International Journal of Technical Research and Applications e-ISSN: 2320-8163,
www.ijtra.com Volume 3, Issue 2 (Mar-Apr 2015), PP. 37-41

DETECTING USER'S BROWSING BEHAVIOR USING FREQUENT PAGES CONSTELLATION ALGORITHM

M. Sulthan Ibrahim, Dr. K. Thangadurai
P.G. and Research Department of Computer Science,
Government Arts College (Autonomous),
Karur-639005, Tamil Nadu, India.
km.sulthan@gmail.com, ktramprasad04@yahoo.com

ABSTRACT- The World Wide Web provides colossal volume of data to the internet users and grows at a rapid pace every day. The server creates log files regarding the page, IP address of the user, agent, operating system and time stamp and this data is mined to extract useful information using web usage mining. The primary objective of this paper is to find the browsing behavior pattern of the users from the log files using a novel algorithm named "Frequent Pages Constellation Algorithm". The FPC Algorithm groups the frequent pages browsed by the users and the results produced is utilized for personalization, forecast market vogues, enhances marketing strategies, improve website page positioning, site restricting etc. The experimental results showcased that the proposed FPC algorithm generates very less candidates, takes very less execution time and utilizes low memory.

Keywords— web mining, pattern extraction, usage mining, preprocessing

I. INTRODUCTION

Web mining is the application of data mining to the web data and traces user's visiting behaviors and extracts their interests using patterns. Since this area is applicable in e-commerce and Web analytics directly, web mining has become one of the important areas in computer science. Web Usage Mining uses mining methods in log data to extract the behavior of users which is used in various applications like personalized services, adaptive web sites, customer profiling, creating attractive web sites.

Similar to all data mining task, the process of Web usage mining also comprises of three major steps [26] (i) data pre-processing, (ii) pattern extraction and (iii) analysis. The input log data has to be pre-processed in order to have the appropriate input for the mining algorithms. The different methods need different input formats, thus the pre-processing phase can provide three types of output data. Pattern extraction means applying the introduction frequent pattern discovery methods to the log data related to the pages visited by the users.

The very purpose of web usage mining is to ascertain the useful data from web data or web log files [24]. The result of web usage mining can be utilized for target advertisement, enhancing web design, enhancing satisfaction of customer and personalize websites.

In Web Usage Mining, dataset can be collected from server logs, browser logs, proxy logs, or obtained from an enterprise's database. There are many types of data that can be utilized in Web Mining.

Web usage mining research focuses on extracting patterns of navigational behavior of users visiting a website [26]. These patterns of navigational behavior are precious to the firms as they provide answers to plethora of questions like, how effective and attractive are our website in delivering information to the users? How the users observe the structure of the website? Can we appropriately predict user's next visit to the website? Canwe design our site to cater to the need of the users? Can we increase the satisfactory level of the users? Can we target specific groups of users and personalize the web content to satisfy them? Almost all the answer to these questions may come from the analysis of the data from log files stored in web servers. Web usage mining has then become an imperative task in order to provide web administrators with meaningful information about users and usage patterns for improving quality of web information and service performance.

A. Web Content

This is the evident data in the Web pages or the information which was meant to be displayed to the users. A major part of this data will be text and images.

B. Web Structure

The Data which describes the organization of the website, it is divided into two types. Intra-page structure information includes the arrangement of various HTML or XML tags within a given page [20].

C. Web Usage

The Data that describes the usage patterns of Web pages, such as IP addresses, page references, and the date and time of accesses and various other information's depending on the log format file.

D. Web Server Logs

These are logs which maintain entire history of page requests given by the user at the server side. The W3C maintains a standard format for web server log files, but other proprietary formats exist. More recent entries are typically appended to the end of the file. Information about the request, including client IP address, request data/time, page requested, HTTP code, bytes served, user agent, and referrer are typically added [25]. These data can be combined into a single file, or separated into distinct logs, such as an access log, error log, or referrer log.

