Manisha Valera, Kirit Rathod / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 1, January -February 2013, pp.269-380

A Novel Approach of Mining Frequent Sequential Pattern from Customized Web Log Preprocessing

Manisha Valera*, Kirit Rathod(Guide)**
* (Department of Computer Engineering, C.U. Shah College Of Engineering and Technology, Gujarat)
**(Department of Computer Engineering, C.U. Shah College Of Engineering and Technology, Gujarat)

ABSTRACT

Millions of visitors interact daily with web sites around the world. The several kinds of data have to be organized in a manner that they can be accessed by several users effectively and efficiently. Web mining is the extraction of exciting and constructive facts and inherent information from artifacts or actions related to the WWW. Web usage mining is a kind of data mining method that can be useful in recommending the web usage patterns with the help of users' session and behavior. Web usage mining includes three process, namely, preprocessing, pattern discovery and pattern analysis. After the completion of these three phases the user can find the required usage patterns and use this information for the specific needs. Web usage mining requires data abstraction for pattern discovery. This data abstraction is achieved through data preprocessing. Experiments have proved that advanced data preprocessing technique can enhanced the quality of data preprocessing results. To capture users' web access behavior, one promising approach is web usage mining which discovers interesting and frequent user access patterns from web logs. Sequential Web page Access pattern mining has been a focused theme in data mining research for over a decade with wide range of applications. The aim of discovering frequent sequential access (usage) patterns in Web log data is to obtain information about the navigational behavior of the users. This can be used for advertising purposes, for creating dynamic user profiles etc. In this paper we survey about the Sequential Pattern Mining Methods.

Keywords - Web Usage Mining (WUM), Preprocessing, Pattern Discovery, Pattern Analysis, Weblog,Sequential Patterns .

I. INTRODUCTION

In this world of Information Technology, Every day we have to go through several kind of information that we need and what we do? Today, internet is playing such a vital role in our everyday life that it is very difficult to survive without it. In addition, survival of plentiful data in the network and the varying and heterogeneous nature of the web, web searching has become a tricky procedure for the majority of the users. In the last fifteen years, the growth in number of web sites and visitors to those web sites has increased exponentially. The number of users by December 31, 2011 was 2,267,233,742 which is 32.7% of the world's population.[111] Due to this growth a huge quantity of web data has been generated.[1]

To mine the interesting data from this huge pool, data mining techniques can be applied. But the web data is unstructured or semi structured. So we can not apply the data mining techniques directly. Rather another discipline is evolved called web mining which can be applied to web data. Web mining is used to discover interest patterns which can be applied to many real world problems like improving web sites, better understanding the visitor's behavior, product recommendation etc.

The web data is:
1. Content: The visible data in the Web pages or the information which was meant to be imparted to the users. A major part of it includes text and graphics (images).
2. Structure: 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. The principal kind of inter-page structure information is the hyper-links used for site navigation.
3. Usage: 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 depending on the log format.

Data is collected in web server when user accesses the web and might be represented in standard formats. The log format of the file is Common log formats, which consists attributes like IP address, access date and time, request method (GET or POST), URL of page accessed, transfer protocol, success return code etc. In order to discover access pattern, preprocessing is necessary, because raw data coming from the web server is incomplete and only few fields are available for pattern discovery. Main objective of this paper is to understand the preprocessing of usage data.

II. DATA SOURCES

The data sources used in Web Usage Mining may include web data repositories like[5]:

```
213.135.131.79 - - [15/May/2002:19:21:49 -0400]
"GET /features.htm HTTP/1.1" 200 9955
```

Fig .1 A Sample Log Entry

1. Web Server Logs These are logs which maintain a history of page requests. The W3C maintains a standard format for web server log files. More recent entries are typically appended to the end of the file. Information about the request, including client IP address, request date/time, page requested, HTTP code, bytes served, user agent, and referrer are typically added. These data can be combined into a single file, or separated into distinct logs, such as an access log, error log, or referrer log. However, server logs typically do not collect user-specific information. These files are usually not accessible to general Internet users, only to the webmaster or other administrative person. A statistical analysis of the server log may be used to examine traffic patterns by time of day, day of week, referrer, or user agent

2. Proxy Server Logs A Web proxy is a caching mechanism which lies between client browsers and Web servers. It helps to reduce the load time of Web pages as well as the network traffic load at the server and client side. Proxy server logs contain the HTTP requests from multiple clients to multiple Web servers. This may serve as a data source to discover the usage pattern of a group of anonymous users, sharing a common proxy server.

3. Browser Logs Various browsers like Mozilla, Internet Explorer Opera etc. can be modified or various JavaScript and Java applets can be used to collect client side data. This implementation of client-side data collection requires user cooperation, either in enabling the functionality of the JavaScript and Java applets, or to voluntarily use the modified browser.[2]

III. CLASSIFICATION OF WEB MINING

Web mining can be categorized into three areas of interest based on which part of the web to mine[3]:
1. Web Content Mining
2. Web Structure Mining
3. Web Usage Mining

1. Web Content Mining

It deals with discovering important and useful knowledge from web page contents. It contains unstructured information like text, image, audio, and video. Search engines, subject directories, intelligent agents, cluster analysis are employed to find what a user might be looking for. We can automatically classify and cluster web pages according to their topics and discover patterns in web pages to extract useful data such as descriptions of products, postings of forums etc.
The various contents of Web Content Mining are

1.1 Web Page: A Web page typically contains a mixture of many kinds of information, e.g., main content, advertisements, navigation panels, copyright notices, etc.

1.2 Search Page: A search page is typically used to search a particular Web page of the site, to be accessed numerous times in relevance to search queries. The clustering and organization in a content database enables effective navigation of the pages by the customer and search engines.

1.3 Result page A result page typically contains the results, the web pages visited and the definition of last accurate result in the result pages of content mining.

