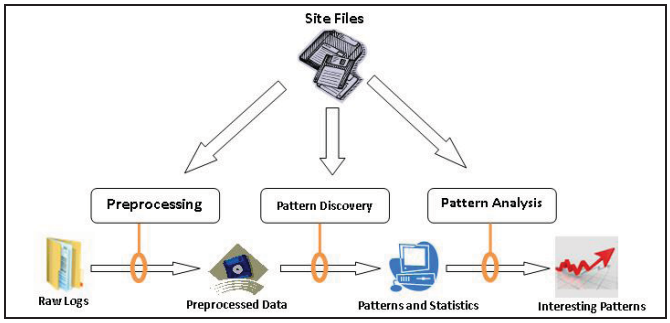

Figure 1: Usage Mining Process.

As shown in Figure 1 from [6], there are three main tasks for performing Web Usage Mining: Pre-processing, Pattern

ISBN: 978-989-95079-6-8

Discovery and Pattern Analysis. This section presents an overview of the tasks for each step.

At least two log file formats exists: Common Log File format (CLF) and Extended Log File format ([16] for more details). Our university log file consists of these fields: Date, Time, client IP address, Method, URI stem, Protocol status, Bytes sent, Protocol version, Host, User Agent and Referrer.

3.1 Pre-processing

As said in [6], pre-processing *"consists of converting the usage, content, and structure information contained in the various available data sources into the data abstractions necessary for pattern discovery"*.

This step can break into at least four substeps: Data Cleaning, User Identification, Session Identification and Formatting.

Unneeded data will be deleted from raw data in web log files in the data cleaning step. When a user requests a page, the request is added to the Log File; but if this page contains images, javascripts, flash animations, video, etc., they are added to the Log file as well. Most of the time, these are not needed for pattern discovery and should be omitted from log files.

There are some methods for User Identification, the second phase: detecting cookies, Identd, through IP address and user name. W3C [17] define cookies as *"data sent by the server to the client, data locally stored in cookies and is sent to the server with each request"*. [18] and [19] use cookies to identify users. But they have two main problems: the users can lock the use of cookies, so server cannot store information locally in the user machine; other problem is the user can delete the cookies.

Identd can be used for user identification. It is a protocol defined in RFC 1413 [20]. It allows us to identify connected users with the unique TCP connection. The problem with Identd is that users should configure with this protocol.

Another way for user detection is through user names added in the log file in field authuser. But this field can be empty (default value dash(-)) according to server/user command.

At last we can identify users through their IP address registered in each record in log file. We used this method although it has several problems: different users can be registered with same IP address; same user can be registered with different IP addresses. But with the help of session identification we can identify a user with IP address and be sure of solving the first problem. The second problem is not important because in this paper, a specific user is not wanted. We want to recognize general user's pattern.

For Session Identification in third phase, first we should define session timeout. Different timeouts considered are between 25-30 minutes .The thirty minute timeout is based on the results of [21]. We assume 30 minutes session timeout for the experimental procedure.

And in the last phase of pre-processing step, we should display pre-processed data in a suitable format, for the second step, pattern discovery.

3.2 Pattern Discovery

As stated in [6], pattern discovery *"draws upon methods and algorithms developed from several fields such as statistics, data mining, machine learning and pattern recognition"*. Several methods and techniques have already been developed for this step as summarized below:

- *Statistical Analysis* such as frequency analysis, mean, median, etc.
- *Clustering* of users help to discover groups of users with similar navigation patterns (provide personalized Web content).
- *Classification* is the technique to map a data item into one of several predefined classes.
- *Association Rules* discover correlations among pages accessed together by a client.
- *Sequential Patterns* extract frequently occurring inter-session patterns such that the presence of a set of items s followed by another item in time order.
- *Dependency Modeling* determines if there are any significant dependencies among the variables in the Web.

We choose clustering to discover users' navigational patterns. Our clustering method is based on Ant-based Clustering algorithm explained in Section 3.4.

3.3 Pattern Analysis

Pattern Analysis is the final stage of WUM (Web Usage Mining), which involves the validation and interpretation of the mined pattern.

