PrefixSpan Algorithm: An Approach for Mining User's Traversal Patterns from Server Log Files

Prof. Komal N. Porwal[#1], Prof. S. B. Patil*[2], Prof. Pallavi S. Kadam[#3]
#Computer Engineering Department, MBT Campus.
Islampur, Maharashtra, India.
[1]komal_porwal@yahoo.com
**Electronics & Telecommunication Engg. Department, MBT Campus.*
Islampur, Maharashtra, India.
[2]patisbp@gmail.com
#Computer Engineering Department, MBT Campus,
Islampur, Maharashtra, India.
[3]pallavikadam14@gmail.com

Abstract- **In today's world of technological advancement, the use of World Wide Web (WWW) as the means for marketing and selling has been in humongous increase. Nonetheless to say, every major entrepreneur has its own website to make aware to the people all around the globe, about their business. The level of E-commerce has reached a benchmark that organizations have to fulfill one's demands of a right level of information to be available online. Now, the question arises as to how one could tell what contents are being read, if a website is effective, or even the extent of readers going through the content. Web Usage Mining (WUM), better known as Web Log Mining, is an application of data mining algorithms to Web access logs to fetch trends and regularities in Web users' traversal patterns.**

This paper introduces PrefixSpan (i.e., Prefix-projected Sequential pattern mining), efficient algorithm to discover behavioral patterns of users interacting with a Website. Our performance study shows that PrefixSpan outperforms both the Apriori-based GSP algorithm and another recently proposed method, FreeSpan, in mining large sequence databases.

Keywords**: WWW, Log Files, Data Mining, Web Usage Mining, PrefixSpan.**

I. INTRODUCTION

The World Wide Web is a humongous source of electronic data that get collected from Web content, collective of billions of publicly available pages, or from web usage, a huge cloud collection of log information collected from servers all around the globe. Web Usage Mining (WUM), also known as Web Log Mining, is an application of data mining algorithms to Web access logs to find trends and regularities in Web users' traversal patterns [1].

Web Mining is that area of Data Mining which deals with the extraction of interesting knowledge from the World Wide Web [2]. Web Usage Mining is that part of Web Mining which deals with the extraction of knowledge from server log files; source data mainly consist of the (textual) logs that are collected when users access web servers [3].

And the results of WUM have been used in improving the design of Website, business and marketing decision support, user profiling, and performance of Web server system.

What are Log Files?

A Web server log is an important source for performing Web Usage Mining because it explicitly records the browsing behavior of site visitors. In conjunction to HTTP request, every hit against the server, the server access log is populated with a single entry generation. These logs can be stored in various formats such as Common log or Extended log formats.

Each log entry (depending on the log format) may contain following fields:

TABLE 1
Web Log Field Descriptions

Sr. No.	Field	Appears as	Description
1.	Date	date	Date on which request was made
2.	Time	time	The time, in coordinated universal time (UTC), at which request was made
3.	Service Name and Instance Number	s-sitename	The Internet service name and instance number that was running on the client
4.	Server IP Address	s-ip	The IP address of the server on which the log files entry was generated.
5.	Method	cs-method	Method of request (Get, Post, etc)
6.	URI Stem	cs-uri-stem	The target of the action, for example, Default.htm.
7.	URI Query	cs-uri-query	Query from Client
8.	Client IP Address	c-ip	IP address of Client that made the request
9.	User Agent	cs(User-Agent)	OS and browser software at the Client
10.	Referrer	cs(Referrer)	URI from where request originated

11.	HTTP Status	sc-status	The HTTP status code
12.	Protocol Substatus	sc-substatus	The sub status error code
13.	Win32 Status	sc-win32-status	The Windows status code
14.	Bytes Sent	sc-bytes	Number of bytes sent by the Server
15.	Bytes Received	cs-bytes	Number of bytes received by the Client
16.	Time taken	time-taken	Time taken to send the response in milliseconds