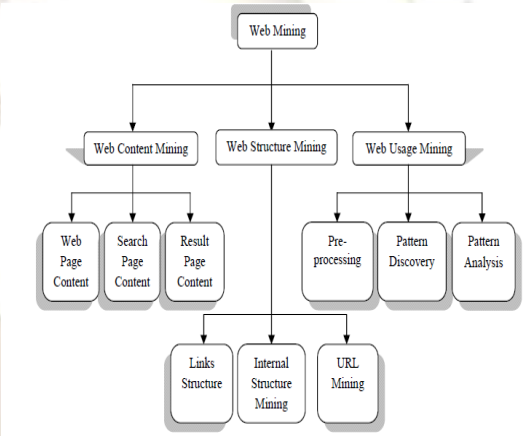

Fig. 2 Classification of Web Mining

2. Web Structure Mining

It deals with discovering and modeling the link structure of the web. Web information retrieval tools make use of only the text available on web pages but ignoring valuable information contained in web links. Web structure mining aims to generate structural summary about web sites and web pages. The main focus of web structure mining is on link information. Web structure mining plays a vital role with various benefits including quick response to the web users, reducing lot of HTTP transactions between users and server. This can help in discovering similarity between sites or in discovering important sites for a particular topic.

2.1 Links Structure Link analysis is an old area of research. However, with the growing interest in Web mining, the research of structure analysis had increased and these efforts have resulted in

a newly emerging research area called Link Mining. It consists Link-based Classification, Link-based Cluster Analysis, Link Type, Link Strength and Link Cardinality.

2.2 Internal Structure Mining It can provide information about page ranking or authoritativeness and enhance search results through filtering i.e., tries to discover the model underlying the link structures of the web. This model is used to analyze the similarity and relationship between different web sites.

2.3 URL Mining It gives a hyperlink which is a structural unit that connects a web page to different location, either within the same web page (intra_document hyperlink) or to a different web page (inter_document) hyperlink.

3. Web Usage Mining

It is the application of data mining techniques to discover interesting usage patterns from Web data, in order to understand and better serve the needs of Web-based applications. Usage data captures the identity or origin of Web users along with their browsing behavior at a Web site. Web usage mining itself can be classified further depending on the kind of usage data considered. There are three main tasks for performing Web Usage Mining or Web Usage Analysis.

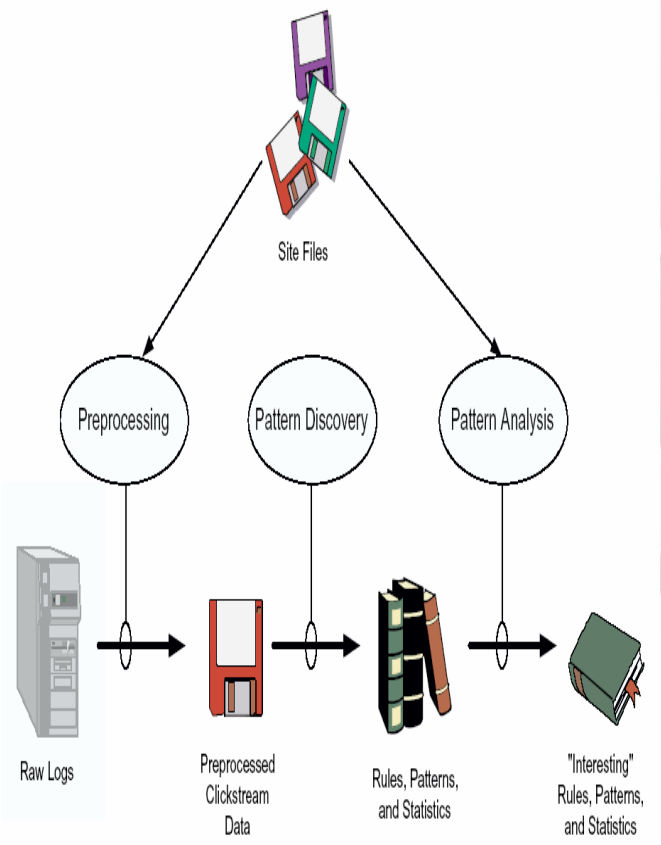

Fig.3 Process Of Web Usage Mining

Five major steps followed in web usage mining are:

3.1 Data collection Web log files, which keeps track of visits of all the visitors

3.2 Data Integration Integrate multiple log files into a single file

3.3 Data preprocessing Cleaning and structuring data to prepare for pattern extraction

3.4 Pattern extraction Extracting interesting patterns

3.5 Pattern analysis and visualization Analyze the extracted pattern

3.6 Pattern applications Apply the pattern in real world problems

IV. WEB USAGE MINING PROCESS

The main processes in Web Usage Mining are:

1. Preprocessing Data preprocessing describes any type of processing performed on raw data to prepare it for another processing procedure. Commonly used as a preliminary data mining practice, data preprocessing transforms the data into a format that will be more easily and effectively processed for the purpose of the user.

2. Pattern Discovery Web Usage mining can be used to uncover patterns in server logs but is often carried out only on samples of data. The mining process will be ineffective if the samples are not a good representation of the larger body of data. The following are the pattern discovery methods.

1. Statistical Analysis
2. Association Rules
3. Clustering
4. Classification
5. Sequential Patterns

3. Pattern Analysis This is the final step in the Web Usage Mining process. After the preprocessing and pattern discovery, the obtained usage patterns are analyzed to filter uninteresting information and extract the useful information. The methods like SQL (Structured Query Language) processing and OLAP (Online Analytical Processing) can be used.

V. DATA PREPROCESSING

It is important to understand that the quality data is a key issue when we are going to mining from it. Nearly 80% of mining efforts often spend to improve the quality of data[8]. The data which is obtained from the logs may be incomplete, noisy and inconsistent. The attributes that we can look for in quality data includes accuracy, completeness, consistency, timeliness, believability,

interpretability and accessibility. There is a need to preprocess data to make it have the above mentioned attributes and to make it easier to mine for knowledge.

There are four steps in preprocessing of log data: data cleaning, user identification, session identification, path completion.