E. Proxy Server Logs

A Web proxy is a cache mechanism which interacts between client browsers and web server. It helps to decrease the load time of web pages and the network traffic load.

F. Browser Logs

The data which are collected at the browser client side after modifying the browsers or by using JavaScript and Java applets.

G. Typical web log data

The web log file consists of many fields like IP address or hostname, User Agent, Referring URL, Method, Protocol, Path, Agent, Date, and Time. The web log file is preprocessed in such a way that it is ready to be used in the

algorithm to fetch useful patterns. The usual web log file is shown in table 1 and this web log file is transformed into algorithmic log file as shown in table 2. The data is converted into records based on the IP address to identify the users browsing or navigational pattern. This preprocessed log file is fed as input to the proposed FPC algorithm to test its workability.

II. DATA PREPROCESSING

Usually the information present in a raw Web server log is not consistent and doesn't represent a user session file[1]. The data preprocessing is done to restore users' activities in the Web server log in a reliable and dependable manner. This preprocessingcomprises of four major tasks: i) eliminating unwanted entries, ii) Categorize users, iii) session identification, and iv) Restore session contents

A. Eliminate Unwanted Entries:

Web logs contain information related to user activity and the irrelevant entries can be removed without noticeably affecting the mining. The image log entries, Robot or crawler accesses can be removed from the entries since images downloaded will be recorded in the log along with the crawler accesses. These log entries are irrelevant and removed from the logs.

B. Categorize Users

A user is an individual who accesses web files from web servers through a browser at client place. A web log sequentially records users' activities according to the time stampoccurred and accessed. In order to study the actual user behavior, users in the log must be distinguished.

C. Session Identification

For logs that span long periods of time, it is very likely that individual users will visit the Web site more than once or their browsing may be interrupted. The goal of session identification is to divide the page accesses of each user into individual sessions. A time threshold is usually used to identify sessions.

D. Restore session contents

The primary task of this process determines if there are important accesses that are not recorded in the access logs. For example, Web caching or using the back button of a browser will cause information discontinuance in logs.

IP	Method	Protocol	Page	Agent	OS	Date	Time
192.163.56.1	GET	HTTP1.1	Page 1	Opera	Win7	10.12.14	22.10.13
192.163.56.1	GET	HTTP1.1	Page 3	Opera	Win7	10.12.14	22.11.29
192.163.56.1	GET	HTTP1.1	Page 4	Opera	Win7	10.12.14	22.17.43
192.163.56.1	GET	HTTP1.1	Page 6	Opera	Win7	10.12.14	22.20.27
65.92.48.17	GET	HTTP1.1	Page 1	IE	Win7	10.12.14	22.17.04
65.92.48.17	GET	HTTP1.1	Page 3	IE	Win7	10.12.14	22.19.13
65.92.48.17	GET	HTTP1.1	Page 7	IE	Win7	10.12.14	22.20.56
65.92.48.17	GET	HTTP1.1	Page 4	IE	Win7	10.12.14	22.31.38
65.92.48.17	GET	HTTP1.1	Page 17	IE	Win7	10.12.14	22.33.41

Table 1: Web Log File

IP	Pages Visited
192.163.56.1	1,3,4,6
65.92.48.17	1,3,7,4,17
129.09.12.77	1,2,4,7,15,17
190.76.02.19	1,3,4,6,15,17
155.33.42.11	1,3,4,17
171.90.12.25	1,4,3,15

Table 2: Preprocessed transaction log file

This preprocessed log file is used to find the user navigational and browsing behavior using the proposed FPC algorithm.

Existing Algorithm

Agarwal et al [21] developed the Apriori algorithm which is based on candidate generation approach.The Apriori algorithm get two inputs as follows: set of transactions (D) and minimum support (α). In each transaction, the items are sorted in their lexicographic order. It denoted as $X[i]$ means that the i^{th} item in X. $F_k$denoted the frequent k-sets means a set X is the k-set $\{X[1]\ldots\ldots.X[k]\}$. The Apriori used a breadth-first search technique to search by repeatedly generating and counting candidate sets. Basically, the Apriori algorithm is using monotonicity property. That is a set, is candidate if all of its subsets are counted and frequent.

