Fuzzy Inference Mechanism Based Approach for Multi-Dimensional Sequential Web Mining

Tansel Özyer, Reda Alhajj, Ken Barker
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada
{ozyer, alhajj, barker}@cpsc.ucalgary.ca

Abstract

There are several applications of sequential web mining, which is used to find the frequent subsequences in a web log in the World Wide Web (the web). We implemented a tool to analyze the sequential behavior of web log access patterns in multiple-dimensions. Sequences of frequent access patterns may change temporally and spatially. Based on the specified criteria like year, month, day, hours and location, the end-user is able to tune the minimum support threshold parameter intuitively using the fuzzy inference mechanism. Domain experts are can access several criteria, including minimum support threshold and number of accesses according to the user intuition, which is later, transformed into fuzzy inference parameters. We propose two different types of rule bases by considering the (support-minimum support, minimum support) and (support, minimum support), i.e., interval and case-based. To test our proposal, we used the web log dataset of the Department of Computer at the University of Calgary to analyze sequential access patterns of students during February and March carried out in the campus by taking the midterm dates into account. The results reported in this paper are promising; they demonstrate the applicability and effectiveness of the proposed approach.

1. Introduction

Data Mining is the notion of all methods and techniques that allow analyzing very large data sets to extract and discover previously unknown structures and relations out of such huge heaps of details. The information is filtered, prepared and classified so that it will be a valuable aid for decisions and strategies [1]. The web contains close to 350 million web pages and its daily growth rate is around one million web pages [2]. With the huge amount of information available online, the web is a fertile area for data mining research. This shows the necessity for analyzing and discovering the web. Web mining appears to be the crossroad of research from several communities, including the database information retrieval within AI and especially the sub-areas of machine learning and natural language processing [3]. Web mining is mining from web repositories. Its main tasks are defined as web structure, web content and web usage mining. Web content mining is related to the discovery of useful information from web content, data, documents and services, whereas web structure mining is to mining the structure of hyperlinks within the web itself (inter-document structure is used instead of web content mining, which is relevant to intra document structure). Web usage mining mines secondary data generated by users' interaction with the web. Web usage data includes data from web server access logs, proxy server logs, browser logs, user profiles, registration files, user sessions or transactions, user queries, bookmark folders, mouse-clicks and scrolls, and any other data generated from users access to the web [4]. Web usage mining process has three main steps: data preparation, pattern discovery, and pattern analysis phases [6].

Sequential pattern mining was first introduced by Aggrawal and Srikant (AprioriAll) [5]. It discovers and analyzes the frequent patterns in a sequence database. Thus, by mining frequent sequential patterns ordered correlations, we get remarkable sequences not necessarily adjacent to each other [5, 8-11,13-19]. Sequential pattern mining may be defined as follows: given a collection of transactions ordered in time and each transaction contains a set of items, the goal is to find sequences of maximal length with support above a user-specified threshold. A sequence is an ordered list of elements, an element being a set of items appearing together in a transaction. Elements do not need to be adjacent in time, but their ordering in a sequence must follow the time ordering of the supporting transactions [7]. Finally, sequential rule mining may contain different types of constraints [20].

Web access pattern is a sequential pattern in a large set of accesses, namely web log. Log analysis is the first step in Web usage mining. Several commercial and shareware systems exist for the log\traffic analysis, and the majority of them are statistical reports, e.g., Analog,WebLogs, WebLog, Ststat, Follow 2, and WUM, which can be characterized rather as a sequence miner [21]. Other type of applications include: to cluster user groups to analyze their navigational patterns and discover some correlations between them or correlations between the web pages and the users for recommendation and personalization purposes in adaptive or non-adaptive manner [22].

Many frequent patterns will be related to time and location due to the changing environment of web sites and user profiles accessing them. This is also valid for several different applications like customer behavior, stock market fluctuations, etc. Using the multi-dimensional information related to web accesses will be more effective since the classfied patterns are more effective. This operation helps experts in filtering required accesses [23]. The minimum support threshold may not be certain. It may be intuitively defined by domain experts. Fuzziness comes from the uncertain and imprecise nature of abstract thoughts and concepts [24].

