CHAPTER 2

REVIEW OF LITERATURE

The WWW, while modernizing the mode of communication and interaction, also offers many new means of business-to-business and business-to-customer transactions, new mechanisms for person-to-person communication, new means of discovering and obtaining information, services and products electronically (Fesenmaler *et al.*, 2006). Both web content and web usage data are potential bearers of precious knowledge (Masand and Spiliopoulou, 2000). Web usage mining is an area of research that plays an important role in extracting this knowledge, and is basically based on methods that use user behaviour and the motivations for such behaviour (Clark *et al.*, 2006).

Currently, a lot of research projects deal with the Web usage mining and Web page recommendation for improving user experience from their navigational behavior to predict future visiting page. Almost all of these projects focus on extracting useful patterns using data mining techniques on web log files. The problem of investigation in the present research work is "*applying data mining techniques that are related to web data for the discovery of usage and navigation patterns in order to predict future page requests*". Mining for information or knowledge is a very important part of modern businesses. Mining large-scale transactional databases is considered a very important research subject for its obvious commercial potential (Berry and Linoff, 1997). It is also a major challenge due to its complexity.

New techniques have been developed over the last decade to solve clustering and classification (Duda *et al.*, 2000), prediction (Haykin 1999) and recommendation (Dai *et al.*, 2000; Krishnapuram *et al.*, 2001) problems in these kinds of databases. To find and evaluate valuable and meaningful patterns requires huge quantities of data, which can be easily maintained by recording all

customers' events into the database called data warehousing. Although large quantities of data are typically generated by dotcom web servers (Pitkow 1997; Lin *et al.*, 1999; Perkowitz and Etzioni., 2000a; Han *et al.*, 2000a; Facca and Lanzi, 2005; Shen *et al.*, 2007; Lee and Yen, 2008; Maruster *et al.*, 2008; Markov and Larose, 2007; Chen *et al.*, 2008; Li *et al.*, 2006; Zorrilla and Alvarez, 2008; Buzikashvili 2007), equal amount of data is also produced by e-commerce websites. This data is obtained by monitoring requested packages going to web site visitors, which are logged into a special purpose file called the web log files.

Web servers around the world generate thousands of giga-bytes of such data every day. According to the Internet Software Consortium, the World Wide Web (WWW) since 1994 has grown from two million servers to more that 110 million in 2001 (https://www.isc.org). The number of home users has increased from 3 million to more than 89 million for the same period, estimated by another Internet research company - Jupiter MM (Schmitt *et al.*, 1999) reported that 84% of interviewed companies received demand for site data to skyrocket by 2001. The online business retail spending has grown from $20.3 billion to $144 billion by 2003.

According to the EIAA Mediascope Europe 2008 study (http://www.eiaa.net), people are deepening their experience of the internet by not only increasingly using it for leisure, but actively enhance and manage their daily lifestyle. 179 million Europeans (60%) are online each week. Over half (55%) of users are online every single day. Three quarters (75%) of internet users are online during their evenings compared to 67% in 2007. 51% of use the internet at the weekend, an increase of 13% since 2007. Because of the growing confidence, consumers made a huge number of purchases online in 2008 or 9.2 purchases per person as against 7.7% in 2007.

By having these figures, it can be concluded that the market will successfully withstand only those companies that take significant attention into their web data and make serious analysis, and others will be pushed out of business (Ansari *et al.*, 2001). Thus, web log data is the largest source of information concerning human interaction with the www and continue to grow rapidly. This enables knowledge discovery from web logs to recognize practical users, improve marketing strategies and increase web site's retention etc.

Currently, most web sites are designed without taking into account how web logs can be used to tune and evaluate the usefulness of the web site. The success of the web site cannot be measured only by hits and page views (Schmitt *et al.*, 1999). Unfortunately, web site designers and web log analysers face problems such as identification of unique user's (Pitkow 1997), construction of discrete users' sessions and collection of essential web pages (Faulstich *et al.*, 1999) for analysis. Several web log mining tools have been developed and widely exploited to solve these problems. However, neither commercial tools (AccrueHitList, DataMiningSuite, MINEit, SAS_Webhound, WUM) nor the free tools (Analog; WUM; Pitkow *et al.*, 1994a; Han *et al.*, 1997) solve these problems adequately.

Several steps called knowledge discovery must be passed through in order to observe and analyze patterns from data. These steps are (a) Data pre-processing which includes such stages as data cleaning, feature selection, transformation, (b) Pattern discovery and (c) Finally results visualization and examination (Fayyad, 1996). This chapter presents the various approaches used by researchers for web usage mining with emphasis on next page prediction applications.