- *Validation*: to eliminate the irrelevant rules or patterns and to extract the interesting rules or patterns from the output of the pattern discovery process.
- *Interpretation*: the output of mining algorithms is mainly in mathematic form and not suitable for direct human interpretations.

3.4 Ant-based Clustering

Deneubourg et al. in [22] proposed ant-based clustering and sorting. In the case of ant-based clustering and sorting, two related types of natural ant behaviors are modeled. When clustering, ants gather items to form heaps. And when sorting, ants discriminate between different kinds of items and spatially arrange them according to their properties [23].

Lumer and Faieta in [24] proposed ant-based data clustering algorithm (shown in Figure 2), which resembles the ant behavior described in [22].

As shown in Figure 2, the agents (ants) and data are randomly initialized on a toroidal grid. By moving agents, data is sorted according to its neighbors.

The picking and dropping probabilities, given a grid position and a particular data item *i*, are computed using the density functions:

$$p_{pick}(i) = \left(\frac{k^{+}}{k^{+} + f(i)}\right)^{2} \quad (1)$$

$$p_{drop}(i) = \begin{cases} 2f(i) & if\ f(i) < k^{-} \\ 1 & otherwise, \end{cases} \quad (2)$$

ISBN: 978-989-95079-6-8

where $k+$ and $k-$ are constants, and $f(i)$ is a neighborhood function:

$$f(i) = \max(0, \frac{1}{\sigma^2} \sum_{j \in L} (1 - \frac{d(i,j)}{\alpha})) \quad (3)$$

where, $d(i, j) \in [0, 1]$ is a measure of the dissimilarity between data points i and j , $\alpha \in [0, 1]$ is a data-dependent scaling parameter, and σ^2 is the size of the local neighborhood L.

Handl & Meyer in [25] proposed an extension of this algorithm where the parameter α is adaptively updated during the execution of the algorithm.

We applied Handl & Meyer's Ant-based clustering algorithm for detecting user's patterns.

```
(1)  Procedure Lumer and Faieta
(2)     randomly scatter data items on the toroidal grid
(3)     randomly place agents on the toroidal grid
(4)     for t = 1 to max_iterations
(5)        j = random agent
(6)        move agent j randomly by stepsize grid cells
(7)        l = does agent j carry a data item?
(8)        e = is agent j's grid position occupied by a data
           item?
(9)        if (i = TRUE) and (e = FALSE) then
(10)          i = data item carried by agent j
(11)          drop = (random() ≤ Pdrop(i)) // see
              equations (2) and (3)
(12)          if drop = TRUE then
(13)             Let agent j drop data item i at its current
                 position
(14)          end if
(15)       end if
(16)       if (l = FALSE) and (e = TRUE) then
(17)          i = data item at agent j's grid position
(18)          pick = (random() ≤Ppick(i)) // see equations
              (1) and (3)
(19)          if pick = TRUE then
(20)             let agent j pick up data item i
(21)          end if
(22)       end if
(23)    end for
(24) end procedure
```

Figure 2: Lumer & Faieta 's ant-based clustering algorithm [24].

4 Experimental Procedure and Results

4.1 Experimental Procedure

We used our university (http://www.um.ac.ir/) web server logs of two weeks for the experimental procedure.

For the first step, i.e. pre-processing, we wrote a c++ program compiled using gcc without any optimization flags. As mentioned in section 3.1, this step contains four phases. First, we omit unneeded records from log file. The log file consists of image files (gif, jpg, bmp, jpeg …) and other unneeded resources like javascripts and errors. For user identification, we use IP address and session timeout of 30 minutes. So, a user with an IP address has 30 minutes to navigate in the web site. After a user's navigational sequence is extracted, it is displayed in a suitable format for the second step, Pattern Discovery. We then classify the URLs of the web site into 28 groups and assign a number to each group, as shown in Table 1. Then, each user's requested URL is substituted with its corresponding number. The output of this step is a file that consists of records, each record representing a navigational sequence of users in numbers.

For the second step, Patten Discovery, we used Ant-based Clustering algorithm based on [25]. Julia Handl's written source code is used in Java and changes are made according to our application. Each user's navigational sequence is defined as an array with 28 elements. The element i is 1 if the related user had seen one of the pages in group i; otherwise it is 0.

