Ravindra Gupta,Prateek Gupta / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 1,Jan-Feb 2012, pp.596-598

Application Oriented Web Usage Mining with Customized Web Log Preprocessing & Frequent Pattern Tree

Ravindra Gupta*, Prateek Gupta**

ABSTRACT

Web Usage Mining discovers interesting patterns in accesses to various Web pages within the Web space associated with a particular server. The Web Usage Mining architecture divides the process into two main parts- the first part includes preprocessing, transaction identification, and data integration components. The second part includes the largely domain independent application of generic data mining and pattern matching. This paper contains an efficient improved iterative FP Tree algorithm for generating frequent access patterns from the access paths of the users. The frequent access patterns are generated by backward tree traversals. This operation will take less time compare to the existing algorithms. Paper also composed of customized web log preprocessing for mined in different applications.

Keywords - Web Usage Mining, FP Tree, Web Logs, Web Log Preprocessing, Customized Web Log Preprocessing

I. INTRODUCTION

The Web Mining [1] is the application of Data Mining techniques to automatically discover and extract information from the web. Web usage mining is the task of discovering the activities of the users while they are browsing and navigating through the Web. The aim of understanding the navigation preferences of the visitors is to enhance the quality of electronic commerce services (e-commerce), to personalize the Web portals [2] or to improve the Web structure and Web server performance Web usage mining [3], from the data mining aspect, is applying data mining techniques to discover usage patterns from Web data. Examples of applications of such knowledge include improving designs of web sites, analyzing system performance as well as network communications, understanding user reaction and motivation, and building adaptive Web sites.

The process of Web usage mining also consists of three main steps: (i) preprocessing, (ii) pattern discovery and (iii) pattern analysis.

In this work pattern discovery means applying the introduced frequent pattern discovery methods to the log data. For this reason the data have to be converted in the preprocessing phase such that the output of the conversion can be used as the input of the algorithms. Log files are stored on the server side, on the client side and on the proxy servers. Logs are processed in Common Log Format. Pattern analysis means understanding the results obtained by the algorithms and drawing conclusions. In pattern discovery phase methods and algorithms used have been developed from several fields such as statistics, machine learning, and databases. This phase of Web usage mining has three main operations of interest: association (i.e. which pages tend to be accessed together), clustering (i.e. finding groups of users, transactions, pages, etc.), and sequential analysis (the order in which web pages tend to be accessed). Pattern analysis is the last phase in the overall process of Web usage mining. In this phase the motivation is to filter out uninteresting rules or patterns found in the previous phase.

If all the transactions are different i.e. no common access paths then Apriori algorithm is good otherwise FP-tree Algorithm is good [4]. Apriori Algorithm will take more time and more memory compare to the FP-tree algorithm. The partition approach is good if data base size is very large; if database size is less then it will work as Apriori algorithm [6]. In this paper we have used two main tasks-Customized web log preprocessing and Improved FP Tree Algorithm

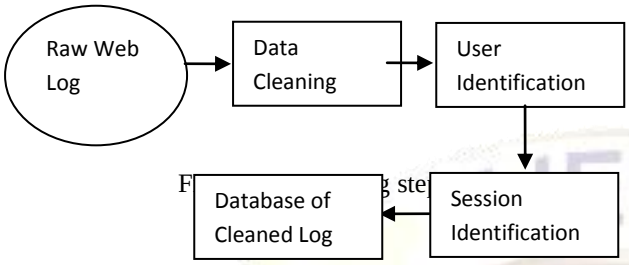
II. EXISTING WORK

A. Web Log Preprocessing

The inputs to the preprocessing phase are the log and site files. The outputs are the user session file, transaction file Web servers register a Web log entry for every single access they get, in which important pieces of information about accessing are recorded, including the URL requested, the IP address from which the request originated, and a timestamp. A log file can be located in three different places-Web Servers, Web proxy Servers, and Client browsers [8]. The most popular log file format is the Common Log Format (CLF)-

<ip><baseurl><date><method><file><protocol><code> <bytes> <referrer><user agent>

Preprocessing [9] contains four sub steps: Data Cleaning, User Identification, Session Identification and Formatting.