The purpose of the work described in this paper is to propose a tool to aid domain-experts in undertaking the process. We assume that this tool has pre-specified criteria like year, month, and week of the day, time interval and location. We store all the information in a database and data is pulled from the web log repository per the specified criteria. Also, experts are eligible for defining the rule base for calculating minimum support threshold. All the frequent sequences are generated depending on the rules entered to the rule base. For the conducted experiments, we used the web access log of the Computer Science Department at the University of Calgary. The reason for giving alternative dimensions, different views of its look, for the criteria entry is that users may vary with respect to the time, web site changes in addition to that, especially, for the university load of the system may change based on exam dates. We propose two different types of fuzzy rule bases. The first one considers the difference support-minimum support value, and minimum support; the other considers support and minimum support values. In this tool, the user enters all the fuzzy linguistic values in terms of the number of accesses and later support and minimum support threshold values are re-scaled according to the maximum and minimum number of accesses given by domain expert. We modified the existing WAP implementation [27], and included these two types of fuzzy rule inference mechanisms in the evaluation of support with respect to user intuition, changing needs, and preferences.

The rest of the paper is organized as follows. An overview of the necessary background is presented in Section 2. The proposed approach is described in Section 3. Experimental results are given in Section 4. Section 5 is the conclusions.

2. The Necessary Background

Sequential Mining of Web Navigational Behavior:

Several accesses existing in the web log can be regarded as sequences of pairs of user and events. This is another version of sequential pattern mining applied on web logs. Let E be a set of pages existing in the web log. Sessions can be identified simply based on web accesses, by giving a threshold [25, 26].

According to the problem statement given by Pei Jian et al.[27]:

A web access sequence $S = e_1, e_2, \ldots, e_n$ $(e_i \in E)$ for $(1 \leq i \leq n)$, where n is the length of the access sequence, is also called an n-sequence and it is not necessary that $e_i \neq e_j$ for $(i \neq j)$ in an access sequence S, i.e., repetition is also allowed in a sequence. For example, *1-1-2* and *1-2* are two different access sequences in which *1* and *2* are different web pages.

An access sequence $S' = e'_1, e'_2, \ldots, e'_l$ is called a subsequence of S, and hence S is a super-sequence of S', denoted $S \supseteq S'$, if and only if there exist $(1 \leq i_1 < i_2 < \ldots < i_l \leq n)$, such that $e'_j = e_{i_j}$ for $(1 \leq j \leq l)$. Further, access sequence S' is a proper subsequence of sequence S, denoted $S \supset S'$, if and only if S' is a subsequence of S and $S' \neq S$.

In access sequence $S = e_1, e_2, \ldots e_k, e_{k+1}, \ldots, e_n$, suffix of S, $S_{suffix} = e_{k+1}, \ldots, e_n$ is a super sequence of the pattern $P = e'_1, e'_2, \ldots, e_l$ where $e_{k+1} = e'_1$ which is a subsequence of S, and $S_{prefix} = e_1, e_2, \ldots, e_k$ is called the prefix of S with respect to pattern P.

Given a set of access sequence $S_1, S_2, \ldots, S_m$ where each S_i $(1 \leq i \leq m)$ is called web access sequence. The support of access sequence S, denoted *sup(S)*, in the web access sequence is defined as:

$$\sup(S) = \frac{|\{S_i | S \subseteq S_i\}|}{m} \quad (1)$$

Table 1. Web Access Database

Transaction ID	Pages Visited in Time
100	1-2-3
200	1-3
300	2-1-2-1
400	1-3

We may have sequences of different length. Events (pages) can be repeated in an access sequence or pattern due to the frequent navigation behaviors of the users. Any pattern can get support at most once from one access sequence. For example: given the minimum support threshold 50% in Table 1: we obtain 1, 2, 3, 1-2, 1-3 as the frequent navigational behaviors of users. Finally, for the purpose of the study described in this paper, we extended the WAP algorithm described in [27], which in general has three basic steps:

1. Scan web access sequences once and find all frequent events.
2. Scan web access sequences to construct a WAP-tree over the set of frequent events.
3. Recursively mine the WAP-tree.

Fuzzy Logic:

A classical set is a set with a crisp boundary, i.e., there is no unambiguous boundary. Membership degree of an element can take two values 0 or 1, i.e., an object is either entirely in the set or not. Whereas a fuzzy set as its name implies is a set without crisp boundary; there is gradual transition from "belonging to a set" to "not belonging to a set", and this smooth transition is characterized by membership functions that give flexibility in modeling commonly used linguistic expressions. Fuzziness comes from the uncertain and imprecise nature of abstract thoughts and concepts [24].