Example:
#Software: Microsoft Internet Information Services 6.0
#Version: 1.0
#Date: 2012-02-13 18:31:40
#Fields: date time s-sitename s-ip cs-method cs-uri-stem cs-uri-query c-ip cs(User-Agent) cs(Referer) sc-status sc-substatus sc-win32-status sc-bytes cs-bytes time-taken
2012-02-13 18:31:39 W3SVC1092988203 64.34.127.116 POST /temp.htm - 210.212.171.169 Mozilla/5.0+(Windows;+U;+Windows+NT+6.1;+en-US;+rv:1.9.2.13)+Gecko/20101203+Firefox/3.6.13 - 405 0 1 1791 660 296

Usage information can be used to restructure a Web site in order to better serve the needs of users of a site. Usage information can also be used to directly aid site navigation by providing a list of "popular" destinations from a particular Web page. Some of the data mining algorithms that are commonly used in Web Usage Mining are association rule generation, sequential pattern generation and clustering. Sequential pattern mining aims at discovering frequent sub-sequences in a sequence database. There are many sequential pattern algorithms applied to Web Usage Mining: (i) PSP+, based on candidate generation and test heuristics (ii) FreeSpan, based on the integration of frequent sequence mining and frequent pattern mining, and (iii) A new projection-based method for mining sequential patterns, called *PrefixSpan*, uses the frequent sequence lattice to partition the database. Its major idea is that, instead of projecting sequence databases by considering all the possible occurrences of frequent subsequences, the projection is based only on frequent prefixes because any frequent subsequence can always be found by growing a frequent prefix.

II. LITERATURE SURVEY

A. *Log files came into existence?*

Web server log files were used initially by the webmasters and system administrators for the purposes of "how much traffic they are getting, how many requests fail, and what kind of errors are being generated", etc. However, Web server log files can also record and trace the visitors' on-line behaviors. For example, after some basic traffic analysis, the log files can help us answer questions such as "from what search engine are visitors coming? What pages are the most and least popular? Which browsers and operating systems are most commonly used by visitors?" Now for an e-commerce company this log files can be used in detecting future customers likely to make a large number of purchases, or predicting which online visitors will click on what ads or banners based on observation of prior visitors who have behaved both positively and negatively to the advertisement banners [1].

B. *Need for Web Usage Mining*

The rapid e-commerce growth has made both business community and customers face a new situation. Due to intense competition on the one hand and the customer's option to choose from several alternatives, the business community has realized the necessity of intelligent marketing strategies and relationship management. Web usage mining attempts to discover useful knowledge from the secondary data obtained from the interactions of the users with the Web. Web usage mining has become very critical for effective Web site management, creating adaptive Web sites, business and support services, personalization, and network traffic flow analysis and so on [4].

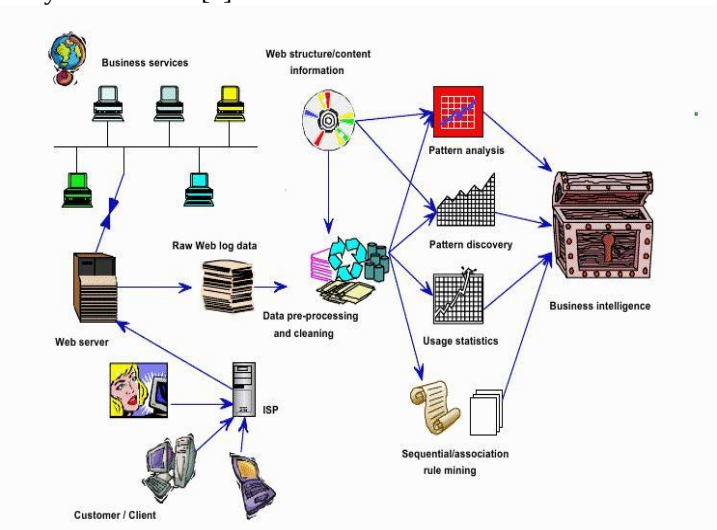

Fig.1 Web Usage Mining Framework

C. *Comparison of Different Techniques*

Different combinations of mining techniques were already suggested for web access recommendation such as GSP, FreeSpan, PrefixSpan, etc [5].