B. FP Tree Algorithm

The FP-tree algorithm avoids candidate generation steps [7]. The main idea of the algorithm is to maintain a frequent pattern tree (FP-Tree) of the database [5]. It is an extended prefix-tree structure, storing crucial quantitative information about frequent sets. The tree nodes are frequent items and are arranged in such a way that more frequently occurring nodes will have a better chances of sharing nodes than the less frequently occurring ones. The method starts from frequent 1-itemsets as an initial suffix pattern and examines only its conditional pattern base (a subset of the database), which consists of the set of frequent items co-occurring with the suffix pattern. The algorithm constructs the conditional FP-tree and performs mining on this tree. A hash-based technique is used to reduce the size of the candidate *k*- patterns. Another variation is to reduce the number of transactions to be scanned at higher values of *k*. Since a transaction that does not contain any frequent *k*-pattern cannot contain any frequent (*k*+1) - pattern, these types of transactions can be marked during the *K*th scanning and are not considered in the subsequent scanning.

III. PROPOSED WORK

A. Customized Web Log Preprocessing

Different web application requires different preprocessing of logs. Multimedia application requires log of multimedia link request like log having jpg, mpg, and gif etc. resource. All application removes error log. E-commerce application requires different user requests. We introduced one more step in traditional preprocessing steps, before data cleaning, Customization. In this step we clean log on the basis of user requirement for application.

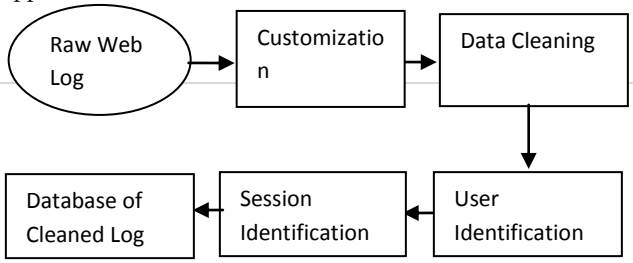


Fig-2 customized preprocessing steps

Steps for customized web log preprocessing are as follows –

Step 1. input raw web accessing log file

Step 2. take user choice for normal, multimedia, graphics or e-commerce applications.

Step 3. read raw web log file and remove logs according to user selection to make intermediate file.

Step 4. identify users & resources uniquely & assign a unique id to them.

Step 5. identify resource accessed by users according to id.

Step 6. create preprocessed file by mapping of user id and resource ids accessed by them.

Step 7. show comparison of file generated after customized preprocessing & without in terms of size.

B. Improved FP Tree Algorithm

There are many existing algorithms for generating frequent access patterns from the access paths. But they have less efficient in terms of execution time and memory requirement. The proposed algorithm is modification of FP-tree Algorithm. In proposed FP tree structure items are stored in descending order of their frequency. This structure contains- Item Table having index, item-id, frequency & pointer to root. Each tree node has node-id, pointer to next siblings, number of occurrences.

Mining with improved fp tree structure is divided into two main processes : creation of modified fp tree & mining.

Steps for construction of improved fp tree are as follows :

Step 1. identify frequent items from sample database & make frequent item table.

Step 2. sort frequent item table in descending order of frequency.

Step 3. assign index for each item in frequent item table.

Step 4. make mapping according to transactions.

Step 5. sort mapping in ascending order of item ids.

Step 6. initialize item with first item of mapping.

Step 7. make current node as root of sub tree referred by item.

Step 8. for each subsequent items of item I from mapping table perform counting for existing or non existing nodes & go to step 4.

Steps for finding frequent patterns from modified fp tree are as follows-

Step 1. create modified fp tree.

Step 2. Initialize sub fp tree with i as root where i is index of least frequent item in frequent item table.