Fuzziness may be summarized as follows: X is usually referred to as the universe of discourse. If X is a collection of objects denoted generically by x then fuzzy set A is defined as a set of ordered pairs:

$$A=\{(x,\mu_A(x)) \mid x \in X\} \qquad (2)$$

where μ_A is called the membership function that maps each object x of domain X to a continuous membership value between 0 and 1.

There are several classes of parameterized ways to define membership functions; triangular, trapezoidal, gaussian and bell functions are mostly used.

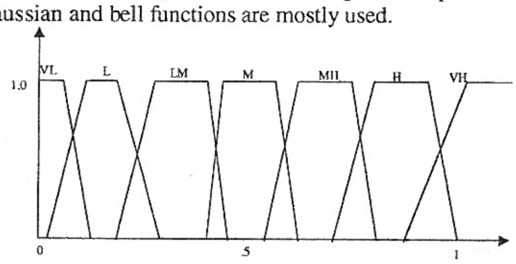

Fig. 1. Fuzzy Space Partitioned with Five fuzzy classes (VL, L, LM, M, MH, H, and VH).

During the definition of our rule base, we have two types of fuzzy space partitioned with seven linguistic variables: L_1={*Very Low, Low, Low Medium, Medium, Medium High, High, Very high*} and L_2={*Very Negative, Negative, Slightly Negative, Zero, Slightly Positive, Positive, Very Positive*}, denoted L_1={*VL, L, LM, M, MH, H, VH*} and L_2={*VN, N, SN, ZE, SP, P, VP*}. Each linguistic variable is a parameterized trapezoidal membership function (see Figure 1). Support and minimum support values may be entered by domain expert in terms of the given linguistic variables represented by trapezoidal membership functions, each specified by four parameters (a, b, c, d):

$$trapezoid(x;a,b,c,d) = \max(\min(\frac{x-a}{b-a},1,\frac{d-x}{d-c}),0) \qquad (3)$$

An object x may be member of different fuzzy classes with different membership degree values. We have two approaches for the rule base. An example of the first one is a conjunctive fuzzy rule R_q:

IF *minimum support* is A_1 and *support* is A_2 THEN pattern is c_q.

where R_q is the q-th fuzzy rule, x=(*minimum support, support*) is a 2-dimensional object of X, c_q is the consequent class (*frequent/infrequent*) and each A_{qi} is an antecedent fuzzy set (see Figure 1).

An example of the second one is a conjunctive fuzzy rule R_q:

IF *minimum support* is A_1 and Δ= *(support-minimum support)* is A_2 THEN pattern is c_q.

where R_q is the q-th fuzzy rule, x=(*minimum support, support- minimum support*) is a 2-dimensional object of X, c_q is the consequent class (*frequent/infrequent*). Note that the interval of the fuzzy variable interval given in Figure 1 will change from [0, 1] to [-1, 1] for this case.

If the degree of membership of an object with each corresponding antecedent A_{qi} is denoted by m_i then the firing strength μ_{A_q} of a rule using max-min composition is:

$$\mu_{A_q}=\min(m_1,m_2,\ldots,m_n). \qquad (4)$$

For the consequent part, c_q, it is also possible to specify the frequent and infrequent linguistic membership functions of the system in terms of trapezoidal functions as shown in Figure 2.

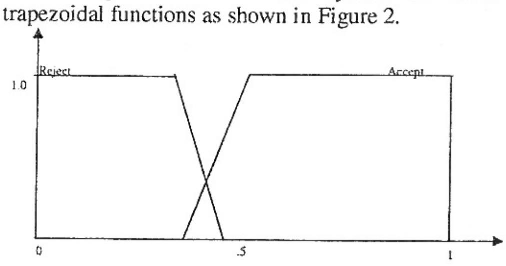

Fig. 2. Fuzzy Space Partitioned with Two fuzzy classes (Reject, Accept).

Fuzzy logic is used in the rule base in order to free domain experts from defining the fuzzy partitioning for both the antecedent and consequent and (s)he will be capable of controlling the support and minimum support and the acceptance policy for the frequency of the navigational behavior.

3. The Proposed Approach

Assume that we have a huge repository of web accesses; our system pulls data from the database by considering the criteria by the domain expert. The parameters are year, month, day of the week, hour interval and access from the campus or outside the campus. The architecture od the developed approach is shown in Figure 3.