1) *GSP*

Input : A sequence database S, and minimum support threshold *min_sup*
Output : The complete set of frequent sequential patterns.
Idea : Adopts a multiple-pass, candidate generation-and-test approach in sequential pattern mining. This is outlined as follows. The first scan finds all of the frequent items which form the set of single item frequent sequences. Each subsequent pass starts with a *seed set* of sequential patterns, which is the set of sequential patterns found in the previous pass. This seed set is used to generate new potential patterns; called *candidate sequences*. Each candidate sequence contains one more item than a seed sequential pattern, where each

element in the pattern may contain one or multiple items. The number of items in a sequence is called the *length* of the sequence. So, all the candidate sequences in a pass will have the same length. The scan of the database in one pass finds the support for each candidate sequence. All of the candidates whose support in the database is no less than min support from the set of the newly found sequential patterns. This set then becomes the seed set for the next pass. The algorithm terminates when no new sequential pattern is found in a pass, or no candidate sequence can be generated.

Shortcomings of Apriori-like approaches:

- Potentially huge set of candidate sequences
- Multiple scans of databases
- Difficulties at mining long sequential patterns

2) FreeSpan

Idea: Its general idea is to use frequent items to recursively project sequence databases into a set of smaller projected databases and grow subsequence fragments in each projected database. This process partitions both the data and the set of frequent patterns to be tested, and confines each test being conducted to the corresponding smaller projected database. Our performance study shows that FreeSpan mines the complete set of patterns and is efficient and runs considerably faster than the Apriori-based GSP algorithm. However, since a subsequence may be generated by any substring combination in a sequence, projection in FreeSpan has to keep the whole sequence in the original database without length reduction. Moreover, since the growth of a subsequence is explored at any split point in a candidate sequence, it is costly.

Drawback:

Comparing with frequent pattern-guided projection, employed in FreeSpan, prefix-projected pattern growth is more progressive. Even in the worst case, PrefixSpan still guarantees that projected databases keep shrinking and only takes care of postfixes. When mining in dense databases, FreeSpan cannot gain much from projections, whereas PrefixSpan can cut both the length and the number of sequences in projected databases dramatically.

3) PrefixSpan

Idea: Examine only the prefix subsequences and project only their corresponding postfix subsequences into projected databases. In each projected database, sequential patterns are grown by exploring only local frequent patterns.

Advantage:

It requires fewer projections and quickly shrinks sequences. PrefixSpan mines the complete set of patterns and is efficient and runs considerably faster than both Apriori-based GSP algorithm and FreeSpan.

Drawback:

The major cost of PrefixSpan is the construction of projected databases. In the worst case, PrefixSpan constructs a projected database for every sequential pattern. If there are a good number of sequential patterns, the cost is non-trivial.

III. WEB USAGE MINING PROCESS

Fig. 1 shows various tasks required for Web usage Mining [6]

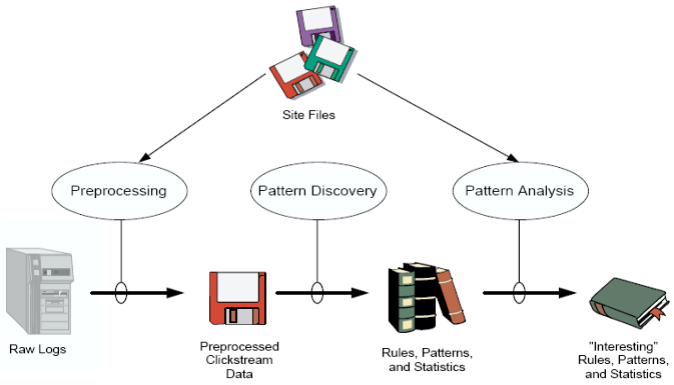

Fig. 2 Web Usage Mining Process

Following modules are required for Web Usage Mining using PrefixSpan algorithm [7]:

A. Data cleaning:

Purpose: To eliminate irrelevant records.

The records of graphics, videos and the format information are irrelevant. Such records have filename suffixes GIF, JPEG, CSS, and so on, which can be found in the URI field of the every record.

INPUT **:** Raw log files.
OUTPUT **:** Cleaned Log files

B. User identification

Purpose: To identify individual users accessing the web site and pages accessed by users.

Individual users are identified by IP + Agent.