1. Data cleaning

The process of data cleaning is removal of outliers or irrelevant data. The Web Log file is in text format then it is required to convert the file in database format and then clean the file. First, all the fields which are not required are removed and finally we will have the fields like date, time, client ip, URL access, Referrer and Browser used/ Access log files consist of large amounts of HTTP server information. Analyzing, this information is very slow and inefficient without an initial cleaning task. Every time a web browser downloads a HTML document on the internet the images are also downloaded and stored in the log file. This is because though a user does not explicitly request graphics that are on a web page, they are automatically downloaded due to HTML tags. The process of data cleaning is to remove irrelevant data. All log entries with file name suffixes such as gif, JPEG, jpeg, GIF, jpg, JPG can be eliminated since they are irrelevant [4].Web robot (WR) (also called spider or bot) is a software tool that periodically a web site to extract its content[6].Web robot automatically follows all the hyper links from web pages. Search engines such as Google periodically use WRs to gather all the pages from a web site in order to update their search indexes. Eliminating WR generated log entries simplifies the mining task[8]. To identify web robot requests the data cleaning module removes the records containing "Robots.txt" in the requested resource name (URL). The HTTP status code is then considered in the next process of cleaning by examining the status field of every record in the web access log, the records with status code over 299 or under 200 are removed because the records with status code between 200 and 299, gives successful response[7].

2. User Identification

This step identify individual user by using their IP address. If new IP address, there is new user. If IP address is same but browser version or operating system is different then it represents different user. User identification an important issue is how exactly the users have to be distinguished. It depends mainly on the task for the mining process is executed. In certain cases the users are identified only with their IP addresses .

Problem at time of User Identification

User's identification is, to identify who access Web site and which pages are accessed. If users have login of their information, it is easy to identify them. In fact, there are lots of user do not register their information. What's more, there are great numbers of users access Web sites through, agent, several users use the same computer, firewall's existence, one user use different browsers, and so forth. All of problems make this task greatly complicated and very difficult, to identify every unique user accurately. We may use cookies to track users' behaviors. But considering personage privacy, many users do not use cookies, so it is necessary to find other methods to solve this problem. For users who use the same computer or use the same agent, how to identify them?

As presented in [10], it uses heuristic method to solve the problem, which is to test if a page is requested that is not directly reachable by a hyperlink from any of the pages visited by the user, the heuristic assumes that there is another user with the same computer or with the same IP address. Ref. [9] presents a method called navigation patterns to identify users automatically. But all of them are not accurate because' they only consider a few aspects that influence the process of users identification.

The success of the web site cannot be measured only by hits and page views. Unfortunately, web site designers and web log analyzers do not usually cooperate. This causes problems such as identification unique user's, construction discrete user's sessions and collection essential web pages for analysis. The result of this is that many web log mining tools have been developed and widely exploited to solve these problems.

3. Session Identification

To group the activities of a single user from the web log files is called a session. As long as user is connected to the website, it is called the session of that particular user. Most of the time, 30 minutes time-out was taken as a default session time-out. A session is a set of page references from one source site during one logical period. Historically a session would be identified by a user logging into a computer, performing work and then logging off. The login and logoff represent the logical start and end of the session.

4. Path completion

Path completion step is carried out to identify missing pages due to cache and 'Back'. Path Set is the incomplete accessed pages in a user session. It is extracted from every user session set. Path Combination and Completion: Path Set (PS) is access path of every USID identified from USS. It is defined as:

PS = {USID,(URI1,Date1, RLength1),… (URIk, Datek,
RLengthk)}

where, Rlength is computed for every record in data cleaning stage.[6] After identifying path for each USID path combination is done if two consecutive pages are same. In the user session if any of the URL specified in the Referrer URL is not equal to the URL in the previous record then that URL in the Referrer Url field of current record is inserted into this session and thus path completion is obtained. The next step is to determine the reference length of new appended pages during path completion and modify the reference length of adjacent ones. Since the assumed pages are normally considered as auxiliary pages the length is determined by the average reference length of auxiliary pages. The reference length of adjacent pages is also adjusted.

VI. PATTERN DISCOVERY

Pattern discovery draws upon methods and algorithms developed from several fields such as statistics, data mining, machine learning and pattern recognition. Various data mining techniques have been investigated for mining web usage logs. They are statistical analysis, association rule mining, clustering, classification and sequential pattern mining.

1. Statistical Analysis

Statistical techniques are the most common method to extract knowledge about visitors to a Web site. By analyzing the session file, one can perform different kinds of descriptive statistical analyses (frequency, mean, median, etc.) on variables such as page views, viewing time and length of a navigational path. Many Web traffic analysis tools produce a periodic report containing statistical information such as the most frequently accessed pages, average view time of a page or average length of a path through a site.

Statistics	Detailed Information
Website Activity Statistics	Total number of visits Mean number of hits Successful/failed/redirected/ hits Average view time Average length of a path through a site
Troubleshooting/Diagnostic Statistics	Server errors Page not found errors
Server Statistics	Top pages visited Top entry/exit pages

Table 1: Important Statistical Information Discovered From Web Logs

This report may include limited low-level error analysis such as detecting unauthorized entry points or finding the most common invalid URI. Despite lacking in the depth of its analysis, this type of knowledge can be potentially useful for improving the system performance, enhancing the security of the system, facilitating the site modification task, and providing support for marketing decisions. Statistical techniques are the most commonly used methods for extracting knowledge from web logs. The useful statistical information discovered from web logs is listed in Table1. Many web traffic analysis tools, such as Web Trends and Web Miner, are available for generating web usage statistics.

2. Path Analysis

There are many different types of graphs that can be formed for performing path analysis. Graph may be representing the physical layout of a Web site, with Web pages as nodes and hypertext links between pages as directed edges. Graphs may be formed based on the types of Web pages with edges representing similarity between pages, or creating edges that give the number of users that go from one page to another. Path analysis could be used to determine most frequently visited paths in a Web site. Other examples of information that can be discovered through path analysis are: 80% of clients left the site after four or less page references. This example indicates that many users don't browse more than four pages into the site, it can be concluded that important information is contained within four pages of the common site entry points.

3. Association Rules

For web usage mining, association rules can be used to find correlations between web pages (or products in an e-commerce website) accessed together during a server session. Such rules indicate the possible relationship between pages that are often viewed together even if they are not directly connected, and can reveal associations between groups of users with specific interests. Apart from being exploited for business applications, the associations can also be used for web recommendation, personalization or improving the system's performance through predicting and prefetching of web data. Discovery of such rules for organizations engaged in electronic commerce can help in the development of effective marketing strategies. But, in addition, association rules discovered from WWW access logs can give an indication of how to best organize the organization's Web space. For example,

- if one discovers that 80% of the clients accessing /computer/products/printer.html and /computer/products/scanner.html also accessed,
- but only 30% of those who accessed /computer/products also accessed computer/products/scanner.html, then it is likely that some information in printer.html leads clients to access scanner.html.

