Extraction of users' behavior patterns for web personalization

Sowbhagya M P
Assistant Professor,
Dayananda Sagar Academy of Technology and Management

Dr.Yogish H K
Professor,
M S Ramaiah Institute of Technology,

Dr.G T Raju
Professor
SJCIT, Chikkaballapurs

Abstract

Frequent Sequential Patterns (FSP) from Web Usage Data (WUD) are essential for analyzing and comprehending user behavior to raise the caliber of services provided by the World Wide Web (WWW). One method for decreasing Web latency and hence improving web retrieval is web personalization. Here, PrefixSpan, SPADE, and WAP-Tree are just a few of the FSP mining techniques we look at for the mining of FSPs from the WUD of an academic website ranging from weekly to quarterly. The number of FSPs each of these FSP algorithms produces with a specific minimal support has been used to evaluate each algorithm's performance. The ***PrefixSpan*** *FSP mining technique outperforms the SPADE and WAP-Tree algorithms, according to experimental results.*

1. INTRODUCTION

Finding FSPs requires seeing inter-transaction patterns when one item appears in the time-stamp ordered transaction set before a collection of other items do. The duration of a user's visit is tracked in WUD. In this case, during the data preprocessing process, a time period will be selected and related to a transaction as the time stamp associated with it. The deployment of web personalization techniques to improve server performance by lowering perceived latency for users and so enhancing the standard of Web services is made possible by the use of extracted FSPs from WUD that help in analysing and forecasting user behaviour.

We look at different FSP mining methods to extract FSPs from websites for time frames ranging from weekly to quarterly and provide a mechanism for putting online personalisation into practice. We first preprocess the raw WUD to obtain each Web user's session database, which contains a list of the Web users (IPs) and the sessions that correspond to them. Then, from this session database, we extract FSPs using FSP mining methods, and we examine them to determine their periodicity.

The essay's remaining sections are organized as follows. Related work can be found under Section 2. A idea and problem statement are both found in Section 3. The SPM module, the FSP mining algorithm, and the prefetching rule generation technique are all presented in Section 4 (sections 4.2, 4.3, and 4.4). Section 5 provides experimental findings and analyses.

2. RELATED WORK

It looked for repeating patterns in the succession of products that customers had bought through time-ordered transactions. Later, the scope of its employment was expanded to include sophisticated tasks like predicting web users, network detection, DNA research, etc. Sequential pattern mining has been approached in a variety of ways. Generally speaking, they fall into two categories: (i) Apriori-based and

(ii) Frequent Pattern growth (FP-growth)-based. Apriori-based mining algorithms such as Apriori-all, GSP [3], SPADE [4], LAPIN-SPAM [5], and LAPIN [6] are used to execute several database scans. These mining algorithms are typically ineffective because a pattern of a given size necessitates scanning the database n times. *The original database is reflected in a tree-based representation used by FP-growth based mining algorithms as FreeSpan, BIDE [7], COBRA [8], PrefixSpan [9], UDDAG [1], etc. Two scans are needed to build the tree.* The incremental analysis method makes it simple to incorporate changes in the source database in the tree. It is possible to extract web access patterns from WUD by using these sequential pattern mining approaches.

Using the Apriori-all technique, which initially stores the original web access sequence database for storing non-sequential data, Gaol [11] investigated user patterns. This is predicated on the idea that fewer users are likely to undertake combinations of these actions as there are more combinations produced, and vice versa. Although this method is straightforward and easy to use, Apriori-all algorithms have been found to be the least effective at mining sequential patterns, making them obsolete.

By using substantially compressed access sequences and the addition of a sub-tree structure, the modified WAP-tree was introduced by Xiaoqiu et al. [13] to avoid the repeated construction of conditional WAP trees and to produce maximal sequences. Better WAP-tree performs better than traditional WAP-tree in both time and space, and it demonstrates greater stability as pattern lengths change. Additionally, the transaction database only needs to be scanned twice in order to mine common access sequences using the WAP-tree. Yang et al.'s Top Down Mine (TD-mine) approach, which makes use of the WAP tree data structure, was created for the aim of mining web access patterns [14]. The WAP tree can be examined both top down and bottom up for the extraction of common access patterns. TD-mine mines patterns close to commonly used nodes by traversing the tree from root to leaf nodes using a header table.