Apriori Algorithm

Input: Datasets D,MinSupp α
Output: $F(D, \alpha)$
Step 1: $C_1 := \{\{i\} \mid i \in I\}$
Step 2: $k := 1$
Step 3: While $C_k \neq \{\}$ do
Step 4: For all transactions $(TID, I) \in D$ do
Step 5: For all candidate sets $X \in C_k$do
Step 6: If $X \subseteq I$then
Step 7: Increment $X.support$by 1
Step 8: End if
Step 9: End for
Step 10: End for
Step 11: $F_k := \{X \in C_k \mid X.support \geq \alpha\}$
Step 12: $C_{k+1} := \{\}$
Step 13: For all $X, Y \in F_k$ Such that $X[i] = Y[i]$
Step 14: For $1 \leq I \leq k$- 1, and $X[k] < Y[k]$ do
Step 15: $I := X \cup \{Y[k]\}$
Step 16: If $\forall J \subset I; |J| = k : J \in F_k$then
Step 17: Add I to C_{k+1}
Step 18:End if
Step 19: End for
Step 20: Increment k by 1
Step 21: End while

Demerits of Apriori

1. It generates huge number of candidate sets.
2. When the longest frequent itemsets is *k*, Apriori needs *k* passes of database scans. So it will have low efficiency.
3. The computation time is very intensive at generating the candidate item sets and computing the support values for application with very low support and vast amount of items.

Proposed Algorithm

The Frequent Pages Constellation Algorithm is designed to find the frequent pages browsed by the users and this algorithm overcomes the speed related issues and candidates generated issues present in the Apriori algorithm. The Apriori algorithm uses power set to create frequent item sets during candidate generation and this power set model doesn't suits the purpose of finding the users browsing behavior. Also the number of candidates generated will be huge if the user session in the server log contains large item sets. But the proposed algorithm FPC generates very less candidates without converting the items lexicographically and this algorithm employs a new approach in generating candidates.

PROCEDURE FPCAlgorithm (Dataset Ds, minsuppmS)
Input: Dataset Ds, minimum support mS **Output**: Frequent Pageset **STEP 1**: SCAN Ds **STEP 2:** For each Row ∈ Ds do begin **STEP 3:** Add GenerateSequentialCombination(Row R) in pageArray **STEP 4:** End **STEP 5:**GeneratePageSets(pageArray,mS) **STEP 5:** End Procedure

Figure 1: Psuedocode of FPC Algorithm

PROCEDURE GenerateSequentialCombination (Row R)
Input: Row { I_1,I_2,I_3…… I_n } **Output**: CandidatesC_{final} without duplicates **STEP 1**: Count = Total items in Row **STEP 2:**For Each Item I ∈ Row do begin **STEP 3**:tempArray= I ∪ I+1 **STEP 4:**END For **STEP 5:**C_{final}←tempArray **STEP 6:**tCount=Total items in tempArray **STEP 7:**Do until [tCount>1] begin **STEP 8:**For Each Item I∈tempArray do begin **STEP 9:**candArray={I}∪{ I+1} **STEP 10:**C_{final}←candArray **STEP 11:** END For **STEP 12:** Clear tempArray **STEP 13:**tempArray←candArray **STEP 14:** Clear candArray **STEP 15:** tCount:=tCount -1 **STEP 15:** END do **STEP 16:**Return C_{final} **STEP 17:** END PROCEDURE