Dissimilarity of two sequences $s1$ at point i and $s2$ at point j in the grid is computed through the following equation:

$$d(i,j) = \frac{\sqrt{\sum_{k=1}^{28} (s1_k - s2_k)^2}}{28} \quad (4)$$

$d(i,j)$ becomes 1 when two sequences do not have any similar elements, and becomes 0 when they are exactly the same.

Table 1: Classification of URLs.

1	Web site content	2	Newsletter
3	Miscellaneous	4	Black board
5	Web services	6	Search
7	News	8	Help
9	FAQ	10	Societies
11	Publications	12	Abstract
13	Photos	14	Web links
15	RSS	16	People
17	Staff	18	Student
19	Faculty	20	Professor
21	User	22	Guest
23	Research	24	About university
25	Education	26	Download
27	English homepage	28	homepages

The output of this program is a grid that contains numbers: >-1 and =-1 indicates if there is/is not a data item, respectively. So, clusters should be extracted according to these numbers and size of the local neighborhood, σ^2. Figure 3 shows the positions of the data points in different phases of running this algorithm. Figure 3.a shows the distribution of data points at the first step of program execution. As the execution continues, Figures 3.b, 3.c and 3.d shows the results after 200, 400 and 600 iterations, respectively. Clusters are created through moving of the ants.

We examined several numbers of ants for the clustering step. The experiments shows that the smaller the number of ants, the slower will be the rate of convergence, but also better results in clustering the data items. On the other hand, the larger the number of ants, the faster will be the speed of convergence, but also weaker results in clustering. So, empirically, we choose number of ants to be %20 of the number of data items.

At last for the third step, Pattern Analysis, the results are shown in an interpretable way. The output of the second step

ISBN: 978-989-95079-6-8

is the clusters of users' navigational sequence. Each cluster may include lots of members, so we should represent a cluster with one pattern and display it in a suitable format for users.

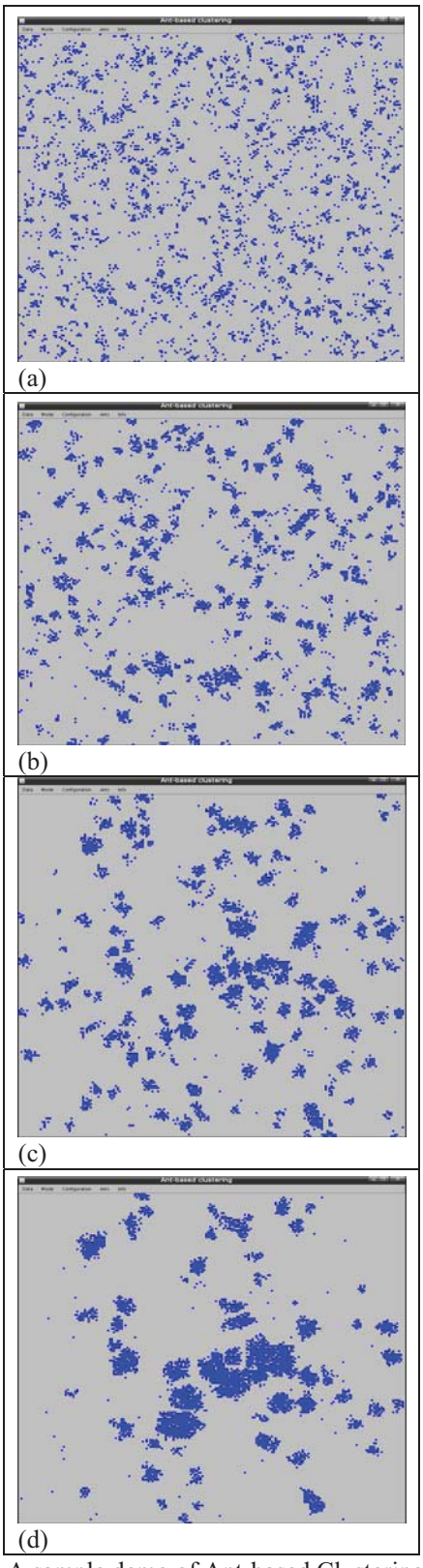

Figure 3: A sample demo of Ant-based Clustering of users' sequences.