Vijayalakshmi, et al. [16] created an enlarged version of PrefixSpan called EXT-Prefixspan method to extract the Constraint-based multidimensional frequent sequential patterns in online usage mining by filtering the dataset in the presence of various pattern constraints. Priority EXT Then Span mines the full set of patterns, but significantly reduces the amount of work needed to produce candidate sequences. This facilitates efficient processing and greatly reduces the anticipated database size. EXT-PrefixSpan can be used to mine frequent sequential patterns of a multi-dimensional type from any web server log file in order to obtain the frequent web access patterns. EXT-PrefixHowever, Span does not outline any particular restrictions that are to be taken into mind.

The CIC-PrefixSpan, a modified form of PrefixSpan that combines PrefixSpan and pseudo-projection to mine and construct Maximal Sequential patterns, was proposed by Wu et al. in [17]. For effective Web sequential pattern mining, the user sessions are first preprocessed to separate them into human, crawler, and resource-download user sessions. The non-human user sessions are then filtered out, leaving the human user sessions, and the transactions are found using Maximum Forward Path (MFP). Utilising CIC-PrefixSpan reduces memory usage and prevents the creation of multiple projections to determine the user's access path tree's most common path. It is demonstrated that when compared to GSP and PrefixSpan, CIC-PrefixSpan produces precise patterns with excellent efficiency and rapid execution. CIC-PrefixSpan cannot, however, mine a pattern's common substructures.

Single Level Algorithm is a brand-new pattern mining algorithm created by Verma et al., [18] for the extraction of behavioural patterns. These patterns are employed to produce suggestions for web users during runtime. The dynamic adaptation of narrowly focused websites with several web pages is taken into consideration when creating the Single Level Algorithm. It integrates preprocessing, mining, analysis to ultimately forecast user behavior, making it effective for certain websites and very scalable. The Apriori algorithm is demonstrated to be less effective than this one. On the other hand, when used on very large Web log datasets, preprocessing can be time-consuming and challenging to integrate with mining and analysis.

Each user has a profile that is created and enhanced with additional domain-specific information aspects to provide a comprehensive perspective of the observed mass usage modes. This framework includes the seen pages, search engine queries, and inquiring and queried companies of a set of individuals with comparable access activities. By assigning certain new sessions to persistent profiles, updating these profiles, and excluding the majority of sessions from further analysis, the mining is concentrated on actually new sessions. But this framework cannot be scaled.

The compressed frequent patterns have been kept in transaction data bases and mines as projected FP-tree data, according to [31]'s Fp growth strategy for frequent pattern mining without candidate creation. It efficiently uses a pattern growth technique to mine huge databases for common patterns. Performance findings demonstrate the efficacy and scalability of the Fp growth method for mining both short and long frequent patterns faster than the Apriori technique.

Even while many academics have made significant contributions to the extraction of FSPs for diverse applications, very few have done so for WUD. In this study, we investigate and compare the performance of PrefixSpan, SPADE, and WAP-Tree FSP mining techniques on WUD. The approach that produces the best results will be used for further processing.

3. STATEMENT OF THE PROBLEM

"Given a session database S, we construct Transaction database of WUD, and Apply FSP mining algorithms to extract FSPs".

Methodology

We preprocess the data and produce a session database SD with a specific WUD, which is shown in Table 4. The session database is a collection of tuples, each of which contains the IP access sequence of the user. Web pages A, B, C, and D were visited in the order listed in Table 2 by the user with the IP address 1.0.1.2: A, B, C, D, E, and F are a collection of distinctive Web pages that have been accessed by various persons. Together, they make up A (A,C) B D>. When brackets () are omitted, it is presumed that the user accessed single page during each session. But in this instance, the order (A, C) demonstrates that the user visited the two Web pages in that particular order within a single session.