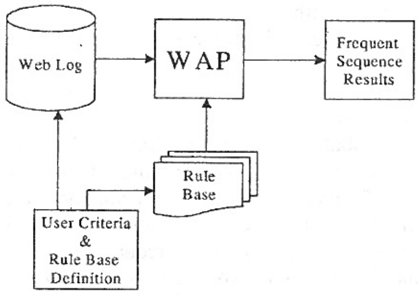

Fig. 3. Architecture of the proposed system

The reason for presenting several criteria to the domain expert is that the profiles of the users, even the load and the content of the web site and navigated pages of the web site may vary occasionally with respect to the location and access time: year, month, day of the week and hour interval and connection from the campus or outside the campus. Even the day time and night time access patterns may differ. For example, it is observable that users accessing the web site during the exam period will definitely navigate lecture notes and lab related pages. On the other hand, after the exam, they will be accessing the pages about the grade announcements. At other times, the web site may be informative about the profiles of the faculty staff, courses, etc. The number of criteria can be increased based on the requirements.

Web log of the Computer Science Department at the University of Calgary was organized in the common log format shown in Table 2 [28]. We partitioned web log accesses into sessions by using the algorithm described in [29], of course after preprocessing the log and removing unnecessary files (other than .htm, .html, .js, .php, .cgi, .doc, .txt, .pdf, .ps).

Table 2 Common Log Format

Remote Host	rfc931	Auth user	[date]	"request"	status	Bytes
192.168.20.38	-	-	[01/Jan/2002:13:07:21 -0700]	"GET /~ozyer/marks_sheet.html HTTP/1.0"	200	1234

Algorithm 1 Sessionization of the Web Accesses

Input : $L, \Delta t, |L|$
Output : $S, |S|$
For each L_i in L
  If method=GET and URL is a Web Page(htm, html, asp, js, php) then
    if $\exists S_k \in$ open sessions with $IP_k=IP_i$ then
      if $((time_i - end\_time(S_k)) < \Delta t)$ then
        $S_k=(IP_k, Pages_k \cup URL_i)$
      else
        Close_Session(S_k)
        Open_Session (IP_i,(URL_{ij}))
      end if
    else
      Open_Session (IP_i,(URL_{ij}))
    end if
end

Using Algorithm 1, web sessions are created and unnecessary objects existing in the web log are filtered out. After the creation of the web log database, the critical part of the system is the user criteria and rule base definition implementation.

It is probable that under different conditions and different criteria and the varying domain expert's intuition, frequent navigational behaviors will give different results. In the entry window shown in Figure 4, it is permitted both to pull the required data under the different dimensions as explained and the fuzzy rule base such as the definition of seven linguistic membership functions and the fuzzy rule membership function to count on the sequence as frequent and infrequent.

In this definition window, both rules and web access pages can be specified. Here, we made a modification. Support and minimum support are re-scaled by considering the entered number of access parameters for the linguistic parameters. When the sequence support values of web pages are calculated, the values are pronounced around $1\text{x}10^{-3}$ s. So, internally for both fuzzy spaces: we use (VL, VH) and (VN, VP) values, respectively, to re-calculate the minimum support and the support values. Apparently, we try to adapt those parameters to the user-given values and then we can play with the values around 10^{-1} s, instead of very small numbers. Besides, it will give a better way to the end user to express the linguistic parameters with numbers. After the related data and rules are specified, WAP is called with the given minimum support threshold value.

We propose two approaches for the rule definition. The first, named Case1, considers the case based analysis of the support and minimum support. For the example given in Figure 4, we have the rule base given in Table 3. Still it is allowed to enter different rules from the entry window after doing some checks. It should be noted that rules should be entered regarding the monotonicity of the sequences.