INPUT : Cleaned logs stored in the database.
OUTPUT : User's separated based on IP+ Agent.

C. Session identification

*Purpose: T*o divide the web page accesses of each user into individual sessions.

A session is a series of web pages user browse in a single access.

INPUT : User's separated in User Identification
OUTPUT : Session table of user containing session id and sequence (Ordered list of pages accessed by user in that session)

D. PrefixSpan Algorithm

Mines frequent sequences by intermediate database generation. Projected databases keep shrinking as the length of browsing pages increases.

INPUT : Sequence Database.
OUTPUT : Set of sequential Patterns.

E. Pattern Tree Construction

INPUT : Set of Sequential patterns.
OUTPUT : Pattern tree of sequential web access pattern

F. Recommendation rule generation

INPUT :

T – Pattern-tree based on a support threshold MinSup

S = a1a2… an - Current access sequence of a user

MinLength - Minimum length of access sequence

MaxLength - Maximum length of access sequence, which should be less than the depth of the Pattern-tree

OUTPUT : RR – recommendation rule of a set of ordered access events for *S.*

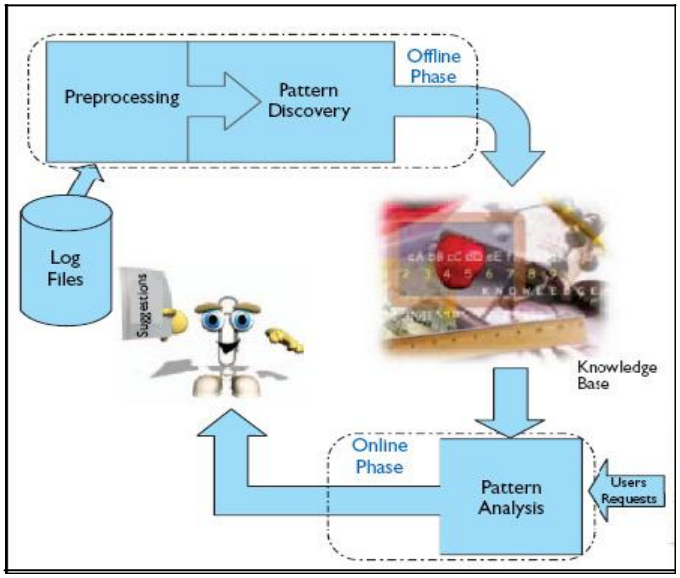

Fig. 3.Architecture of the SUGGEST online Recommender System

IV. PREFIXSPAN ALGORITHM

A. Definition

- *Item set:* An item set is a non-empty set of items
- *Sequence:* Sequence is an ordered list of item sets
- *Length of Sequence:* The length of a sequence is the total number of item occurrences in it. Example:- Sequence <a(abc)(ac)d(cf)> has 5 elements and 9 items
- *Subsequence:* Let SA and SB respectively denote two sequences <A1A2…An> and <B1 B2…Bm >, where Ai's and Bj's are item sets and m ≥ n. If there exist integers i1<i2< … <in, such that A1⊆Bi1, A2⊆Bi2… and An⊆ Bin, it is said that SB contains SA and SA is a subsequence of SB.
- *Support:* The support count of a sequence is the number of sequences that support it.
- *Frequent sequence:* If the support count of a sequence is larger than a user-specified *minimum support count*, we call it a *frequent sequence*.

B. Steps of algorithm

Set min_support for page, so that sequence of pages which have support less than min_support are discarded and is not considered for next pass [8].

1) *Find length-1 sequential patterns.* Scan S once to find all frequent items in sequences. Each of these frequent items is a length-1 sequential pattern. They are <a>:4, <b>:4, <c>:4, <d>:3, <e>:3, <f>:3, where <pattern>: count represents the pattern and its associated support count.
2) *Step 2: Divide search space*. The complete set of sequential patterns can be partitioned into the following six subsets according to the six prefixes:
   - The ones having prefix <a>;
   - The ones having prefix <b>;
   - ……
   - The ones having prefix <f>
3) *Step 3: Find subsets of sequential patterns*. The subsets of sequential patterns can be mined by constructing corresponding *projected databases* and mine each recursively.

