Web Log Mining: An Enhancement to Server Performance and Sight Navigation

Heena Goyal
CSE Department,
ITM University, Gurgaon
heenaki.goyal@gmail.com

Nidhi
IT Department,
ITM University, Gurgaon
nidhisharma035@gmail.com

ShilpaYadav
CSE Department,
ITM University, Gurgaon
shippu.rao@gmail.com

N.N.Das
IT Department,
ITM University, Gurgaon
nndas@itmindia.edu

Abstract: **The aim of discovering frequent patterns in Web log data is to obtain information about the navigational behavior of the users. Web Server register a log entry for every single access they get. A huge number of accesses are registered and collected in an ever growing web log. In this paper, we study how an efficient web log mining contributes to enhance server performance, improve web site navigation and system design of web applications.**

Key Words: **User session, Pattern Discovery and Data Location.**

I. INTRODUCTION

The expansion of the World Wide Web has resulted in a large amount of data that is now freely available for user access. The different types of data have to be managed and organized in such a way that they can be accessed by different users efficiently. Therefore, the application of data mining techniques on the Web is now the focus of an increasing number of researchers. Several data mining methods are used to discover the hidden information in the Web. However, Web mining does not only mean applying data mining techniques to the data stored in the Web. The algorithms have to be modified such that they better suit the demands of the Web.

New approaches should be used which better fit the properties of Web data [6]. Furthermore, not only data mining algorithms, but also artificial intelligence, information retrieval and natural language processing techniques can be used efficiently. Thus, Web mining has been developed into an autonomous research area. Many websites have a hierarchical organization of content. This organization may be quite different from the organization expected by visitors to the website. In particular, it is often unclear where a specific document is located. For optimization of benefits for site navigation and server performance improvement, there should be clear separation between expected and target pages.

II. USER SESSION

It is the session of activity that a user with a unique IP address spends on a Web site during a specified period of time. The number of user sessions on a site is used in measuring the amount of traffic a Web site gets. The site administrator determines what the time frame of a user session will be (e.g., 15 minutes). If the visitor comes back to the site within that time period, it is still considered one user session because any number of visits within those 30 minutes will only count as one session. If the visitor returns to the site after the allotted time period has expired, say an hour from the initial visit, then it is counted as a separate user session. The extraction of the data from Web logs gives access to information that have to be managed efficiently in order to be able to exploit them for analyses [2,4].

A request represents the data of the HTTP request that are recorded in the Web log files. A session is a particular set of requests made in a certain interval of time by the same client. Sessions are found, when information about sessions are not available, as in our case, through empirical rules, the heuristics. Organizing the HTTP requests in a single session permits to have a better view of the actions performed by visitors. A procedure, named "session reconstruction", may be used in order to map the list of activities performed by every single user to the visitors of the site. A heuristic has been used that identifies a single user with the pair IP address and user-agent, and permits only a fixed gap of time between two consecutive requests.
Here, a new request is put in an existing session if following conditions are true [2]:

- The IP address and the user-agent are the same of the requests already inserted in the session.
- The request is inserted within the fifteen minutes after the last request inserted.

The reason for the choice of the couple of IP address and user-agent as identifiers is to distinguish different users coming from the same proxy.
Different people using the same proxy result in requests done by the same IP, despite of the real identity of the clients [5]. The introduction of the user-agent permits to differentiate more clearly the source of requests. For example:-

TABLE I. NUMBER OF HTTP REQUESTS FOR EACH METHOD

HTTP method	Total Number
CONNECT	4
LINK	10
PROPFIND	240
PUT	2,500
OPTIONS	3,500
HEAD	33,770
POST	84,000
GET	20,3450
TOTAL	327474

Method Specifications:

- CONNECT: This specification reserves the method name CONNECT for use with a proxy that can dynamically switch to being a tunnel.
- LINK: The LINK method of HTTP adds meta information (object header information) to an object, without touching the object content.
- PUT: The PUT method requests that the enclosed entity be stored under the supplied Request-URI. If the Request-URI refers to an already existing resource, the enclosed entity SHOULD be considered as a modified version of the one residing on the origin server.
- OPTIONS: This method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.
  This method allows the client to determine the options and/or requirements associated with a resource, or the capabilities of a server, without implying a resource action or initiating resource retrieval.
- HEAD: This method is identical to GET except that the servers MUST NOT return a message-body in the response.
- POST: The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.
- GET: The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI. If the Request-URI refers to a data-producing process, it is the produced data which shall be returned as the entity in the response and not the source text of the process, unless that text happens to be the output of the process.

III. PATTERN DISCOVERY

A traversal pattern is a list of pages visited by a user in one session**.** Several different traversal patterns and the corresponding methods of discovering them have been presented here namely, Association Rules, Sequential Patterns and Frequent Episodes [9].