Table 1. Pages Visited

Pages	Number of Accesses
A	5
B	4

C	5
D	4
E	3
F	2
G	1
H	1

Table 2. Illustration Sessions Database

User (IP)	Sequences
1.0.1.2	<A (A,C) B D>
1.0.1.3	<A D (E, F)>
1.0.1.4	<(B, D) C F>
1.0.1.5	<(C, E) (A, B, C, D) >
1.0.1.6	<A B C D E>

The sessions database is mined for web sequential patterns using various FSP mining techniques. Database transformation which is used to eliminate pages that are infrequently utilised or on which MinSup is less than 2. The frequently occurring objects with MinSupport 2 in Table 1 are (A), (B), (C), (D), (E), (F), and (A, C), (B, D). We only require these patterns, so we can eliminate the others by substituting non-negative values for them, such as (A)-1, (A, C)-2, (B)-3, (B, D)-4, (C)-5, (D)-6, (E)-7, and (F)-8. We also provide every IP a special identification for ease of representation. Table 3 displays the converted database.

Table 3. Following database transformation.

User (IP address)	Orders(Sequences)
P	<1, 2, 3, 6>
Q	<1, 6, (7, 8)>
R	<4, 5, 8>
S	<(5, 7) ,(1, 3, 5, 6) >
T	<1 ,3 ,5 ,6, 7>

4. ARCHITECTURE OF THE SYSTEM

The suggested work's system design is shown in figure 1. It features modules for sequential pattern mining and preprocessing.

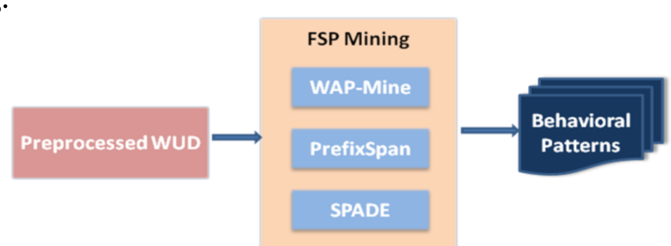


Fig. 1: Framework for the System

Weblogs

A web log, commonly referred to as WUD, on a web server records transactions made by web users.

Preprocessing

- ***Cleaning of Data:***

The web usage data is cleaned up in this step; Only entries that were initiated by humans are retained, including crawler entries and URLs with the extensions HTML, XHTML, PHP, and JSP. Few elements, such as user IP addresses (referrers), URLs, dates, and timestamps, as well as the type of file in the URL (text, picture, or script), are crucial to the mining process; hence, the others can be disregarded once the crucial values have been retrieved or aggregated.

- ***Identification of User:***

In order to identify between various users and their transactions, the cleaned database is grouped by a number of IP addresses and sorted by Date and Time for each IP.

- ***Identification of Session***

In order to provide the precise order of each user's actions, the user activity records of each user are divided into sessions through the process of session identification. The navigation history of a user makes up the actions of a session. Combining these sessions into a session database is the following stage. By giving each session a distinct identity, the user activity records are split into sessions. Each user has their own session, and if they stay for longer (15–20 minutes), they also have their own session. In Table 1, Shown are typical user sessions for an academic website. The information for a 50-page academic website's web transactions is shown in Table 2.

Table 4: An academic website's sample user sessions.

Session Id	IP Address	Date &Time	URL Accessed
1	117.216.148.95	2014-04-30 17:57:03	http://www.msrit..ac.in/Invitations.html
1	117.216.148.95	2014-04-30 17:58:20	http:// www.msrit..ac.in /facilities.html
2	70.39.187.99	2014-04-30 18:03:38	http:// www.msrit..ac.in /hostel.html
2	70.39.187.99	2014-04-30 18:03:39	http:// www.msrit..ac.in /hostel.html
3	210.212.194.2166	2014-04-30 18:27:52	http://www.rnsrit.ac.in rank_holders.html
4	115.118.176.83	2014-04-30 18:28:42	http://www.rnsrit.ac.in /Admissions.html
4	115.118.176.83	2014-04-30 18:28:43	http://www.rnsrit.ac.in /Admissions.html
5	117.208.191.46	2014-04-30 18:36:50	http://www.rnsrit.ac.in /cse-dep.html
5	117.208.191.46	2014-04-30 18:36:51	http://www.rnsrit.ac.in /cse-dep.html