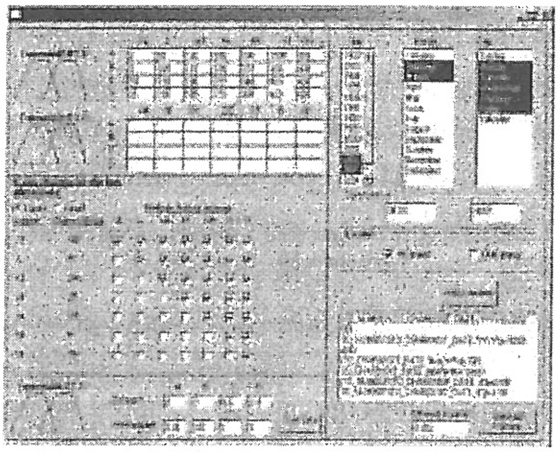

Fig. 4. User Criteria and Rule Base Definition

Table 3. Rule Base for Case1 (Frequent (F)-Infrequent (NF))

		Minimum support						
		VL	L	LM	M	MH	H	VH
Support	VL	F	F	F	F	F	F	F
	L	NF	F	F	F	F	F	F
	LM	NF	NF	F	F	F	F	F
	M	NF	NF	NF	F	F	F	F
	MH	NF	NF	NF	NF	F	F	F
	H	NF	NF	NF	NF	NF	F	F
	VH	NF	NF	NF	NF	NF	NF	F

The second approach, named Case2, considers the interval based analysis of Δ=(support – minimum support) and minimum support. For the example shown in Figue 4, we have the rule base given in Table 4. Here we may assume that we only take the negative part of Δ into account for this example, but the user can modify the selections in the cross-table, so it is not fixed.

Table 4. Rule Base for Case2 (Frequent (F)-Infrequent (NF))

		Minimum support						
		VL	L	LM	M	MH	H	VH
Δ	VN	NF	NF	NF	NF	NF	NF	F
	N	NF	NF	NF	NF	F	F	F
	SN	NF	NF	NF	F	F	F	F
	ZE	F	F	F	F	F	F	F
	SP	F	F	F	F	F	F	F
	P	F	F	F	F	F	F	F
	VP	F	F	F	F	F	F	F

Based on Table 3, let *L={VL, L, LM , M, MH, H, VH}* be the definition of the linguistics, and *(support,* *minimum support)* values be represented by (p_1,p_2). The linguistic variables for the result, i.e., to decide if a sequence is frequent or infrequent is $R=\{F,NF\}$.

$\exists (L_1,L_2) \rightarrow R_1 \mid (L_1,L_2) \in L\times L$ and $R_1 \in R$ and $R_1=F$ and we get (L_1,L_2) by,

$$arg\ max(min(\mu_{Li}(p_1),\ \mu_{Lj}(p_2))) \quad (5)$$

where $(1\leq i,j\leq n)$ and the membership function for the frequent sequence specified as:

$$\mu_F(p_1,p_2)=(min(\mu_{Li}(p_1),\ \mu_{Lj}(p_2))) \quad (6)$$

Based on Table 4, let $L=\{VL, L, LM, M, MH, H, VH\}$ be the definition of the linguistics, and $M=\{VN, N, SN, ZE, SP, P, VP\}$ and *(support, minimum support)* values be represented by (p_1,p_2). The linguistic variables for the result, i.e., to decide if a sequence is frequent or infrequent is $R=\{F,NF\}$.

$\exists (L_1,L_2) \rightarrow R_1 \mid (L_1,L_2) \in L\times M$ and $R_1 \in R$ and $R_1=F$ and we get (L_1,L_2) by and $\mu_F(p_1,p_2)$ with the same equations used above (5) and (6).

Also, we have the output variable having two linguistic values, namely Reject and Accept. $\mu_F(p_1,p_2)$ is used as an input to the membership function of Reject/Accept (see Figure 2.), so let O be defined as $O=\{Reject,\ Accept\}$, then we accept the sequence as frequent if $\mu_{Accept}(\mu_F(p_1,p_2)) > \mu_{Reject}(\mu_F(p_1,p_2))$.

After that, sequences are generated by the modified version of WAP [27]. WAP uses a highly compressed WAP-tree structure to mine the sequences efficiently and completely. Hence, non-reduntant accesses are eliminated during the process.

4. Experiments

We conducted our experiments on Intel Xeon 1.40 GHz CPU, 512 MB RAM Windows XP Dell PC. We used the web log of the department of computer science at the University of Calgary as the dataset.

During the pre-processing stage, the information stored in the web log is extracted and all the sessions are identified using Perl script with the time threshold specified as 30 minutes (see Algorithm 1). The entire users IPs are also kept confidential for privacy issues.