The clustering algorithm results in clusters of similar sequences, which is a key element for sequence alignment. The alignment of sequences leads to a weighted sequence (as defined in [26]), represented as follows: $SA = \{I1 : n1, I2 : n2, \ldots, Ir, nr\} : m$. In this representation, m stands for the total number of sequences involved in the alignment. Ip ($1 \leq p \leq r$) is an itemset represented as ($xi1 : mi1 , \ldots xit : mit$), where mit is the number of sequences containing the item xi at the npth position in the aligned sequences. Finally, np is the number of occurrences of itemset Ip in the alignment. Figure 4 shows an example of the alignment process. The approximate sequential pattern can be obtained by specifying k: the number of occurrences of an item in order for it to be displayed. For instance, with the sequence $SA14$ from Figure 4 and $k = 3$, the filtered aligned sequence will be: {2, 4, 8}

To show the results, the aligned patterns that consist of numbers is re-substituted by the groups in Table 1.

S1={	2	4	5	8}
S2={	2		5	8}
SA12={	(2:2):2	(4:1):1	(5:2):2	(8:2):2}
SA12={	(2:2):2	(4:1):1	(5:2):2	(8:2):2}
S3={	2	4		8}
SA13={	(2:3):3	(4:2):2	(5:2):2	(8:3):3}
SA13={	(2:3):3	(4:2):2	(5:2):2	(8:3):3}
S4={	2	4		8}
SA14={	(2:4):4	(4:3):3	(5:2):2	(8:4):4}

Figure 4: Alignment processing example.

4.2 Results

The above mentioned procedure is applied to our university (http://www.um.ac.ir/) web server logs. All the experiments were performed on an Intel Core 2 Duo 2.5GHz PC running Linux (Mint).

The log files are collected in two different weeks in 2008: first week of June and the middle week of August each have 300 MB. Two different weeks are selected and the result of applying the proposed method on each week is compared. Applying ant-based clustering on each week, on average, took 300 seconds for 800 iterations. Average hourly web traffic for each group in these two weeks can be seen in Figures 5 and 6, respectively. The extracted behaviors from these two weeks are listed in Table 2 and Table 3, respectively. As shown in Table 2 and Table 3, behaviors of users in these two weeks were similar in most of the cases. One should notice that these results show just most frequent patterns of users' navigational behaviors, not all of the users' behaviors. Patterns of single navigations are not listed above, i.e. patterns that contain only one navigation.

Comparing Table 2 and Table 3 with Figure 5 and Figure 6, one could understand that pattern <Web site content → News > is the most popular one. <Professor → Homepages> is frequent, too. On the other hand, groups that do not exist in the patterns, like FAQ and Societies, have the least access in hours of a day, too.

Comparing our method to other methods, it has some advantages:

- It is simple. One only needs to define a suitable dissimilarity function for the clustering step and do

ISBN: 978-989-95079-6-8

not need to involve in complex mathematical relations.

- It does not depend on pattern length. Other similar methods are limited in terms of the length of the extracted patterns. The proposed method is able to extract patterns of any length.

Different patterns can be extracted depending on the occurrence of that page group in a cluster. So, even page groups that are less accessed can be extracted.

These kinds of results are useful for web site owners. They can put their advertisements in these sequences, because these are the most frequent ones. They are useful for page pre-fetching, too.

5 Conclusions

In this paper, we have proposed a new method to extract navigational patterns from web logs. Ant-based clustering has been used for this purpose. It needs a neighborhood function to be defined for. After the clustering is completed, alignment processing has been applied to the extracted sequences in each cluster and extract the representative for each cluster.

We apply the following procedure on our university web server logs for two different weeks (http://www.um.ac.ir/) and the results are satisfactory and true according to the hourly web traffic. These kinds of results are suitable for web site owners, for example, to put their advertisements there or to even change the structure of the web site according to users' navigational behavior.