Table 5: Transaction data for a 50-page academic website

Record Number	User id	Pattern id
1.	U5	P1,P9,P24
2.	U23	P32,P24,P5
3.	U34	P36,P14
4.	U36	P23,P14

5.	U4	P46,P14,P27
6.	U27	P9,P48,P24
7.	U17	p2,P16
8.	U1	P31,P26,P22,P12
9.	U2	P1,P9,P14
10.	U35	P20,P47
11.	U12	P44,P41
12.	U35	P18,P44
13.	U12	P23,P47

5. Frequent Sequential Pattern MINING ALGORITHMS

***SPM*:** The SPM algorithm, sometimes known as the GSP (Generalised Sequential Pattern) method [6], is a system for mining sequential patterns based on the Apriori concept. Compared to Agarwal's Apriori algorithm, it is considerably faster [2]. When it comes to the quantity of transactions per data sequence and the quantity of objects per transaction, it offers excellent scaling features. But because it cannot provide more candidate sequence and because several database scans are required because each candidate's length increases by one with each database scan, it is ineffective for mining huge sequences of databases with numerous patterns or long patterns. Below is the SPM pseudo code.

Sequential Pattern Mining (*SPM):*

Input:

D = Collection of Transactions; U = Generation of Candidates

P = the group of often occurring 1- s; K = Length

Output: *Frequently Occurring Sequential Patterns in D*

1. *Data base scanning session*
2. *Verify the existence of a specific Pid after scanningLet k=1;*
3. *Check for pattern frequently visited*
4. *Do while P(k)!=null;*
5. *Create candidate sets of candidate k+1 sequences (uk+1 set).*
6. *If U_{k+1} is not empty, find P_{k+1} i.e. the length range (k+1) sequential patterns*
7. *k=k++;*
8. *End do*

***SPADE*:** The sequences are presented in vertical order rather than horizontal format utilising the Apriori-based sequential pattern mining method SPADE (Sequential Pattern Discovery utilising Equivalence classes). This algorithm reduces the cost of computing support counts by utilising the ID-List approach. It is made up of ID-List pairs in which the first value designates a customer sequence and the second value designates a transaction inside of that sequence. The method uses either depthfirst or breadth-first search strategies while looking for new sequences. In order to mine, databases must be scanned. The SPADE pseudo code is shown below.

Input:

D = is a list of sequences.. SID = Sequence ID; EID = ID of Event; P = Pages in order; S = Support

Output: *Frequently Occurring Sequential Patterns in D*

1. *First Scan S and transforms it into vertical format.*
2. *Let P1= frequent 1-sequences*
3. *Let P2=Frequent 2-sequences*
4. *Check equivalence classes belongs for all 1 sequences (with minimum support*
5. *The second scan verifies that all two sequences have equivalence classes.For each [S] belongs and do*
6. *Look for common sequences P;*
7. *End*

Frequent pattern Growth: There are two ways to find FSPs: the FP tree approach and the vertical data format method. Without the need for candidate generation, FP growth mines the complete collection of frequently occurring web pages using a divide-and-conquer technique. First, it compresses the database of often occurring sites into a frequently occurring pattern tree, which keeps track of the page set association data. The compressed database is then divided into several conditional databases. Set of transactions in TID page set format and TID set is a set of transactions IDs comprising the pages, can be mined for frequently occurring pages utilizing vertical format and FP growth algorithms. The primary benefit of vertical format is that it outperforms the Apriori algorithm. Below is a pseudo code for FP growth.

Frequent Pattern Growth:

Input:

D: Denotes database of Transactions.