We used the SQL Loader utility to transfer the data to Oracle database 8i Personal Server Edition on the server side. We implemented the User Criteria and Rule Base Definition component in Microsoft Visual Basic 6.0. This module selects the accesses conforming to the first step of the algorithm in WAP in order to reduce the search space as much as possible and hence for reducing the memory requirements as much as possible. In addition to this, the existing code of WAP was modified by attaching a fuzzy inference mechanism. We entered the criteria as: *Year* in *(2002, 2003)*; *Month* in *(February, March)*; *Time* between *8:00-18:00* and *Accesses from the campus*.

For test purposes, we obtained 645,000 web accesses. According to the experiments, for minimum support threshold values (0.002, 0.003, 0.004, 0.005, 0.006, 0.007, 0.008, 0.009 and 0.01), for both cases, we got the results given in Table 5. Here, given threshold values are the actual values. Later in the program, they are recalculated according to the linguistic parameters given for the rule base.

Table 1 Experiment Results for WAP, Case1 and Case2

	Number of Sequences		
Minimum Support	WAP	Case1	Case2
0.002	3416	4743	3732
0.003	1105	2122	1779
0.004	496	1022	715
0.005	299	493	367
0.006	207	493	281
0.007	156	474	226
0.008	118	370	182
0.009	92	194	123
0.010	81	194	105

We had the inputs for $L=\{VL, L, LM, M, MH, H, VH\}$, $M=\{VN, N, SN, ZE, SP, P, VP\}$ *and* $O=\{Accept, Reject\}$ in Case1 and Case2 to be defined as trapezoid fuzzy functions. Also, we had the inputs for $L=\{VL, L, LM, M, MH, H, VH\}$, $M=\{VN, N, SN, ZE, SP, P, VP\}$ *and* $O=\{Accept, Reject\}$ in Case1 and Case2 to be defined as trapezoid fuzzy functions:

Case 1:

For *L:* *VL*=(0,0,70,95), *L*=(90,100,110,135), *LM*=(120,110,180,185), *M*=(180,190,255,270), *MH*=(260,315,340,365), *H*=(350,380,400,420), *VH*=(410,430,430,430)

For *O:* *Reject*=(0,0,0.65,0.75), *Accept* =(0.68,0.95,1,1).

Case 2:

For *L:* *VL*=(0,0,70,95), *L*=(90,100,110,135), *LM*=(120,110,180,185), *M*=(180,190,255,270), *MH*=(260,315,340,365), *H*=(350,380,400,420), *VH*=(410,430,430,430)

For *M*: *VN*=(-70,-70,-60,-55) *SN*=(-58,-50,-45,-40), *N*=(-42,-38,-35,-30), *ZE*=(-26,-22,-16,-12), *SP*=(-15,-4,4,15), *P*=(12,20,25,33), *VP*=(30,37,50,50).

For *O:* *Reject*=(0,0,0.35,0.81), *Accept* =(0.43,0.56,1,1).

5. Conclusions

This study aimed at proposing a new improvement on the existing implementation proposed in [27]. A fuzzy rule mechanism with two different approaches has been proposed to help domain-experts in discovering interesting patterns. The analysis of interesting patterns found with the fuzzy inference mechanism could give the navigational pattern behavior by taking domain-expert's intuition into account better by controlling several parameters leading to several cases or intervals. We tested both alternatives with different threshold values. Tuning the parameters also requires an iterative feedback mechanism. We are working on approaches to tune the fuzzy parameters with the selected minimum threshold value set automatically. Another improvement would be regarding the limitations of WAP since it runs in the main memory, a pre-processing was done, although it may not be enough for some cases.

References

[1] Grabmeier J., "Rudolph Techniques of Cluster Algorithms in Data Mining," *Data Mining and Knowledge Discovery*, Vol.6, 303–360, 2003.

[2] Fayyad U., Piatetsky-Shapiro, G., and Smyth P., "The KDD Process for Extracting Useful Knowledge from Volumes of Data:," *Communications of ACM*, Vol.39, No.11, pp.27-34, 1996.

[3] Kosala R., Blockeel H., "Web Mining Research: A Survey: *SIGKDD: SIGKDD Explorations: Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining. ACM*, Volume 2, 2000

[4] Pal S. K., Talwar V., and Mitra P., "Web Mining in Soft Computing Framework: Relevance, State of the Art and Future Directions," *IEEE Trans. On Neural Networks*, Vol.13, No.5, 2002.