This correlation might suggest that this information should be moved to a higher level to increase access to scanner.html. This also helps in making business strategy that people who want to buy printer; they are also interested in buying scanner. So vendors can offer some discount on buying combo pack of printer and scanner. Or they can offer discount on one item for the purchase of both or they can apply buy one, get one free strategy.

Since usually such transaction databases contain extremely large amounts of data, current association rule discovery techniques try to prune the search space according to support for items under consideration. Support is a measure based on the number of occurrences of user transactions within transaction logs. Discovery of such rules for organizations engaged in electronic commerce can help in the development of effective marketing strategies.

4. Sequential Patterns

The problem of discovering sequential patterns is to find inter-transaction patterns such that the presence of a set of items is followed by another item in the time-stamp ordered transaction set. In Web server transaction logs, a visit by a client is recorded over a period of time. The time stamp associated with a transaction in this case will be a time interval which is determined and attached to the transaction during the data cleaning or transaction identification processes. The discovery of sequential patterns in Web server access logs allows Web-based organizations to predict user visit patterns and helps in targeting advertising aimed at groups of users based on these patterns. By analyzing this information, the Web mining system can determine temporal relationships among data items such as the following:

- 30% of clients who visited /company/products/, had done a search in Yahoo, within the past week on keyword data mining; or
- 60% of clients who placed an online order in /computer/products/webminer.html, also placed an online order in /computer/products/iis.html within 10 days.

From these relationships, vendors can develop strategies and expand business.

5. Clustering and Classification

In Web mining, classification techniques allow one to develop a profile for clients who access particular server files based on demographic information available on those clients, or based on their access patterns. For example classification on WWW access logs may lead to the discovery of relationships such as the following:

- clients from state or government agencies who visit the site tend to be interested in the page /company/lic.html or
- 60% of clients, who placed an online order in/company/products /product2, were in the 35-45 age group and lived in Chandigarh.

Clustering analysis allows one to group together clients or data items that have similar characteristics. Clustering of client information or data items on Web transaction logs, can facilitate the development and execution of future marketing strategies, both online and off-line, such as automated return mail to clients fallingwithin a certain cluster, or dynamically changing a particular site for a client, on a return visit, based on past classification of that client. For web usage mining, clustering techniques are mainly used to discover two kinds of useful clusters, namely user clusters and page clusters. User clustering attempts to find groups of users with similar browsing preference and habit, whereas web page clustering aims to discover groups of pages that seem to be conceptually related according to the users' perception. Such knowledge is useful for performing market segmentation in ecommerce and web personalization applications.

VII. SEQUENTIAL PATTERN MINING

The concept of sequence Data Mining was first introduced by Rakesh Agrawal and Ramakrishnan Srikant in the year 1995. The problem was first introduced in the context of market analysis. It aimed to retrieve frequent patterns in the sequences of products purchased by customers through time ordered transactions. Later on its application was extended to complex applications like telecommunication, network detection, DNA research, etc. Several algorithms were proposed. The very first was Apriori algorithm, which was put forward by the founders themselves. Later more scalable algorithms for complex applications were developed. E.g. GSP, Spade, PrefixSpan etc. The area underwent considerable advancements since its introduction in a short span.

1. Basic Concepts of Sequential Pattern Mining

1. Let I = {x1, . . . , xn} be a set of items, each possibly being associated with a set of attributes, such as value, price, profit, calling distance, period, etc. The value on attribute A of item x is denoted by x.A. An itemset is a non-empty subset of items, and an itemset with k items is called a k-itemset.

2. A sequence α = <X1... Xl> is an ordered list of item sets. An itemset Xi ($1 \le i \le l$) in a sequence is called a transaction, a term originated from analyzing customers' shopping sequences in a transaction database. A transaction Xi may have a special attribute, time-stamp, denoted by Xi.time,which registers the time when the transaction was executed. For a sequence α = <X1 ... Xl>, we assume Xi.time < Xj.time for $1 \le i < j \le l$.

3 The number of transactions in a sequence is called the length of the sequence. A sequence with length l is called an l-sequence. For an l-sequence α,we have len(α)= l.Furthermore, the i-th itemset is denoted by α[i]. An itemcan occur at most once in an itemset, but can occurmultiple times in various itemsets in a sequence.

4. A sequence α = <X1 . . . Xn> is called a subsequence of another sequence β = <Y1 . . .Ym> (n ≤ m), and β a super-sequence of α, if there exist integers 1 ≤ i1 < . . < in≤ m such that X1 Yi1 , . . . , Xn Yin.

5. A sequence database SDB is a set of 2-tuples (sid, α), where sid is a sequence-id and α a sequence. A tuple (sid, α) in a sequence database SDB is said to contain a sequence γ if γ is a subsequence of α. The number of tuples in a sequence database SDB containing sequence γ is called the support of γ, denoted by sup (γ). Given a positive integer min_sup as the support threshold, a sequence γ is a sequential pattern in sequence database SDB if sup (γ) ≥ min_sup. The sequential pattern mining problem is to find the complete set of sequential patterns with respect to a given sequence database SDB and a support threshold min_sup.

VIII. CLASSIFICATION OF SEQUENTIAL PATTERN MINING ALGORITHM

In general, there are two main research issues in sequential pattern mining.

1. The first is to improve the efficiency in sequential pattern mining process while the other one is to
2. Extend the mining of sequential pattern to other time-related patterns.

A. Improve the Efficiency by Designing Novel Algorithms

According to previous research done in the field of sequential pattern mining, Sequential Pattern Mining Algorithms mainly differ in two ways [14]:

(1) The way in which candidate sequences are generated and stored. The main goal here is to minimize the number of candidate sequences generated so as to minimize I/O cost.

(2) The way in which support is counted and how candidate sequences are tested for frequency. The key strategy here is to eliminate any database or data structure that has to be maintained all the time for support of counting purposes only.