Output: Frequent Sequential Patterns in D

1. *Let P denotes the set of page set*
2. *Let k=1;*
3. *Do while P(k)!=Null;*
4. *Construct candidate set* U_{k+1}
5. *TID set pf P[k] && P[k+1] as same Transaction ID's*
6. *If* U_{k+1} *is not empty, find* T_{k+1} *, i.e the set of sequences*
7. *k=k+1;*
8. *End do;*

6. Discussions and Results from Experiments

Java has been used to implement the suggested algorithms on the WUD of the academic website rnsit.ac.in. Every series in the dataset represents a user's page views over a period of time, ranging from weekly to quarterly. The events in the sequence are responses to page requests made by users. The server logs do not record the page requests handled by caching mechanisms; hence the data does not contain them. Research is being done to compare the effectiveness of various FSP mining techniques. For a number of factors, including the total number of access sequences, the total number of pages, and the average lengths of sequences, the suggested techniques showed strong scale-up characteristics.

Table 6. Statistics show that a website with 25 pages has a baseline degree of support of 2.

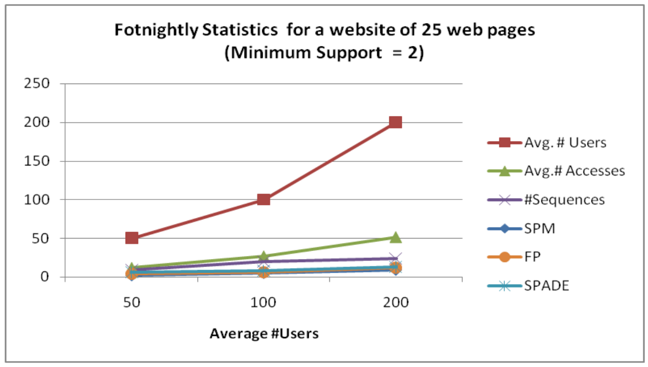
Period	Avg. No. Users	Avg.No. Accesses	No.Sequences	FSPs		
				Spade	PrefixSpan	WAP-Tree
Weekly	25	8	6	0	2	1
	50	11	8	2	4	3
	100	21	15	5	6	6
Fortnightly	50	12	9	2	6	4
	100	27	20	5	8	6
	200	52	24	9	13	12
Monthly	100	24	18	8	13	12
	200	51	23	22	28	27
	500	134	25	51	56	53
Quarterly	200	54	23	24	38	26
	500	132	24	54	56	53
	750	213	26	72	80	71
	1000	262	30	113	114	113

Figure 2 displays statistics such as the typical user count, average number of accesses, and average number of sequences for a website with 25 pages, and the total number of FSPs generated throughout time periods ranging from weekly to quarterly using different FSP-using techniques including SPADE, SPM, and FP growth.

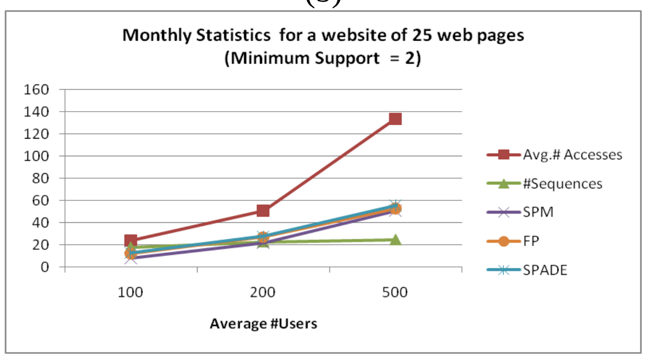
Table 7. A website with 50 pages has statistics for a basic level of support of 2

Period	Avg. No. of Users	Avg. No. of Accesses	No. of Sequences	FSP		
				SPADE	PrefixSpan	WAP-Tree
Weekly	25	7	3	2	6	4
	50	13	5	4	10	8
	100	24	11	11	14	13
Fortnightly	50	12	8	6	9	8
	100	31	19	10	2	10
	200	53	47	13	16	15
Monthly	100	29	21	8	7	9
	200	47	33	15	18	16
	500	274	43	45	45	44
Quarterly	200	54	36	13	14	13
	500	141	50	56	58	47
	750	193	52	76	77	74
	1000	262	53	115	117	113