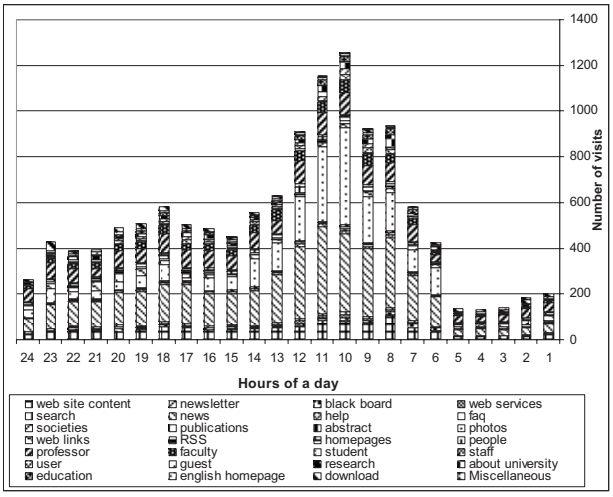

Figure 5: Hourly web traffic of the first week.

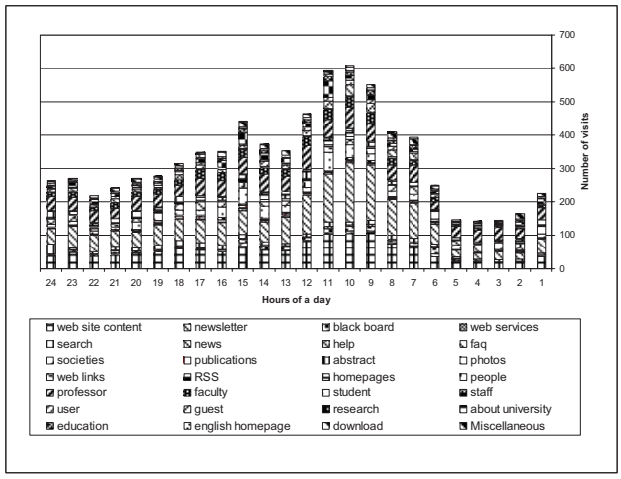

Figure 6: Hourly web traffic of the second week.

Table 2: Extracted behaviors from the first week.

Web site content	→	News						
Web services	→	Search						
Web site content	→	News	→	Faculty	→	Education		
News	→	Faculty	→	Education				
News	→	Professor						
Web site content	→	News	→	Professor	→	Homepages		
Web site content	→	News	→	Help	→	About university	→	Education
Web site content	→	News	→	Search	→	Photos	→	Web links → Professor → About university → Downloads
Web site content	→	Black board						
English homepage	→	Guest						
Professor	→	Homepages						
Professor	→	Faculty						
News	→	Faculty						

Table 3: Extracted behaviors from the second week.

Web site content	→	News						
Faculty	→	English Homepage						
Web site content	→	News	→	Users				
Web site content	→	Users						
Web site content	→	Search	→	News	→	Web links	→	Professor → About university → Download
Web site content	→	Faculty						
News	→	Professor	→	Homepages				
Faculty	→	Education						
Web site content	→	Photos	→	Web links	→	Professor	→	Homepages → English homepage → Download
News	→	Professor						
News	→	Download						

ISBN: 978-989-95079-6-8

References

[1] A. Abraham. Natural Computation for Business Intelligence from Web Usage Mining, *Proceeding of Seventh International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNAC2005),* pp. 3-11, 2005.

[2] F. Masseglia, P. Poncelet, and R. Cicchetti, An Efficient Algorithm for Web Usage Mining, *Networking and Information Systems Journal (NIS)*, 2(5-6), pp. 571-603, 1999.

[3] R. Cooley, Web Usage Mining: Discovery and Application of Interesting patterns from Web Data, Ph. D. Thesis, University of Minnesota, Department of Computer Science, 2000.

[4] P. Pirolli, J. Pitkow, and R. Rao, Silk From a Sow‘s Ear: Extracting Usable Structures from the Web, *Proceeding on Human Factors in Computing Systems (CHI’96)*, ACM Press, pp. 118-125, 1996.

[5] M. Spiliopoulou, and L.C. Faulstich, WUM: A Web Utilization Miner, *Proceeding of EDBT Workshop on the Web and Data Bases (WebDB’98)*, Springer Verlag, pp. 109-115, 1999.