Based on these criteria's sequential pattern mining can be divided broadly into two parts:

- Apriori Based
- Pattern Growth Based

1. Apriori-Based Algorithms

The Apriori [Agrawal and Srikant 1994] and AprioriAll [Agrawal and Srikant 1995] set the basis for a breed of algorithms that depend largely on the apriori property and use the Apriori-generate join procedure to generate candidate sequences. The apriori property states that "All nonempty subsets of a frequent itemset must also be Frequent". It is also described as antimonotonic (or downward-closed), in that if a sequence cannot pass the minimum support test, its entire super sequences will also fail the test.

Key features of Apriori-based algorithm are: [12]

(1) Breadth-first search: Apriori-based algorithms are described as breath-first (level-wise) search algorithms because they construct all the k-sequences, in kth iteration of the algorithm, as they traverse the search space.

(2) Generate-and-test: This feature is used by the very early algorithms in sequential pattern mining. Algorithms that depend on this feature only display an inefficient pruning method and generate an explosive number of candidate sequences and then test each one by one for satisfying some user specified constraints, consuming a lot of memory in the early stages of mining.

(3) Multiple scans of the database: This feature entails scanning the original database to ascertain whether a long list of generated candidate sequences is frequent or not. It is a very undesirable characteristic of most apriori-based algorithms and requires a lot of processing time and I/O cost.

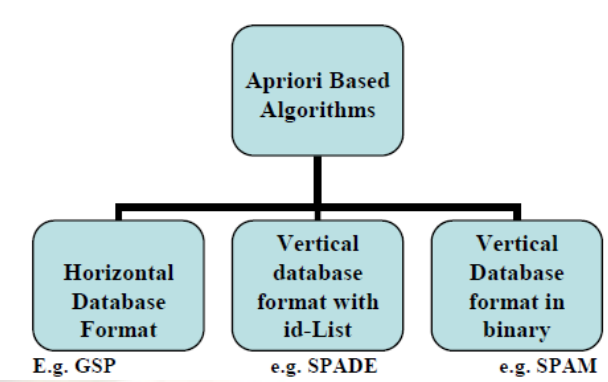

Fig.4 Classification of Apriori-Based Algorithms

i. GSP: The GSP algorithm described by Agrawal and Shrikant [12] makes multiple passes over the data. This algorithm is not a main-memory algorithm. If the candidates do not fit in memory, the algorithm generates only as many candidates as will fit in memory and the data is scanned to count the support of these candidates. Frequent sequences resulting from these candidates are written to disk, while those candidates without minimum support are deleted. This procedure is repeated until all the

candidates have been counted. As shown in Fig 2, first GSP algorithm finds all the length-1 candidates (using one database scan) and orders them with respect to their support ignoring ones for which support < min_sup. Then for each level (i.e., sequences of length-k), the algorithm scans database to collect support count for each candidate sequence and generates candidate length (k+1) sequences from length-k frequent sequences using Apriori. This is repeated until no frequent sequence or no candidate can be found.

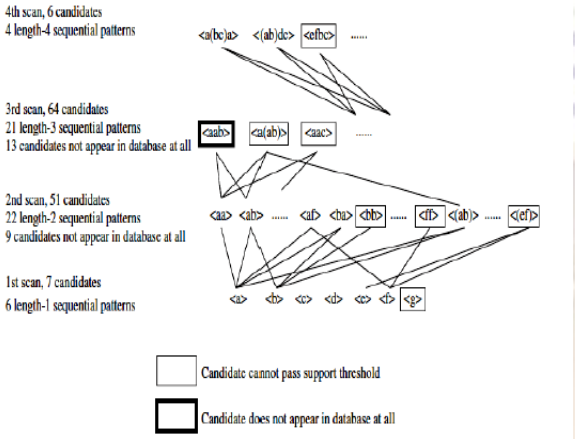

Fig.7 Candidates, Candidate generation and Sequential Patterns in GSP

ii. SPIRIT: The Novel idea of the SPIRIT algorithm is to use regular expressions as flexible constraint specification tool [12]. It involves a generic user-specified regular expression constraint on the mined patterns, thus enabling considerably versatile and powerful restrictions. In order to push the constraining inside the mining process, in practice the algorithm uses an appropriately relaxed, that is less restrictive, version of the constraint. There exist several versions of the algorithm, differing in the degree to which the constraints are enforced to prune the search space of pattern during computation. Choice of regular expressions (REs) as a constraint specification tool is motivated by two important factors. First, REs provide a simple, natural syntax for the succinct specification of families of sequential patterns. Second, REs possess sufficient expressive power for specifying a wide range of interesting, non-trivial pattern constraints.

iii. SPADE: Besides the horizontal formatting method (GSP), the sequence database can be transformed into a vertical format consisting of items' id-lists. The id-list of an item as shown in fig 3, is a list of (sequence-id, timestamp) pairs indicating the occurring timestamps of the item in that sequence. Searching in the lattice formed by id-list intersections, the SPADE (Sequential Pattern Discovery using Equivalence classes) algorithm presented by M.J.Jaki [12] completes the mining in three passes of database scanning. Nevertheless, additional computation time is required to transform a database of horizontal layout to vertical format, which also requires additional storage space several times larger than that of the original sequence database.