Figure 2: Pseudo code of generateSequentialcombination

1	3	3	7
1	**3**		**7**

PROCEDURE GeneratePageSets(Candidates CList , MinSupmS)
Input: Candidate CList, minimum support mS **Output**: Frequent PagesetP_{final} **STEP 1**: Initialize Pageset P:= ϕ **STEP 2**: For Each Itemset I ∈CList do begin **STEP 3:** For Each Itemset I+1∈CList do begin **STEP 4:**IF [CList[I] ⊆CList[I+1] do **STEP 5:** P:= P ∪ { I } **STEP 6:**END **STEP 7:** END For **STEP 8:** Find Minsup for P **STEP 9:** IF [MinSup> = mS] do **STEP 10:** P_{final}← P **STEP 11:** END For **STEP 12:** ReturnP_{final} **STEP 13:** END PROCEDURE

Figure 3: Pseudo code for GeneratePagesets

Explanation of the algorithm

Let us assume that the sample dataset contains six transaction rows as shown below

1,3,4,6
1,3,7,4,17
1,2,4,7,15,17
1,3,4,6,15,17
1,3,4,17
1,4,3,15

Table 3: Sample preprocessed Log transaction file

First the Log transaction file is scanned and the total number of transactions is found out and each row in the transaction log file is fetched to generate the combinations as follows.

Let us consider the second transaction row

{ 1,3,7,4,17 }

STEP 1: combine the Items {1,3},{3,7},{7,4},{4,17} without using power set kind of approach.

STEP 2: Now take the first and second itemsets found from STEP 1 and check for equalities to merge the value as shown below

Result = {1,3,7},{3,7,4},{7,4,17}

7	4	4	17
7	**4**		**17**

3	7	7	4
3	**7**		**4**

STEP 3:

Similarly the previous result item sets found in STEP2 are combined to get the result

1	3	7	3	7	4
1	**3**		**7**		**4**

3	7	4	7	4	17
3	**7**		**4**		**17**

Resultant Items combined is

{1,3,7,4},{3,7,4,17}. This process continues recursively until the itemset becomes one.

Result : { 1,3,7,4,17 }

1	3	7	4	3	7	4	17
1	**3**		**7**		**4**		**17**

III. RESULTS AND DISCUSSION (EXPERIMENTAL SETUP)

The algorithm FPC is implemented using C#.Net and the system configuration used is dual core 2.6GHz Processor with 1GB RAM. A sample click stream dataset MSNBC from UCI repository is used. The original dataset contains 9,89,818 sequences and the shortest sequences are removed to keep only 31,790 sequences. The number of distinct items in the data set is 17 (an item is a web page category). The average number of itemsets per sequence is 13.33 and the average number of distinct item per sequence is 5.33. The frequent item set combination obtained for sample data displayed in table 3 is illustrated hereunder.

1,3	3,7,4,17	4,7,15,17	1,3,4,6,15	1,4,3,15
3,4	1,3,7,4,17	1,3	3,4,6,15,17	
4,6	1,2	3,4	1,3	
1,3,4	2,4	4,6	3,4	
3,4,6	4,7	6,15	4,17	
1,3	7,15	15,17	1,3,4	
3,7	15,17	1,3,4	3,4,17	
7,4	1,2,4	3,4,6	1,3,4,17	
4,17	2,4,7	4,6,15	1,4	
1,3,7	4,7,15	6,15,17	4,3	
3,7,4	7,15,17	1,3,4,6	3,15	
7,4,17	1,2,4,7	3,4,6,15	1,4,3	
1,3,7,4	2,4,7,15	4,6,15,17	4,3,15	

Table 4: Combination sample

The combination itemset along with the transaction row number is generated and the final step of creating frequent page set is performed. The minimum support is fixed at 0.3 and 0.4 and the pagesets for these two minsup are generated.