[5] Agrawal R., and Srikant R., "Web usage mining Discovery and applications of usage patterns from web data," *Proc. of IEEE ICDE*, Taipei, 1995.

[6] Srivastava J., Cooley R. Deshpande M., and Tan P-N., "Discovery and applications of usage patterns from web data," *SIGKDD Explorations*, Vol.1, No.2, pp.12–23, 2000.

[7] Berendt B., and Spiliopoulou M., "Analysis of navigation behaviour in web sites integrating multiple information systems," *VLDB*, Vol.9, pp.56–75 ,2000.

[8] Agrawal R., and Srikant R., "Mining Sequential Patterns: Generalizations and performance improvements," *Proc. of EDBT*, Avignon, 1996.

[9] Mannila H., and Toivonen H., "Discovering generalized episodes using minimal occurences," *Proc. of ACM KDD*, pp.146–151, 1996.

[10] Wang K, Discovering patterns from large and dynamic sequential data: *Intelligent Information Systems*, Vol.9, pp.8–33, 1997.

[11] Spiliopoulou M., "The laborious way from data mining to web mining," *Int. Journal of Comp. Sys., Sci. & Eng., Special Issue on "Semantics of the Web"*, Vol.14, pp.113–126, Mar. 1999.

[12] Agrawal R., Srikant R., Mining quantitative association rules in large relational tables: *Proc. of ACM-SIGMOD*, pp.1-12, 1996.

[13] Lu H., Han J. and Feng L., "Stock movement and n-dimensional inter-transaction association rules," *Proc. 1998 SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, pp.121-127, Seattle, 1998.

[14] Bettini C., Wang S. X. and Jajodia S., "Mining temporal relationships with multiple granularities in time sequences," *Data Engineering Bulletin*, Vol.21, pp.32-38, 1998.

[15]Ozden B., Ramaswamy S. and Silberschatz A., "Cyclic association rules," *Proc. IEEE ICDE*, pp. 412-421, 1998.

[16]Han J., Dong G. and Yin Y., "Efficient mining of partial periodic patterns in time series database, "*Proc. of IEEE ICDE*, pp.106-115, 1999.

[17]Pei J. et al. PrefixSpan, "Mining sequential patterns efficiently by prefix-projected pattern growth," *Proc. of IEEE ICDE*, 2001.

[18]Zaki M., "SPADE: An efficient algorithm for mining frequent sequences," *Machine Learning*, Vol.40, pp.31–60, 2001.

[19]Garofalakis M., Rastogi R., and Shim K., "Spirit: Sequential pattern mining with regular expression constraints," *Proc. of VLDB*, 1999.

[20]Pei J., Han J., Wang W., "Mining Sequential Patterns with Constraints in Large Databases," *Proc. of ACM CIKM*, pp.18-25, 2002.

[21]Spiliopoulou M., and Faulstich C., "WUM: A tool for web utilization analysis," *Proc. of EDBT Workshop WebDB, LNCS1590*, pp.184-203. Springer-Verlag, 1999.

[22] Eirinaki M.and Vazirgiannis M., "Web Mining for Web Personalization," *ACM Trans. on Internet Technology*, Vol.3, No.1, 2003.

[23] Pinto H., Han J., Pei J., Wang K., Chen Q., Dayal U., "Multidimensional Sequential Pattern Mining," *Proc. of ACM CIKM*, 2001.

[24] Jang J.-S. R and Sun C.-T., "Neuro-Fuzzy Modeling and Control," *IEE Proceedings*, Vol.83, pp.378-406, 1995.

[25] Catledge L., Pitkow J., "Characterizing Browsing behaviors on the World Wide Web," *Computer Networks and ISDN Systems*, Vol.27, No.6, 1995.

[26] Srivastava J., Cooley R., Deshpande M., Tan P., "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," *SIGKDD* Explorations, Vol.1, No.2, 2000.

[27] Pei J., Han J., Mortazavi-Asl B., and Zhu H., "Mining Access Patterns Efficiently from Web Logs," *Proc. of PAKDD*, Kyoto, 2000.

[28] Logging Control in W3C httpd, Available at:
http://www.w3.org/Daemon/User/Config/Logging.html

[29] Frias-Martinez E., and Karamcheti V., A Prediction Model for User Access Sequences: *Proc. of WEBKDD*, 2000.