Seq. ID	Sequence
1	< a (abc) (ac) d (df) >
2	(ad) c (bc) (ae)
3	< (ef) (ab) (df) c b >
4	< e g (af) c b c >

SID	EID	Items
1	1	a
1	2	abc
1	3	ac
1	4	d
1	5	cf
2	1	ad
2	2	c
2	3	bc
2	4	ae
3	1	ef
3	2	abc
3	3	df
3	4	c
3	5	b
4	1	e
4	2	g
4	3	af
4	4	c
4	5	b
4	6	c

a		b		...
SID	EID	SID	EID	...
1	1	1	2	
1	2	2	3	
1	3	3	2	
2	1	3	3	
2	4	4	5	
3	2			
4	3			

ab			ba			...
SID	EID (a)	EID (b)	SID	EID (b)	EID (a)	...
1	1	2	1	2	3	
2	1	3	2	3	4	
3	2	5				
4	3	5				

aba				...
SID	EID (a)	EID (b)	EID (a)	...
1	1	2	3	
2	1	3	4	

Fig. 3 Working of SPADE algorithm

iv. SPAM: SPAM integrates the ideas of GSP, SPADE, and FreeSpan. The entire algorithm with its data structures fits in main memory, and is claimed to be the first strategy for mining sequential patterns to traverse the lexicographical sequence tree in depth-first fashion. SPAM traverses the sequence tree in depth-first search manner and checks the support of each sequence-extended or item set-extended child against min_sup recursively for efficient support-counting SPAM uses a vertical bitmap data structure representation of the database as shown in fig 4,which is similar to the id list in SPADE. SPAM is similar to SPADE, but it uses bitwise operations rather than regular and temporal joins. When SPAM was compared to SPADE, it was found to outperform SPADE by a factor of 2.5, while SPADE is 5 to 20 times more space-efficient than SPAM, making the choice between the two a matter of a space-time trade-off.

CID	TID	Itemset
1	1	{a, b, d}
1	3	{b, c, d}
1	6	{b, c, d}
2	2	{b}
2	4	{a, b, c}
3	5	{a, b}
3	7	{b, c, d}

Dataset Sorted by CID and TID

CID	TID	{a}	{b}	{c}	{d}
1	1	1	1	0	1
1	3	0	1	1	1
1	6	0	1	1	1
-	-	0	0	0	0
2	2	0	1	0	0
2	4	1	1	1	0
-	-	0	0	0	0
-	-	0	0	0	0
3	5	1	1	0	0
3	7	0	1	1	1
-	-	0	0	0	0
-	-	0	0	0	0

Fig.8 Transformation of Sequence database to Vertical binary format

2. Pattern-Growth Algorithms

Soon after the apriori-based methods of the mid-1990s, the pattern growth-method emerged in the early 2000s, as a solution to the problem of generate-and-test. The key idea is to avoid the candidate generation step altogether, and to focus

the search on a restricted portion of the initial database. The search space partitioning feature plays an important role in pattern-growth. Almost every pattern-growth algorithm starts by building a representation of the database to be mined, then proposes a way to partition the search space, and generates as few candidate sequences as possible by growing on the already mined frequent sequences, and applying the apriori property as the search space is being traversed recursively looking for frequent sequences. The early algorithms started by using projected databases, for example, FreeSpan [Han et al. 2000], PrefixSpan [Pei et al. 2001], with the latter being the most influential.

Key features of pattern growth-based algorithm are:
(1) Search space partitioning: It allows partitioning of the generated search space of large candidate sequences for efficient memory management. There are different ways to partition the search space. Once the search space is partitioned, smaller partitions can be mined in parallel. Advanced techniques for search space partitioning include projected databases and conditional search, referred to as split-and-project techniques.

(2)Tree projection: Tree projection usually accompanies pattern-growth algorithms. Here, algorithms implement a physical tree data structure representation of the search space, which is then traversed breadth-first or depth-first in search of frequent sequences, and pruning is based on the apriori property.

(3) Depth-first traversal: That depth-first search of the search space makes a big difference in performance, and also helps in the early pruning of candidate sequences as well as mining of closed sequences [Wang and Han 2004]. The main reason for this performance is the fact that depth-first traversal utilizes far less memory, more directed search space, and thus less candidate sequence generation than breadth-first or post-order which are used by some early algorithms.

(4)Candidate sequence pruning: Pattern-growth algorithms try to utilize a data structure that allows them to prune candidate sequences early in the mining process. This result in early display of smaller search space and maintain a more directed and narrower search procedure.

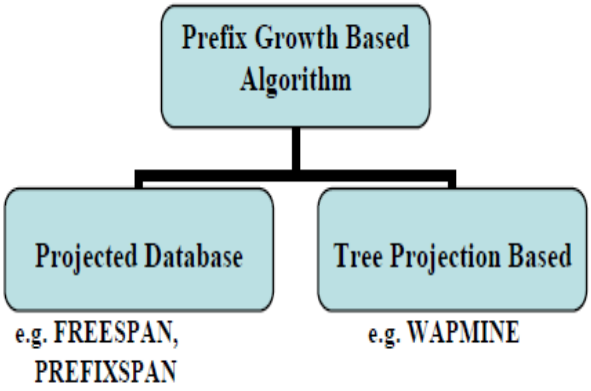

Fig 5: Classification of Prefix Growth based mining algorithm

i. FREESPAN: FreeSpan was developed to substantially reduce the expensive candidate generation and testing of Apriori, while maintaining its basic heuristic. In general, FreeSpan uses frequent items to recursively project the sequence database into projected databases while growing subsequence fragments in each projected database. Each projection partitions the database and confines further testing to progressively smaller and more manageable units. The trade-off is a considerable amount of sequence duplication as the same sequence could appear in more than one projected database. However, the size of each projected database usually (but not necessarily) decreases rapidly with recursion.

ii. WAP-MINE: It is a pattern growth and tree structure-mining technique with its WAP-tree structure. Here the sequence database is scanned only twice to build the WAP-tree from frequent sequences along with their support; a ―header table‖ is maintained to point at the first occurrence for each item in a frequent itemset, which is later tracked in a threaded way to mine the tree for frequent sequences, building on the suffix. The WAP-mine algorithm is reported to have better scalability than GSP and to outperform it by a margin. Although it scans the database only twice and can avoid the problem of generating explosive candidates as in apriori-based methods, WAP-mine suffers from a memory consumption problem, as it recursively reconstructs numerous intermediate WAP-trees during mining, and in particular, as the number of mined frequent patterns increases. This problem was solved by the PLWAP algorithm [Lu and Ezeife 2003], which builds on the prefix using position- coded nodes.