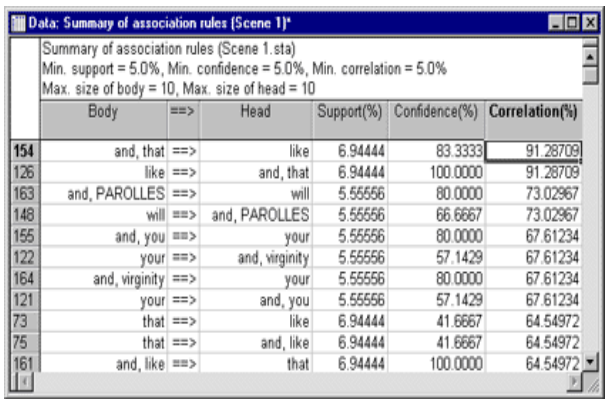
A. Association Rules

Association rules describe the associations among items bought by customers in the same transaction, e.g., 75% of customers who bought music systems would also buy CDs in some store. The goal of the techniques described in this topic is to detect relationships or associations between specific values of categorical variables in large data sets.
Suppose we are collecting data at the check-out cash registers at a large book store. Each customer transaction is logged in a database, and consists of the titles of the books purchased by the respective customer, perhaps additional magazine titles and other gift items that were purchased, and so on. Hence, each record in the database will represent one customer (transaction), and books purchased by that customer. The purpose of the analysis is to find associations between the items that were purchased, i.e., to derive association rules that identify the items and co-occurrences of different items that appear with the greatest frequencies.

For example, we want to learn which books are likely to be purchased by a customer who we know already purchased (or is about to purchase) a particular book. This type of information could then quickly be used to suggest to the customer those additional titles[9].

- Support: Relative frequency of the Body or Head of the rule.
- Confidence: conditional probability of the Head given the Body of the rule.
- Correlation: support for Body and Head, divided by the square root of the product of the support for the body and the support for the Head.

Fig1.TABLE FORM OF ASSOCIATION RULE EXAMPLE



Data: Summary of association rules (Scene 1)*

Summary of association rules (Scene 1.sta)
Min. support = 5.0%, Min. confidence = 5.0%, Min. correlation = 5.0%
Max. size of body = 10, Max. size of head = 10

	Body	==>	Head	Support(%)	Confidence(%)	Correlation(%)
154	and, that	==>	like	6.94444	83.3333	91.28709
126	like	==>	and, that	6.94444	100.0000	91.28709
163	and, PAROLLES	==>	will	5.55556	80.0000	73.02967
148	will	==>	and, PAROLLES	5.55556	66.6667	73.02967
155	and, you	==>	your	5.55556	80.0000	67.61234
122	your	==>	and, virginity	5.55556	57.1429	67.61234
164	and, virginity	==>	your	5.55556	80.0000	67.61234
121	your	==>	and, you	5.55556	57.1429	67.61234
73	that	==>	like	6.94444	41.6667	64.54972
75	that	==>	and, like	6.94444	41.6667	64.54972
161	and, like	==>	that	6.94444	100.0000	64.54972

B. *Sequential patterns*

Sequence analysis is concerned with a subsequent purchase of a product or products given a previous buy. For instance, buying an extended warranty is more likely to follow (in that specific sequential order) the purchase of a TV or other electric appliances. Sequence rules, however, are not always that obvious, and sequence analysis helps you to extract such rules no matter how hidden they may be in your market basket data. There is a wide range of applications for sequence analysis in many areas of industry including customer shopping patterns, phone call patterns, the fluctuation of the stock market, DNA sequence, and Web log streams. Sequential patterns have also been applied to Web logs .The sessions are ordered by the user id and the access time. As for association rules, the duplicate pages are discarded. Then for each user, there is a user sequence, which consists of all sessions of the user [9]. A sequential pattern is a maximal sequence of item sets whose support is not less than some predefined threshold. A sequence is maximal if it is not contained in any other sequence. The support of a sequence is the percentage of user sequences that contain the sequence.

C. Frequent Episodes

Discovery of frequent episodes is a data mining method for temporal data. Temporal data is a type of the data which refer to the time. There are several different tasks of the temporal mining as temporal clustering and other. The main idea of discovery of the frequent episodes is to discover a frequently occurs sequence of events. An episode is a sequent of events which have some dependence between each other. This dependence appears in the order of occurring of events. Events in episodes occur close to each other. Proximity is shown in an order of the occurring [9].