2.1. WEB USAGE MINING TECHNIQUES

Web mining has been successfully applied to various applications areas including researches in trend identification (Tho *et al.*, 2003), robot detection and

filtering to separate human and non-human behaviour (Kohavi, 2001), user profiling (Masand *et al.*, 2002), fraud and threat analysis (Lazarevic *et al.*, 2002) and identifying web communities (Gibson *et al.*, 1998). Several tools which analyze web log data to provide knowledge that will help web administrators in building effective websites have been developed. These tools help to understand the web usage process, web structure or web content from the web data.

Chi *et al.* (1998) developed a Web Ecology and Evolution Visualization (WEEV) tool to understand the relationship between Web content, Web structure and Web Usage over a period of time. The site hierarchy is represented in a circular form called the "Disk Tree" and the evolution of the Web is viewed as a "Time Tube".

Hsien *et al.* (2005) described a method of web usage mining approach to discover patterns in the navigation of websites known as Unexpected Browsing Behaviours (UBBs) and called their technique as UBB mining. Web designers can review these UBBs to improve their website. Their results proved that the combination of predefined routing algorithm and UBB mining algorithm can discover interesting browsing patterns.

The most widely used approach is web usage mining that entails many models like Markov models, Association rules and clustering (Srivastava *et al.*, 2000). However, there are some challenges with the current state of the art solutions when it comes to accuracy, coverage and performance. A Markov model is a popular approach to predict what pages are likely to be accessed next (Cadez *et al.*, 2003; Khalil *et al.*, 2007; Deshpande and Karypis, 2004). A problem that faces Markov models is the difficulty in identifying the optimal number of Markov model orders which affects the system accuracy, coverage and performance.

The second is Association rules (Agrawal *et al.*, 1993). It is based on the relationship of co-occurrence of pages without considering the sequence of them.

This makes Association rules generally produce low precision, but high recall in the prediction (Kim *et al.*, 2005). Yang *et al.* (2003a) studied five different representations of association rules, namely, Subset rules, Subsequence rules, Latest subsequence rules, Substring rules and Latest substring rules.

Using the success results of Markov model and association rule mining, Khalil *et al.* (2006) proposed the combination of association rules and Markov model system. They used lower order all k-th Markov models to predict the next page to be accessed. In ambiguous predictions, association rules are used to compliment Markov models. The advantage of this combination is that when Markov models are unable to make the prediction, association rules look further back at the previously visited pages and leads to the most appropriate page for prediction. The main problem is dependent on the length of the web user session and it is difficult to perform the analysis with short user sessions.

Later, Khalil *et al.* (2008) introduced the Integration Prediction Model (IPM) by combining Markov model, Association rules and clustering algorithm together. Here, the prediction is performed on the cluster sets rather than the actual sessions. This method was adopted by several researchers (Vakali *et al.*, 2004; Pallis *et al.*, 2007; Borges and Levene, 2005; Lu *et al.*, 2005a).

Salin and Senkul, (2009) provided a structure for combining semantic information with Web usage mining. The common navigational patterns are obtained as the form of ontology instances rather than Web page addresses and the outcome is used for generating Web page recommendations to the visitor. Additionally, an assessment method is implemented with the intention of testing the accomplishment of the recommendation. Test result confirms that precise recommendations can be achieved by including semantic information in the Web usage mining.

Han *et al.* (2008) investigated Web Mining Algorithms based on Usage Mining. It provides the design approach of an electronic commerce website. This approach is easy, efficient and easy to understand, it is appropriate to the Web usage mining demand of building a low cost B2C website.

Web Usage Mining is the most sought after tool in the Internet community where data from online web is converted to meaningful knowledge. The knowledge thus discovered can be used in web personalization, general system improvement, improve business intelligence, site modification and discover usage characteristics.

2.1.1. Web personalization

Web personalization is the process of customizing a web site to the needs of specific users, taking advantage of the knowledge acquired from the analysis of the user's navigational behavior (usage data) in correlation with other information collected in the Web context, namely, structure, content and user profile data. In general, personalization techniques are divided into offline and online techniques. Offline personalization is based on simple user profiling and manual decision rule systems. Web usage mining is an online personalization data source. By evaluating site behavior and usage, a view of the website user is gained which yields to more effective personalization strategies. User profiles are an important source of data for data personalization. Due to the explosive growth of the Web, the domain of Web personalization has gained great momentum both in the research and commercial areas (Cingil *et al.*, 2000).

Mobasher *et al.* (1999) proposed an effective technique for capturing user profiles based on association rules discovery and usage based clustering combined with current status of an on-going activity to perform real time personalization. This was followed by the work of Toolan and Kushmerick (2002) who proposed

techniques based on web usage mining to deliver Personalized Site Maps that are specialized to the interest of each individual visitor.