Example: Here consider <f> as length-1 pattern.
Min_sup=2; Support of <f>=3

TABLE 2
Sequence Database

SID	SEQUENCE
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

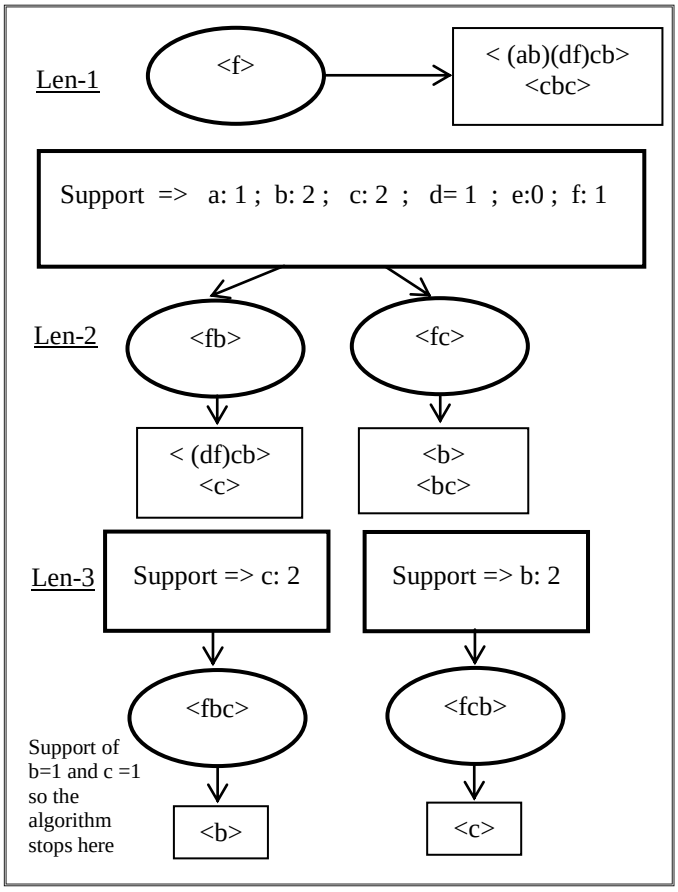

Fig. 4 Example of sequential pattern mining using PrefixSpan algorithm

V. CONCLUSION

Web usage mining model is a kind of mining server logs. It plays an important role in realizing enhancing the usability of the website design, the improvement of customers' relations and improving the requirement of system performance and so on. Web Usage Mining is a new research field that is avidly followed by many scholars and commercial businesses and its importance will continue to grow with the popularity of WWW and undoubtedly will have a significant impact on the study of online user behavior.

PrefixSpan algorithm is an efficient algorithm which is able to efficiently mine useful sequential patterns from the knowledge base obtained from historical usage data (Log files) and to generate a list of links to pages (suggestions) of potentially interest for the user.

REFERENCES

[1] Behzad Mortazavi-Asl, "Discovering and Mining User Web-Page Traversal Patterns", http://citeseerx.ist.psu.edu, 2001.

[2] Hengshan Wang, Cheng Yang and Hua Zeng, "Design and Implementation of a Web Usage Mining Model Based On Fpgrowth and Prefixspan", http://www.iima.org, 2006.

[3] Federico Michele Facca and Pier Luca Lanzi, "Recent Developments in Web Usage Mining Research", http://www.ualberta.ca, 2003.

[4] Ajith Abraham, "Business Intelligence from Web Usage Mining", http://www.softcomputing.net, 2003.

[5] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal and Mei-Chun Hsu, "Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach", http://www.cs.sfu.ca, 2004.

[6] Jaideep Srivastava, Robert Cooley, Mukund Deshpande and Pang-Ning Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data", http://nlp.uned.es, 2000.

[7] Ding-Ying Chiu, Yi-Hung Wu and Arbee L.P. Chen, "An Efficient Algorithm for Mining Frequent Sequences by a New Strategy without Support Counting", http://www.cs.nthu.edu.tw, 2004.

[8] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl and Helen Pinto, "PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth", http://www.cs.sfu.ca, 2004.