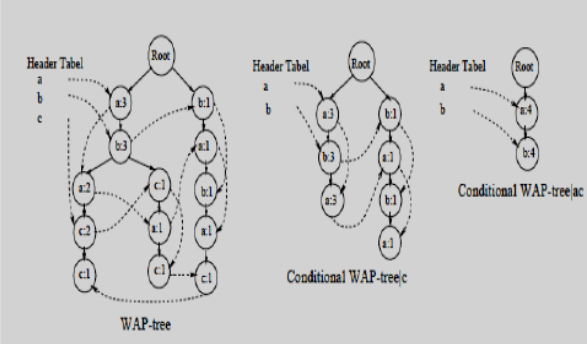

Fig.5 Classification of Prefix Growth based mining algorithm

ii. PREFIXSPAN: The PrefixSpan (Prefix-projected Sequential pattern mining) algorithm presented by Jian Pei, Jiawei Han and Helen Pinto representing the pattern-growth methodology, which finds the frequent items after scanning the sequence database once. The database is then projected as shown in Fig.7, according to the frequent items, into several smaller databases. Finally, the complete set of sequential patterns is found by recursively growing subsequence fragments in each projected database. Although the PrefixSpan algorithm successfully discovered patterns employing the divide-and-conquer strategy, the cost of memory space might be high due to the creation and processing of huge number of projected sub-databases.

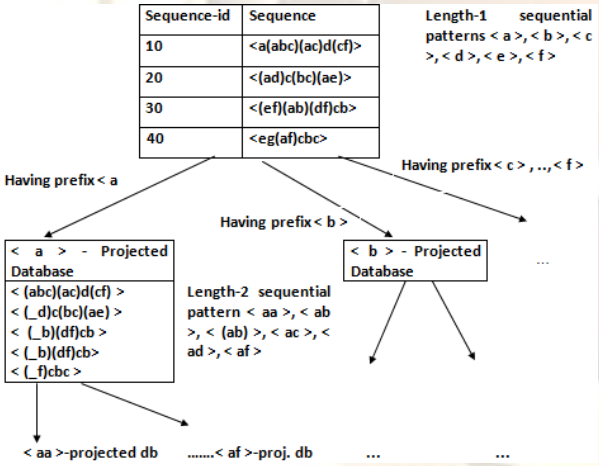

Fig.7 Construction of Projected Databases in PrefixSpan Algorithm

B. Extensions of Sequential Pattern Mining to Other Time-Related Patterns

Sequential pattern mining has been intensively studied during recent years; there exists a great diversity of algorithms for sequential pattern mining. Along with that Motivated by the potential applications for the sequential patterns, numerous extensions of the initial definition have been proposed which may be related to other types of time-related patterns or to the addition of time constraints. Some extensions of those algorithms for special purposes such as multidimensional, closed, time interval, and constraint based sequential pattern mining are discussed in following section.

i. Multidimensional Sequential Pattern Mining

Mining sequential patterns with single dimension means that we only consider one attribute along with time stamps in pattern discovery process, while mining sequential patterns with multiple dimensions we can consider multiple attributes at the same time. In contrast to sequential pattern mining in single dimension, mining multiple dimensional sequential patterns introduced by Helen Pinto and Jiawei Han can give us more informative and useful patterns. For example we may get a traditional sequential pattern from the supermarket database that after buying product a most people also buy product b in a defined time interval. However, using multiple dimensional sequential pattern mining we can further find different groups of people have different purchase patterns.

For example, M.E. students always buy product b after they buy product a, while this sequential rule weakens for other groups of students. Hence, we can see that multiple-dimensional sequential pattern mining can provide more accurate information for further decision support.

ii. Discovering Constraint Based Sequential Pattern Although efficiency of mining the complete set of sequential patterns has been improved substantially, in many cases, sequential pattern mining still faces tough challenges in both effectiveness and efficiency. On the one hand, there could be a large number of sequential patterns in a large database. A user is often interested in only a small subset of such patterns. Presenting the complete set of sequential patterns may make the mining result hard to understand and hard to use. To overcome this problem Jian Pei, Jiawei Han and Wei Wang [12] have systematically presented the problem of pushing various constraints deep into sequential pattern mining using pattern growth methods. Constraint-based mining may overcome the difficulties of effectiveness and efficiency since constraints usually represent user's interest and focus, which limits the patterns to be found to a particular subset satisfying some strong conditions. (Pei, Han, & Wang, 2007) mention seven categories of constraints: 1. Item constraint: An item constraint specifies subset of items that should or should not be present in the patterns. 2. Length constraint: A length constraint specifies the requirement on the length of the patterns, where the length can be either the number of occurrences of items or the number of transactions. 3. Super-pattern constraint: Super-patterns are ones that contain at least one of a particular set of patterns as sub-patterns. 4. Aggregate constraint: An aggregate constraint is the constraint on an aggregate of items in a pattern, where the aggregate function can be sum, avg, max,

min, standard deviation, etc. 5. Regular expression constraint: A regular expression constraint CRE is a constraint specified as a regular expression over the set of items using the established set of regular expression operators, such as disjunction and Kleene closure. 6. Duration constraint: A duration constraint is defined only in sequence databases where each transaction in every sequence has a time-stamp. It requires that the sequential patterns in the sequence database must have the property such that the time-stamp difference between the first and the last transactions in a sequential pattern must be longer or shorter than a given period. 7. Gap constraint: A gap constraint set is defined only in sequence databases where each transaction in every sequence has a timestamp. It requires that the sequential patterns in the sequence database must have the property such that the timestamp difference between every two adjacent transactions must be longer or shorter than given gap. Other Constraints: R (Recency) is specified by giving a recency minimum support (r_minsup), which is the number of days away from the starting date of the sequence database. For example, if our sequence database is from 27/12/2007 to 31/12/2008 and if we set r_minsup = 200 then the recency constraint ensures that the last transaction of the discovered pattern must occur after 27/12/2007+200 days. In other words, suppose the discovered pattern is < (a), (bc)>, which means ―after buying item a, the customer returns to buy item b and item c‖. Then, the transaction in the sequence that buys item b and item c must satisfy recency constraint. [17] M (Monetary) is specified by giving monetary minimum support (m_minsup). It ensures that the total value of the discovered pattern must be greater than m_minsup. Suppose the pattern is < (a), (bc)>. Then we can say that a sequence satisfies this pattern with respect to the monetary constraint, if we can find an occurrence of pattern < (a), (bc)> in this data sequence whose total value must be greater than m_minsup. C (Compactness) constraint, which means the time span between the first and the last purchase in a customer sequence, must be within a user-specified threshold. This constraint can assure that the purchasing behavior implied by a sequential pattern must occur in a reasonable period. Target-Oriented A target-oriented sequential pattern is a sequential pattern with a concerned itemset in the end of pattern. For most decision makers, when they want to make efficient marketing strategies, they usually concern the happening order of a concerned itemsets only, and thus, most sequential patterns discovered by using traditional algorithms are irrelevant and useless.