Shahabi and Kashani (2003) described a complete framework for web-usage mining to satisfy the challenging requirements of web-personalization applications. They introduced a distributed user-tracking approach for accurate, scalable and implicit collection of the usage data and proposed a feature-matrices (FM) model, to discover and interpret users' access patterns. A novel similarity measure, based on FM, was designed for accurate classification of partial navigation patterns in real time. This system worked well with both synthetic and real data for anonymous and efficient web personalization. It was at this period that web usage mining bloomed and a review of the popular techniques and tools available for web personalization was provided by Eirinaki and Vazirgiannis (2003).

Recently, Baraglia and Silvestri (2007) proposed a dynamic personalization system which would personalize a site without the intervention of web users, using the information collected from log files and navigation pattern.

During the same period, Ouamani *et al.* (2007) designed a web usage mining architecture for web personalization (PWUM) which was implemented using a multi-agent platform. This latter is composed of a set of autonomous agents interacting together in order to fulfill the main goal of the system. Agents are divided into modules that have well-defined tasks and that are further divided into two working groups, offline and online. The personalization agent uses the user model knowledge, along with the previously discovered sequential patterns, and applies a set of personalization rules, in order to deliver the personalization tasks or functions like the memorization of personal information, user salutation, recommendation of links related to what users in the same group previously

choose, or links that the same user usually views, objects differentiation by presenting different features of each object.

2.1.2. Site Improvement

Apart from developing web mining tools, work on general improvement of knowledge extraction from web data has also been proposed. Performance and other service quality attributes are crucial to user satisfaction from services such as databases, networks, etc. Similar qualities are expected from the users of Web services. Web usage mining provides the key to understanding Web traffic behavior, which can in turn be used for developing policies for Web caching, network transmission, load balancing, or data distribution (Cohen *et al.*, 1998). Security is an acutely growing concern for Web-based services, especially as electronic commerce continues to grow at an exponential rate (Fawcett and Provost, 1999). Web usage mining can also provide patterns which are useful for detecting intrusion, fraud, attempted break-ins, etc.

Almeida *et al.* (1996) propose models for predicting the locality, both temporal as well as spatial, amongst Web pages requested from a particular user or a group of users accessing from the same proxy server. The locality measure can then be used for deciding pre-fetching and caching strategies for the proxy server. The increasing use of dynamic content has reduced the benefits of caching at both the client and server level. Schechter *et al.* (1998) has developed algorithms for creating path profiles from data contained in server logs. These profiles are then used to pregenerate dynamic Homepages based on the current user profile in order to reduce latency due to Page generation.

Cooley *et al.* (1999) in his investigation has successfully differentiated “web content mining” and “web usage mining”. According to this investigation, both the areas overlap, depending on whether the data used in the knowledge discovery process. During 1997, Cooley *et al.* presented a paper on the discovery

and application of interesting patterns from web data using web mining techniques. This was followed by another work from the same team (Srivastava *et al.*, 2000) where they presented different methods for identifying and discovering usage patterns from web data.

This was followed by Seo *et al.* (2001) method of building intelligent systems for mining information and extraction rules from semi-structured Web pages by using domain knowledge. At the same period, Kohavi *et al.* (2001a) presented a method to mine log data across all customer touch points to extract web knowledge. Similarly, the same authors (Kohavi *et al.*, 2001b) also tested their system on e-commerce sites and identified the challenges and issues in it. Later in 2002, Srivastava *et al.* proposed hyperlink analysis technique and applied this technique to various applications and proved that their technique is superior to the existing methods. Madria *et al.* (1999) presented the issues involving web data mining, and a comprehensive survey of various techniques is presented by Kosala and Blockeel (2000).

2.1.3. Business Intelligence

Information on how customers use a web site is critical information for marketers of e-business. Buchner and Mulvenna, (1998) have presented a knowledge discovery process in order to discover marketing intelligence from Web data. They defined a Web log data hypercube that will consolidate Web usage data along with marketing data for e-commerce applications. They identified four distinct steps in customer relationship life cycle that can be supported by their knowledge discovery techniques: customer attraction, customer retention, cross sales and customer departure.

There are several commercial products, such as Surf Aid, Accrue, Net-Genesis, Aria, Hotlist, and Web Trends that provide Web traffic analysis mainly for the purpose of gathering business intelligence. Accrue, Net Genesis and Aria

axe designed to analyze ecommerce events such as products bought and advertisement click-through rates in addition to straight forward usage statistics. Accrue provides apathy analysis visualization tool and IBM's Surf Aid provides LAP through a data cube and clustering of users in addition to page view statistics. Padmanabhan and Tuzhilin (1998) use Web server logs to generate beliefs about the access patterns of Web pages at a given Web site. Algorithms for finding interesting rules based on the unexpectedness of the rule were also developed.