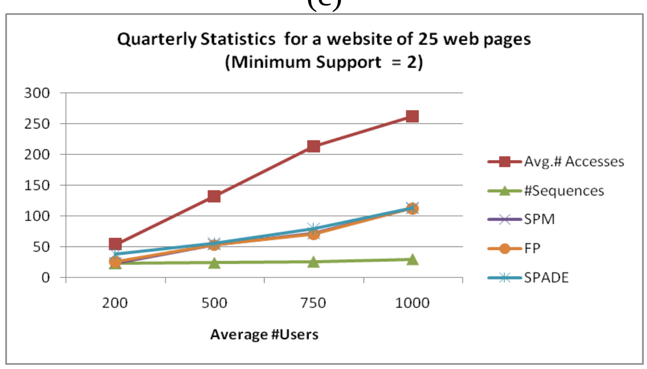
(a)



(b)



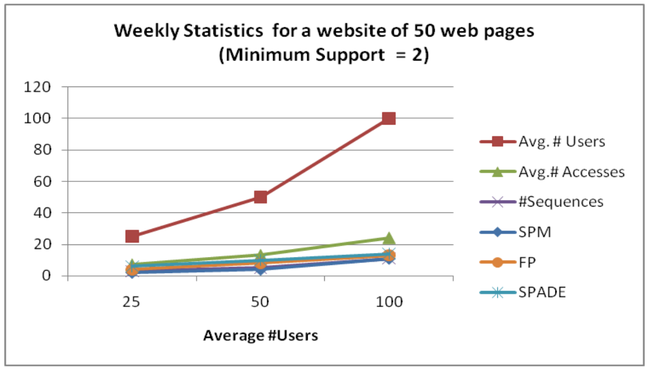
(c)



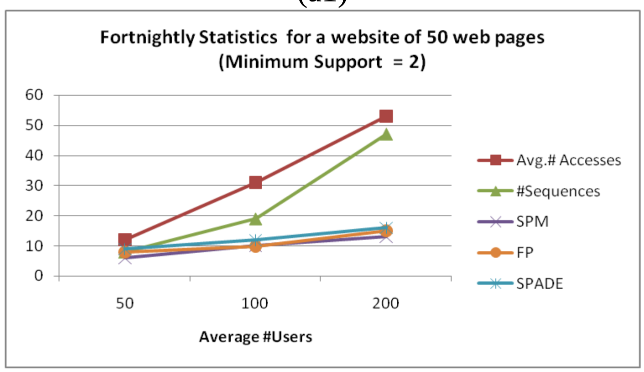
(d)

Figure 2. Statistics for a website with 25 web pages that indicate the average number of users, accesses, and sequences

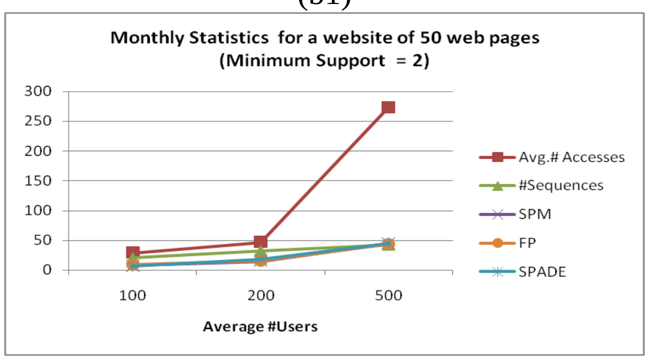
FSPs constructed over the course of a time period utilising the Prefix Span, SPADE, and WAP-Tree algorithms with a minimum support of two Weekly, fortnightly, monthly, quarterly, etc.