Frequent page set with minsup> 0.4	**Interpreted Result**
[1,3] : [0,1,3,4]	Page1 → page3 Support = 0.6
[3,4] : [0,3,4]	page3 → Page4 Support = 0.5
[1,3,4] : [0,3,4]	Page1 → Page3 → Page4 Support 0.5

Table: 5 Rules Generation for minimum support is fixed at 0.3

The result is interpreted in such a way to help understand the browsing and navigation behavior of the users. Usually the users after hitting the page1 tends to move to the page 3 and the users who hits the page three moves to page

Frequent page set with minsup>= 0.3	**Interpreted Result**
[1,3] : [0,1,3,4]	Page1 → page3 Support = 0.6
[3,4] : [0,3,4]	page3 → Page4 Support = 0.5
[1,3,4] : [0,3,4]	Page1 → Page3 → Page4 Support 0.5
[4,6] : [0, 3]	Page 4 → Page 6 Support 0.33
[4,17] : [1,4]	Page 4 → Page 17 Support 0.33

Table: 6 Rules Generation for minimum support is fixed at 0.4

From the above interpreted result an interesting finding is unearthed. The users from page 4 navigate to page 6 and page 17 equally but the users from Page 3 are more likely to navigate through page 4 and through page 6.

IV. FUTURE WORK

In Future research the user navigational behavior can be found using algorithms like FP-Growth [16] for generating association between web pages and then applying some rule generating algorithm like RuleGen algorithm [18] and PrefixSpan [19].These algorithms can be applied for determining the frequent users' navigation and browsing style of web site and a novel system can be developed to predict the next probable page the user is bound to land. Also the existing hierarchical agglomerative clustering can be employed to cluster users' browsing behaviors and identify the navigational and browsing behavior of the users.

V. CONCLUSION

Foreseeing the user's browsing and navigational behavior or pattern is a remarkable technique which provides an in depth knowledge about the global trends about the web page positioning, provides many enterprises to promote their service and product easily visible on web sites. Frequent access behavior for the users can be used to identify needed links to improve the overall performance of future accesses. Also the Prefetching and caching policies can be made on the basis of frequently accessed pages to improve latency time. Overall Common access behaviors of the users can be found and used to improve the design of web pages and for making other modifications to a Web site. Usage patterns can be used for business intelligence in order to improve sales in enterprises. The execution time taken for the proposed frequent pages Constellation algorithm is very less when compared to the classic Apriori algorithm.

REFERENCES

[1] Chen Hu, XuliZong, Chung-wei Lee and Jyh-haw Yeh, "World Wide Web Usage Mining Systems and Technologies", Journal of SYSTEMICS, CYBERNETICS AND INFORMATICS Vol. 1, No. 4, Pages53-59, 2003.

[2] FlorentMasseglia, Pascal Poncelet, RosineCicchetti, "An efficient algorithm for Web usage mining", Networking and Information Systems Journal. Volume X, 2000

[3] R. Pamnani, P. Chawan "Web Usage Mining: A Research Area in Web Mining"

[4] Qiankun Zhao, Sourav S. Bhowmick, "Sequential Pattern Mining: A Survey", Technical Report, CAIS, Nanyang Technological University, Singapore, No. 2003118, 2003.

[5] S. Rawat, L. Rajamani, "Discovering Potential User Browsing Behaviors Using Custom-Built APRIORI Algorithm", International journal of computer science & information Technology (IJCSIT) Vol.2, No.4, August 2010

[6] Ming-Syan Chen, Jong Soo Park, Philip S. Yu, "Efficient Data Mining for Path Traversal Patterns", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 10, NO. 2, MARCH/APRIL 1998.

[7] Jianhan Zhu, Jun Hong, John G. Hughes, "Using Markov Chains for Link Prediction in Adaptive Web Sites", Soft-Ware 2002, LNCS 2311, pp. 60–73, 2002

[8] WANG Tong, HE Pi-lian, "Web Log Mining by an Improved AprioriAll Algorithm", World Academy of Science, Engineering and Technology 4 2005

[9] Hengshan Wang, Cheng Yang, HuaZeng, " Design and Implementation of a Web Usage Mining Model Based On

Fpgrowth and Prefixspan", Communications of the IIMA 2006 Volume 6 Issue 2

[10] Paola Britos, DamiánMartinelli, HernánMerlino, Ramón GarcíaMartínez, "Web Usage Mining Using Self Organized Maps", International Journal of Computer Science and Network Security, VOL.7 No.6, June 2007.