2.1.4. Site Modification

The attractiveness of a web site, in terms of both content and structure, is crucial to many applications, like a product catalog for e-commerce. Web usage mining provides a detailed feedback on user behavior, providing the Web site designer information on which to base redesign decisions. While the results of any of the projects could lead to redesigning the structure and content of a site, the adaptive Web site project (SCML algorithm) (Perkowitz and Etzioni, 1998) focuses on automatically changing the structure of a site based on usage patterns discovered from server logs. Clustering of pages issued to determine which pages should be directly linked.

2.1.5. Usage Characteristics

While most projects work on characterizing the usage, content, and structure of the Web, there is large amount of overlap between Web characterization research and web usage mining. Catledge and Pitkow (1995) discussed the results of a study conducted at the Georgia Institute of Technology, in which the Web browser Mosaic was modified to log client side activity. The results collected provide detailed information on the user's interaction with the browser interface as well as the navigational strategy used go browse a particular site. The project also provides detailed statistics on occurrence of the various client side events such as the clicking the back/forward buttons, saving a file, adding to bookmarks, etc.

Huberman *et al.* (1998) proposed a model which can be used to predict the probability distribution of various pages a user might visit on a given site. This model works by assigning a value to all the pages on a site, based on various attributes of that page. The formulas and threshold values used in the model are derived from an extensive empirical study, carried out on various browsing communities and their browsing patterns.

Arlitt and Williamson (1997) discussed various performance metrics for Web servers along with details about the relationship between each of these metrics for different workloads. Manley (1997) developed a technique for generating a custom made benchmark for a given site based on its current workload. This benchmark, which he calls a self-configuring benchmark, can be used to perform scalability and load balancing studies on a Web server. Chi *et al.* (1998) describe a system called WEEV (Web Ecologyand Evolution Visualization) which is a visualization tool to study the evolving relationship of web usage, content landsite topology with respect to time.

In order to offer the online prediction efficiently, Shinde and Kulkarni (2008) formulated a architecture for online recommendation for predicting in Web Usage Mining System. This approach provides the structural design of on-line recommendation system in Web usage mining (OLRWMS) for enhancing the exactness of classification by dealing between classifications, estimation and provides user activities and user profile in online phase of this architecture.

Bin *et al.* (2009) have applied negative association rules approach to Web usage mining. In the course of the research the authors have proved that the negative association rules have an additional significant role on access pattern to Web visitors, provide the mining algorithms, to resolve the deficiencies in which positive association rules are referred.

2.2. LOG FILE ANALYSIS FOR WEB USAGE MINING

Mining the user click-stream for user behavior and using it to adapt the 'look-and-feel' of a site to a reader's needs was first proposed by Perkowitz and Etzioni (1998).

Analysis of Web server transaction logs provides comprehensive information on Web server traffic (Toolan *et al.*, 2003). Knowledge of server traffic can provide information on who is accessing a current Web site, what site they are coming from and when and where they are visiting. This type of data can be beneficial in assessing what pages on the site receive the most frequent traffic and who is using them. This can help the design team to further identify target user groups for their current site or for redesigns (Blackett *et al.*, 2003).

Log analysis is typically conducted as an automated procedure with log analyzer software. During log analysis, all Web server activity is recorded. This includes data such as the IP address and/or domain of the individual requesting a Web page from the server, the date and time the request was made, the filename of the page accessed and the number of bytes of data served (Peng *et al.*, 2005). This section reviews some work done in this area.

A number of articles have discussed Web server log analysis for libraries, since libraries began to develop Web presences (Li, 1991; Stabin and Owen, 1997; Nicholas *et al.*, 2000). These articles describe summary level metrics of Website usage, such as the total number of user sessions, broken down by variables such as date, time or host domain of the requestor. As noted by many of these authors and by Goldberg (2003), these studies have been constrained by two main factors. One, data provided by the hypertext transfer protocol (HTTP) that governs user transactions on the Web is very limited. Second, usage logs are designed for use by system administrators, not for tracking users. While the information available in logs is limited, some user and resource usage data can be gleaned from them.

For example, the Internet protocol (IP) or network address of users can provide some insight into who is using a site and the requested file can be used to make some conclusions about what content is being used.