IV. DATA LOCATION IDENTIFICATION

Web sites have lots of contents organized in a particular manner useful for information seekers but the pattern in which they get the data is not predefined means there can be difference between their expected location of getting the data from and the actual location where it exists. And without a clear separation of specification of content and navigation, it is hard to differentiate between backtrackers, one who browse for a set of target pages, and others who backtrack because they are searching for a single target page. In this paper, we proposed an efficient method to easily identify different backtrackers using time thresholds, user sessions and priorities of pages accessed by users who are navigating through the site so that the performance of Server can be improved by maintaining the log information [1].

There are two kinds of locations from where data can be obtained.

- Expected Location: Where the user expects the information to be found.
- Target Location: Where the user finally gets the information. He can search for a single target page or set of target pages.

There are many websites like Amazon and EBay, having a clear separation between content pages and index (or navigation) pages; product pages on these websites are content pages, and category pages are index or navigation pages. In such cases, we can consider the target pages for a visitor to be exactly the set of content pages requested by the visitor. Other websites such as information portals or corporate websites may not have a clear separation between content and index pages. For example, Yahoo! lists websites on the internal nodes of its hierarchy, not just on the leaf nodes. In this case, we can use a time threshold to distinguish whether or not a page is a target page. Pages where the visitor spent more time than the threshold are considered target pages.

A. Visitor's Search Pattern Scenario

Single Target Case: Consider the case where the visitor is looking for a single specific target page *T*. Here, it is expected from the visitor to execute the following search strategy[1,6]:

1. Start from the root.
2. While (current location *CL* is not the target page *TP*) do
   (a) If any of the links from *CL* seem likely to lead to *TP*, follow the link that appears most likely to lead to *TP*. It is verified from the set of keywords entered by the user.
   (b) Else, either backtrack and go to the parent of *CL* and follow a different link since the estimates of the link is updated.

Multiple Target case[1]: Now consider the scenario where the visitor wants to find a set of target pages T_1, T_2... T_n. The search pattern is similar, except that after finding (or giving up on) T_i, the visitor then starts looking for T_{i+1}:

1. for i: = 1 to n
(a) If i = 1, start from the root, else from the current location *CL*.
(b) While (current location *CL* is not the target page T_i) do

- If any of the links from *CL* seem likely to lead to T_i, follow the link that appears most likely to lead to T_i.
- Else, either backtrack and go to the parent of *CL* with some probability, or give up on T_i and start looking for T_{i+1} at step 1(a) with some probability.

V. CONCLUSION AND FUTURE WORK

Mining data over the web using different tools and technologies of data mining is termed as web mining. This area has become one of the most famous among the researchers within last few years. Without any doubt the World Wide Web is the best repository that can be mined for data. In this paper, we presented different terms and methods to show how a server performance and navigational capabilities can be improved by updating user sessions and keeping record of that. In future, many more applications and algorithms can be derived for optimizing the performance.

ACKNOWLEDGMENT

We would like to thank Mrs. Kavita Chaudhary, Mrs. Niharika Garg and team members for their support and help in presenting this paper.

REFERENCES

[1] Yinghui Yang* Dept. of Operations & Information Management Wharton Business School University of Pennsylvania 3620 Locust Walk, Suite 1300 Philadelphia, PA 19104

[2] Maristella Agosti and Giorgio Maria Di Nunzio Department of Information Engineering – University of Padua via Gradegnigo 6/a, 35131 Padova, Italy

[3] D. Nicholas, P. Huntington, A. Watkinson "Scholarly journal usage: the results of deep log analysis", Journal of Documentation Vol.61 No. 2, 2005.

[4] P.M. Hallam-Baker, B. Behlendorf "Extended Log File Format, W3C Working Draft WD-logfile-960323" http://www.w3.org/TR/WD-logfile.html

[5] M. Agosti, G.M. Di Nunzio and A. Niero "From Web Log Analysis to Web User Profiling" In DELOS Conference 2007. Working Notes. Pisa, Italy, 2007, pp 121–132.

[6] Qingtian Han, Xiaoyan Gao, Wenguo Wu, "Study on Web Mining Algorithm Based on Usage Mining", Computer-Aided Industrial Design and Conceptual Design, 2008. CAID/CD 2008. 9th International Conference on 22-25 Nov. 2008

[7] Qingtian Han, Xiaoyan Gao, "Research of Distributed Algorithm Based on Usage Mining", Knowledge Discovery and Data Mining, 2009, WKDD 2009, Second International Workshop on 23-25 Jan. 2009

[8] Yan Li, Boqin Feng, Qinjiao Mao, "Research on Path Completion Technique in Web Usage Mining", Computer Science and Computational Technology, 2008. ISCSCT '08. International Symposium on Volume 1, 20-22 Dec. 2008

[9] http://what-when-how.com/information-science-and-technology/traversal-pattern-mining-in-web-usage-data/