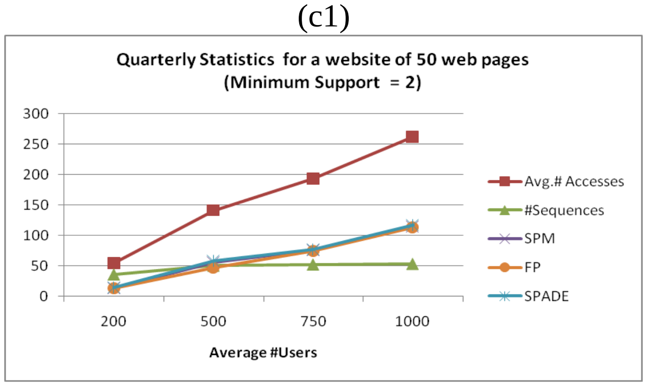
(a1)



(b1)



(c1)



(d1)

Figure 3. A 50-page website's statistics, including the typical number of users, accesses, and sequences

FSPs produced during a time period using the Prefix Span, SPADE, and a minimum support of two WAP-Tree algorithms Weekly, fortnightly, monthly, quarterly, etc.

It was also noted that fortnightly data perform better than weekly statistics and that as the number of users grows, so do the number of accesses. Improving the quantity of consecutive patterns.

Conclusion

Here, the Web sequential patterns that produce Web users' behavioral patterns utilizing various FSP mining techniques are presented. Multiple scans of the sequence database are necessary for sequential pattern mining approaches based on Priory, which produce a large number of candidate sets for lengthy web access sequences. Due to this issue, these algorithms are incapable of handling huge sequence sets with lengthy sequences, particularly web logs. Prefix Span has more benefits than the others, including speed, less database scans, and great performance.

References

[1]. Jinlin Chen, (2010) "An UpDown Directed Acyclic Graph Approach for Sequential Pattern Mining", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 7, pp. 913-928.

[2]. Ding-An Chiang, Cheng-Tzu Wang, Shao-Ping Chen. & Chun-Chi Chen, (2009) "The Cyclic Model Analysis on Sequential Patterns", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, No. 11, pp. 1617 – 1628.

[3]. Yu Hirate. & Hayato Yamana, (2006) "Generalized Sequential Pattern Mining with Item Intervals", *Journal of Computers*, Vol. 1, No. 3, pp. 51-60.

[4]. Mohammed J. Zaki, (2001) "Spade: An Efficient Algorithm For Mining Frequent sequences", *Machine Learning*, Vol. 42, pp. 31-60.

[5]. Zhenglu Yang. & Masaru Kitsuregawa, (2005) "LAPIN-SPAM: An Improved Algorithm for MiningSequential Pattern", *IEEE International Conference on Data Engineering Workshops*, pp. 1222-1226.

[6]. Zhenglu Yang, Yitong Wang. & Masaru Kitsuregawa, (2007) "LAPIN: Effective Sequential Pattern Mining Algorithms by Last Position Induction for Dense Databases", *International Conference on Database systems for advanced applications*, pp. 1020-1023.

[7]. Jianyong Wang, Jiawei Han. & Chun Li, (2007) "Frequent Closed Sequence Mining without Candidate Maintenance", *IEEE Transactions on Knowledge and Data Engineering*, Vol.19, No. 8, pp. 1042-1056.

[8]. Kuo-Yu Huang, Chia-Hui Chang, Jiun-Hung Tung. & Cheng-Tao Ho, (2006) "COBRA: Closed Sequential Pattern Mining Using Bi-phase Reduction Approach", *International Conference on Data Warehousing and Knowledge Discovery*, pp. 280-291.

[9]. Xiaorong Cheng. & Hong Liu, (2009) "Personalized Services Research Based On Web Data Mining Technology", IEEE International Symposium on Computational Intelligence and Design, pp. 177-180.

[10]. Ford Lumban Gaol, (2010) "Exploring The Pattern of Habits of Users Using Web Log Sequential Pattern", *IEEE International Conference onAdvances inComputing, Control and Telecommunication Technologies*, pp. 161-163.

[11]. Jian Pei, Jiawei Han, Behzad Mortazavi-asl. & Hua Zhu, (2000) "Mining Access Patterns Efficiently from Web Logs", *Pacific-Asia Conference on Knowledge Discovery and Data Mining Current Issues and New Applications*, pp. 396-407.

[12]. Tan Xiaoqiu, Yao Min. & Zhang Jianke, (2006) "Mining Maximal Frequent Access Sequences Based on Improved WAP-tree", *IEEE International Conference on Intelligent Systems Design and Applications*, pp. 616-620.