iii. Discovering Time-interval Sequential Pattern

Although sequential patterns can tell us what items are frequently bought together and in what order, they cannot provide information about the time span between items for further decision support. In other words, although we know which items will be bought after the preceding items, we have no idea when the next purchase will happen. Y. L. Chen, M. C. Chiang, and M. T. Kao [12] have given the solution of this problem that is to generalize the mining problem into discovering time-interval sequential patterns, which tells not only the order of items but also the time intervals between successive items. An example of time-interval sequential pattern is (a, I1, b, I2, c), meaning that we buy item a first, then after an interval of I1 we buy item b, and finally after an interval of I2 we buy item c. Similar type of work done by C. Antunes, A. L. Oliveira, by presenting the concept of gap constraint. A gap constraint imposes a limit on the separation of two consecutive elements of an identified sequence. This type of constraints is critical for the applicability of these methods to a number of problems, especially those with long sequence.

iv. Closed Sequential Pattern Mining

The sequential pattern mining algorithms developed so far have good performance in databases consisting of short frequent sequences. Unfortunately, when mining long frequent sequences, or when using very low support thresholds, the performance of such algorithms often degrades dramatically. This is not surprising: Assume the database contains only one long frequent sequence < (a1) (a2) . . . (a100) >, it will generate 2100−1 frequent subsequence if the minimum support is 1, although all of them except the longest one are redundant because they have the same support as that of < (a1) (a2) . . . (a100) > . So proposed an alternative but equally powerful solution: instead of mining the complete set of frequent subsequence, we mine frequent closed subsequence only, i.e., those containing no super-sequence with the same support. This mining technique will generate a significant less number of discovered sequences than the traditional methods while preserving the same expressive power since the whole set of frequent subsequences together with their supports, can be derived easily from the mining results.

IX. CONCLUSION

Preprocessing involves removal of unnecessary data from log file. Log file used for debugging purpose. It has undergone various steps such as data cleaning, user identification, session identification, path completion and transaction identification. Data cleaning phase includes the removal of records of graphics, videos and the format information, the records with the failed HTTP status code and finally robots cleaning. Data preprocessing is an important steps to filter and organize appropriate information before using to

web mining algorithm. Future work needs to be done to combine whole process of WUM. A complete methodology covering such as pattern discovery and pattern analysis will be more useful in identification method.

Web mining is a very broad research area trying to solve issues that arise due to the WWW phenomenon. In this paper a little attempt is made to provide an up-to-date survey of the rapidly growing area of Web Usage mining and how the various pattern discovery techniques help in developing business plans especially in the area of e-business. However, Web Usage mining raises some hard scientific questions that must be answered before robust tools can be developed. This article has aimed at describing such challenges, and the hope is that the research community will take up the challenge of addressing them. Therefore the need for discovering new methods and techniques to handle the amounts of data existing in this universal framework will always exist which help in maintaining the trust between customers and traders.

REFERENCES

[1] S.K. Pani, L.Panigrahy, V.H.Sankar, Bikram Keshari Ratha, A.K.Mandal, S.K.Padhi, "Web Usage Mining: A Survey on Pattern Extraction from Web Logs", *International Journal of Instrumentation, Control & Automation (IJICA), Volume 1, Issue 1, 2011*

[2] Yogish H K, Dr. G T Raju, Manjunath T N, "The Descriptive Study of Knowledge Discovery from Web Usage Mining", *IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 1, September 2011*

[3] Udayasri.B, Sushmitha.N, Padmavathi.S, "A LimeLight on the Emerging Trends of Web Mining" , *Special Issue of International Journal of Computer Science & Informatics (IJCSI), ISSN (PRINT):2231–5292,Vol.-II,Issue-1,2*

[4] Navin Kumar Tyagi, A.K. Solanki & Sanjay Tyagi. "An Algorithmic approach to data preprocessing in Web usage mining", *International Journal of Information Technology and Knowledge Management July-December 2010, Volume 2, No. 2, pp. 279-283*

[5] Surbhi Anand , Rinkle Rani Aggarwal, "An Efficient Algorithm for Data Cleaning of Log File using File Extensions", *International Journal of Computer Applications (0975 – 888),Volume 48– No.8, June 2012*

[6] J. Vellingiri and S. Chenthur Pandian, "A Novel Technique for Web Log mining with Better Data Cleaning and Transaction Identification", *Journal of Computer Science7 (5): 683-689,2011 ISSN 1549-3636 © 2011 Science Publications*

[7] Priyanka Patil and Ujwala Patil, " Preprocessing of web server log file for web mining", *National Conference on Emerging Trends in Computer Technology (NCETCT-2012)", April 21, 2012*

[8] Vijayashri Losarwar, Dr. Madhuri Joshi, "Data Preprocessing in Web Usage Mining", *International Conference on Artificial Intelligence and Embedded Systems (ICAIES'2012) July 15-16, 2012 Singapore*

[9] V.Chitraa , Dr.Antony Selvadoss Thanamani, "A Novel Technique for Sessions Identification in Web Usage Mining Preprocessing" , *International Journal of Computer Applications (0975 – 8887) Volume 34– No.9, November 2011*

[10] Spilipoulou M.and Mobasher B, Berendt B.,"A framework for the Evaluation of Session Reconstruction Heuristics in Web Usage Analysis", *INFORMS Journal on Computing Spring ,2003*

[11] Sachin yele, Beerendra Kumar, Nitin Namdev, Devilal Birla, Kamlesh Patidar.,"Web Usage Mining for Pattern Discovery", *International Journal of Advanced Engineering & Applications, January 2011.*

[12] Chetna Chand, Amit Thakkar, Amit Ganatra, "Sequential Pattern Mining: Survey and Current Research Challenges", *International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012*