Step 3. construct sub item table.
Step 4. attach child with each frequent item in sub item table.
Step 5. construct sub fp tree from modified fp tree.
Step 6. for each child i:
(a) set all counts to 0.
(b) for each occurrence of node in sub fp tree increment count of each item in the path of root.
(c) for each frequent item attach child of item. & repeat step 6.
Step 7. print frequent item set.
Step 8. go to step 2.

IV. IMPLEMENTATION

I have implemented these in C++ and Java. I had use NetBeans IDE 6.7 . Input to my program is raw web log file. Then customized preprocessing step generates compressed log file having access behaviour in numeric form, which is gone for mining by modified FP tree algorithm. Output is then mapped in the form of log resource. Screen shot of interface is shown below :

Fig-3 implementation

V. RESULT

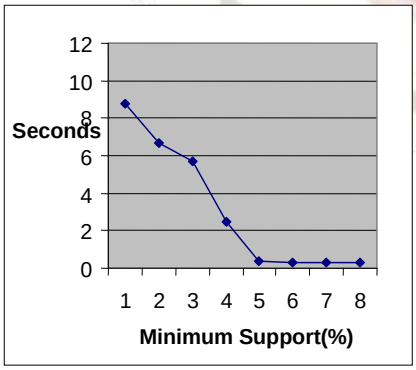


Fig-4 proposed algorithm result

The above figure shows the result of proposed algorithms in term of execution time. An evident from the figure, the proposed Algorithm is not generating any candidate sets, but more number of patterns will be generated, due to this the number of tree traversals will be more. From the above Figure 4 the proposed algorithm is taking less time.

VI. CONCLUSION

Research work includes preprocessing phase and pattern discovery phase of web usage mining which can be utilized in industry and application oriented system.
We uses customized web log preprocessing rather than traditional approach which may reduces size of raw web log file. Improved fp tree algorithm is best in the respect of execution time and memory complexity.
In future research, work is carried out for developing a web usage mining tool with customized web log preprocessing and combined pattern analysis approaches according to different application.

REFERENCES

[1] R. Cooly, B. Mobasher and J. Srivastava, "Web Mining : In- formation and Pattern Discovery on the World Wide Web" , IEEE, August 1997. Pp.558 – 566.
[2] Charu C. Aggarwal and Philip S. Yu, "An Automated Sys - tem for Web Portal Personalization", Proceedings of the 28th VLDB Conference, Hong Kong, China,2002
[3] Renáta Iváncsy, István Vajk, "Frequent Pattern Mining in Web Log Data", Acta Polytechnica Hungarica, Vol.3, No. 1, 2006
[4] WangBin and LiuZhijing. Web Mining Research, Fifith International Conference on Computational Intelligence and Multimedia Applications, Jan 2003.
[5] B.Santhosh Kumar, K.V.Rukmani, "Implementation of Web Usage Mining Using Apriori and FP Growth Algorithm", Int. J. of Advanced Networking and Applications,Volume:01, Issue:06, (2010), Pages: 400-404
[6] Park Jong Soo, Chen Ming-Syan, and Yu Philip S. Using a hash-based method with transaction trimming for mining association rules. IEEE transactions on knowledge and data Engineering, 9 ,no. 5,Sept/oct 1997.
[7] Shui Wang, Le Wang, "An implementation of FP growth algorithm based on high level data structure of Weka-JUNG framework", Journal of Convergence Information Technology, Volume 5, Number 9, 2010
[8]Kotsiantis S, Kanellopoulos D., "Association Rules Mining: A Recent Overview", GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp.71- 82
[9] J. Han, J. Pei, and Y. Yin. "Mining frequent patterns with – out candidate generation". IEEE, Sept.1998 pp-365-378.
[10] K. R. Suneetha and Dr. R. Krishnamoorthi, "Identifying User Behavior by Analyzing Web Server Access Log File",IJCSNS International Journal of Computer Science and Network Se-curity, VOL.9 No.4, April 2009, pp. 327-3