[13]. Sen Yang, Jiankui Guo. & Yangyong Zhu, (2007) "An Efficient Algorithm for Web Access Pattern Mining", *Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 726 – 729.

[14]. Lizhi Liu. & Jun Liu, (2010) "Mining Web Log Sequential Patterns with Layer Coded Breadth-First Linked WAP-Tree", *IEEE International Conference on Information Science and Management Engineering*, pp. 28-31.

[15]. V. Mohan, S. Vijayalakshmi . & S. Suresh Raja, (2009) "Mining Constraint-based Multidimensional Frequent Sequential Pattern in Web Logs", *European Journal of Scientific Research*, Vol. 36, No. 3, pp. 480-490.

[16]. Hai-yan Wu, Jing-jun Zhu. & Xin-yu Zhang, (2009) "The Explore of the Web-based Learning Environment based on

Web Sequential Pattern Mining", *IEEE International Conference on Computational Intelligence and Software Engineering*, pp. 1-6.

[17]. Bhupendra Verma, Karunesh Gupta, Shivani Panchal. & Rajesh Nigam, (2010) "Single Level Algorithm: An Improved Approach for Extracting User Navigational Patterns to Technology", *International Conference on Computer & Communication Technology*, pp. 436-441.

[18]. Olfa Nasraoui, Maha Soliman, Esin Saka, Antonio Badia. & Richard Germain, (2008) "A Web Usage Mining Framework for Mining Evolving User Profiles in Dynamic Web Sites", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 20, No. 2, pp. 202-215.

[19]. Arthur Pitman. & Markus Zanker, (2010) "Insights From Applying Sequential Pattern Mining To E- Commerce Click Stream Data", *IEEE International Conference on Data Mining Workshops*, pp.967-975.

[20]. Baoyao Zhou, Siu Cheung Hui. & Kuiyu Chang, (2004) "An Intelligent Recommender System using Sequential Web Access Patterns", *IEEE Conference on Cybernetics and Intelligent Systems*, pp. 393-398.

[21]. Show-Jane Yen, Yue-Shi Lee. & Min-Chi Hsieh, (2005) "An Efficient Incremental Algorithm for Mining Web Traversal Patterns", *IEEE International Conference on e-Business Engineering*, pp. 274-281.

[22]. Zhennan Zhang, Xu Qian. & Yu Zhao, (2008) "Galois Lattice for Web Sequential Patterns Mining", *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, pp. 102-106.

[23]. Dr. Suresh Jain, Ratnesh Kumar Jain. & Dr. R. S. Kasana, (2009) "Efficient Web Log Mining using Doubly Linked Tree", *International Journal of Computer Science and Information Security*, Vol. 3, No. 1, pp. 1-5.

[24]. Dhirendra Kumar Jha, Anil Rajput, Manmohan Singh. & Archana Tomar, (2010) "An Efficient Model for Information Gain of Sequential Pattern from Web Logs based on Dynamic Weight Constraint", *IEEE International Conference on Computer Information Systems and Industrial Management Applications*, pp. 518-523.

[25]. Xiaogang Wang, Yan Bai. & Yue Li, (2010) "An Information Retrieval Method Based on Sequential Access Patterns", *IEEE Asia-Pacific Conference on Wearable Computing Systems*, pp. 247-250.

[26]. Kanak Saxena. & Rahul Shukla, (2010) "Significant Interval and Frequent Pattern Discovery in Web Log Data", *IJCSI International Journal of Computer Science Issues*, Vol. 7, No. 3, pp. 29-36.

[27]. Diamanto Oikonomopoulou, Maria Rigou, Spiros Sirmakessis. & Athanasios Tsakalidis, (2004) "Full- Web Prediction based on Web Usage Mining and Site Topology", *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 716-719.

[28]. Rajimol A. & Raju G, (2011) "Mining Maximal Web Access Patterns- A New Approach", *International Journal of Machine Intelligence*, Vol. 3, No. 4, pp. 346-348.