[11] MehrdadJalali, Norwati Mustapha, Ali Mamat, Md. Nasir B Sulaiman, "WEB USER NAVIGATION PATTERN MINING APPROACH BASED ON GRAPH PARTITIONING ALGORITHM", Journal of Theoretical and Applied Information Technology

[12] Kobra Etminani, Mohammad- R. Akbarzadeh- T., Noorali Raeeji Yanehsari, "Web Usage Mining: users' navigational patterns extraction from web logs using Ant-based Clustering Method", IFSA-EUSFLAT 2009

[13] Sandeep Singh Rawat, Lakshmi Rajamani, "DISCOVERING POTENTIAL USER BROWSING BEHAVIORS USING CUSTOM-BUILT APRIORI ALGORITHM", International journal of computer science & information Technology (IJCSIT) Vol.2, No.4, August 2010

[14] Mahdi Khosravi, Mohammad J. Tarokh, "Dynamic Mining of Users Interest Navigation Patterns Using Naive Bayesian Method", 978-1-42448230-6/10/$26.00 ©2010 IEEE

[15] N. Sujatha, K. Iyakutty, "Refinement of Web usage Data Clustering from K-means with Genetic Algorithm", European Journal of Scientific Research ISSN 1450-216X Vol.42 No.3 (2010), pp.464-476

[16] JIAWEI HAN, JIAN PEI, YIWEN YIN, RUNYING MAO, Mining Frequent Patterns without Candidate Generation: A Frequent Pattern Tree Approach, Data Mining and Knowledge Discovery, Volume 8, Issue 1, 53–87, 2004.

[17] RakeshAgrawal, Ramakrishnan Srikant, Fast Algorithms for Mining Association Rules, IBM Almaden Research Center 650 Harry Road,San Jose, CA 95120,1994

[18] M. J. Zaki, SPADE: An Efficient Algorithm for Mining Frequent Sequences, Machine Learning, vol. 42, no.12,2001, pp. 3160.

[19] Jian Pei, Jiawei Ha şialţii, Mining Sequential Patterns by Pattern Growth: The Prefix Span Approach, IEEE TRANSACTIONS ONKNOWLEDGE AND DATA ENGINEERING, VOL. 16, NO. 10, OCTOBER 2004.

[20] Mazilescu V.: Fuzzy Dynamic Discrimation Algorithms for Distributed Knowledge Management Systems, Annals of Dunarea de Jos University of Galati. Fascicile I. Economics and Applied Informatics, 2010, Years XVI, no. 2, p. 1526.

[21] Agrawal R., Srikant R.: Mining sequential patterns. In Proceedings of the 11th International Conference on Data Engineering, 314, September 1995. Aideep Srivastave, Robert Cooley, Mukund Deshpande, Pang-Ning Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," SIGKDD Explorations. ACM SIGKDD, 2000.

[22] Liu B.: Web Data Mining: Exploring Hyperlinks, Contents and Usage Data, Springer Berlin Heidelberg New York, 2006.

[23] M.Sulthan Ibrahim, Dr.K.Thangadurai, "An Overview of Web Mining Research**"**, published in *Journal of Computer Science and Application (JCSA)*, Volume: 6, Number: 1, 2014, pp. 357-360, ISSN: 2231-1270.

[24] M.Sulthan Ibrahim, Dr.K.Thangadurai, "Analysis of web mining for web Personalization**"**, published in International Journal Of Advanced Research In Datamining And Cloud Computing *ISSN 2321-8754 online ISSN 2321-8924 print volume 3, issue 1, january 2015 pp 8-14, impact factor: 0.603***.**

[25] http://httpd.apache.org/docs/1.3/logs.html

[26] http://www.w3.org/TR/WD-logfile.html

[27] http://www.internetworldstats.com

[28] http://www.domaintools.com/internet-statistics/