[6] J. Srivastava, R. Cooley, M. Deshpande, and P.N. Tan, Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data, *SIGKDD Explorations*, 1(2), pp. 12-23, 2000.

[7] F. Masseglia , P. Poncelet, M. Teisseire, A. Marascu, Web usage mining: extracting unexpected periods from web logs, *Data Min Knowl Disc*, 16, pp.39–65, 2008.

[8] M. Spiliopoulou , L.C. Faulstich , K. Winkler , A data miner analyzing the navigational behavior of web users, *Proceeding of the workshop on machine learning in user modeling of the ACAI’99 international conference Creta*, Greece, 1999.

[9] F. Bonchi , F. Giannotti , C. Gozzi , G. Manco, M. Nanni , D. Pedreschi, C. Renso , S. Ruggieri, Web log data warehousing and mining for intelligent web caching , *Data Knowl Eng*, 39(2), pp. 165–189, 2001.

[10] B. Hay , G. Wets, K. Vanhoof , Mining navigation patterns using a sequence alignment method , *Knowl Inf Syst*, 6(2), pp.150–163, 2004.

[11] Zhu, J., Hong, J., Hughes, J.G. 2002 Using Markov chains for link prediction in adaptive web sites. *Proceeding of soft-ware: first international conference on computing in an imperfect world*, Belfast, UK, pp. 60–73, 2002.

[12] M. Nakagawa, B. Mobasher, Impact of site characteristics on recommendation models based on association rules and sequential patterns. *Proceeding of the IJCAI’03 workshop on intelligent techniques for web personalization*, Mexico, 2003.

[13] R. Srikant, R. Agrawal, Mining sequential patterns: generalizations and performance improvements. *Proceeding of the 5th international conference on extending database technology (EDBT’96)*, pp. 3–17, France, 1996.

[14] A. Mueller, Fast sequential and parallel algorithms for association rules mining: a comparison, Technical report CS-TR-3515, Department of Computer Science, University of Maryland-College Park, 1995.

[15] A. Abraham, V. Ramos, Web Usage Mining Using Artificial Ant Colony Clustering and Genetic Programming, *Congress on Evolutionary Computation (CEC)*, IEEE 2003.

[16] W3C extended log file format. Available at ,http://www.w3.org/TR/WD-logfile

[17] WCA. Web characterization terminology & definitions. Available at ,http://www.w3.org/1999/05/WCA-terms/ .

[18] M. Eirinaki, M.Vazirgiannis, Web Mining for Web Personalization, Athens University of Economics and Business, 2003.

[19] J. Huysmans, B. Baesens , J. Vanthienen , Web Usage Mining: A Practical Study, Katholieke Universiteit Leuven, Dept. of Applied Economic Sciences, 2003.

[20] RFC 1413. Identification Protocol. Available at ,http://www.rfceditor.org/rfc/rfc1413.txt .

[21] L. Catledge, J. Pitkow, Characterizing browsing behaviors on the World Wide Web, *Computer Networks and ISDN Systems*, 27(6), 1995.

[22] J. Deneubourg -L., S. Goss, N. Franks, A. Sendova-Franks, C. Detrain, L. Chrétien, The dynamics of collective sorting: robot-like ants and ant-like robots. *Proceeding of the first international conference on simulation of adaptive behavior*, pp. 356–365, MIT Press, 1991.

[23] J. Handl, B. Meyer, Ant-based and Swarm-based clustering, *Swarm Intelligence*, 1, pp. 95–113, 2007.

[24] E. Lumer, B. Faieta, Diversity and adaptation in populations of clustering ants. *Proceeding of the third international conference on simulation of adaptive behaviour*, pp. 501–508, MIT Press, 1994.

[25] J. Handl, B. Meyer, Improved ant-based clustering and sorting in a document retrieval interface. *Proceeding of the Seventh International Conference on Parallel Problem Solving from Nature*, Vol. 2439 of Lecture Notes in Computer Science, pp. 913-923, Germany: Springer-Verlag, 2002.

[26] H. Kum, J. Pei, W. Wang, D. Duncan, ApproxMAP: approximate mining of consensus sequential patterns, *Proceeding of SIAM International Conference on data mining*, CA, 2003.

ISBN: 978-989-95079-6-8