Statistical analysis applies a variety of techniques including ordinary least squares (OLS) and logistic regression, cluster analysis, decision trees and neural networks. Web usage mining often analyzes sequences of page accesses to provide personalization and targeted marketing (Dunham, 2003; Han and Kamber, 2001). Feng and Murtagh (2000) provide an example of using Web usage mining techniques to develop a personalization system. Davis (2004) analyzes the information-seeking behavior of chemists based on Web log analysis.

A semantic session analysis model partitioning Web usage logs was presented by (Zhou *et al.*, 2006a). The model enhances usage logs with semantic using Markov chain model based on ontology semantic measurement. The competitive method is applied to determine the end of the sessions. Compared with other algorithms, more successful sessions are additionally detected by semantic outlier analysis. Tanasa and Trousse (2004) preprocessed web log files to reduce their size. They also used data summarization techniques to increase the quality of data obtained after classical preprocessing.

Cooley *et al.* (1997), presented methods for user identification, sessionizing (i.e. constructing or reconstructing sessions) and page view identification. In another work, Murata and Saito (2006), highlight the importance of analyzing users web log data and extracting their interests of web-watching behaviors and describes a method for clarifying users interests based on an analysis of the site-keyword graph.

Singh *et al.* (1998) and Pabarskait (2003) discussed the importance of decision trees in web log mining. The authors suggested several hypotheses to improve web sites retention and showed that it is possible to predict future user

actions with reasonable misclassification error as well as to find combinations of sequential pages resulting in browsing termination.

Al-Khajri *et al.* (2005) discussed the importance of navigation patterns of users. It provides information about the major aspects and problems related to the task of modeling the user behavior. It also provides information on recent developments related to automatic web navigation, implicit capturing of user interests and future directions on web log analysis.

2.3. WEB PAGE RECOMMENDATION

Numerous researches are available in the literature for web recommendation system using sequential pattern mining and this section presents some of them. Zhou *et al.* (2004) proposed an intelligent Web recommender system identified as SWARS (Sequential Web Access-based Recommender System) that employed sequential access pattern mining. In the proposed system, CS-mine, an efficient sequential pattern mining algorithm was used to recognize frequent sequential Web access patterns. The access patterns were then stored in a compact tree structure (Pattern-tree) which was then employed for matching and generating Web pages for recommendations. The performance of the proposed system was analyzed on the basis of precision, satisfaction and applicability.

An efficient sequential access pattern mining algorithm, called CSB-mine (Conditional Sequence Base mining algorithm) was presented by Zhou *et al.,* 2006b). The presented CSB-mine algorithm was on the basis of conditional sequence bases of each frequent event which removes the need for constructing WAP-trees. This enhanced the efficiency of the mining process considerably in comparison to WAP tree-based mining algorithms, particularly when the value of support threshold becomes smaller and the database size gets larger. The authors have also described a sequential access-based web recommender system that has included the CSB-mine algorithm for web recommendations.

Wei *et al.* (2009) presented a hybrid web personalization system that was based on clustering and contiguous sequential patterns. This system clustered log files to find out the basic architecture of websites and for each cluster, employed contiguous sequential pattern mining to optimize the topologies of websites further.

Yang *et al.* (2006) proposed an efficient sequential mining algorithm called LAPIN_WEB (LAst Position INduction for WEB log), which is an extension of the previous LAPIN algorithm to extract user access patterns from traversal path in Web logs. Web log mining system comprises of data preprocessing, sequential pattern mining and visualization. The experimental results and performance studies established that LAPIN_WEB was efficient and outplayed familiar PrefixSpan by up to an order of magnitude on real Web log datasets. Furthermore, they also implemented a visualization tool to aid interpret mining results and also forecast users' future requests.

Huang *et al.* (2006) introduced an approach that extended frequent sequence that used closed itemsets instead of single items. The closed sequential patterns were made up of only closed itemsets. Therefore, needless item extensions which generate non-closed sequential patterns were prevented. Experimental results proved that the proposed approach was two orders of magnitude faster than the existing related works with a reasonable memory cost.

An efficient algorithm, known as TSP (Top-k closed Sequential Patterns), was developed by Tzvetkov *et al.* (2005) for mining closed patterns without min_support. Starting at (absolute) min_support=1, the algorithm used the length constraint and the properties of top-k closed sequential patterns to execute dynamic support raising and projected database pruning. The performance study illustrated that TSP has high performance. It outplayed the efficient closed

sequential pattern-mining algorithm, CloSpan, even when the latter was processed with the best tuned min_support threshold.

Uno *et al.* (2004) presented an efficient algorithm called LCM (Linear time Closed pattern Miner) for mining frequent closed patterns from large transaction databases. The major contribution towards the theoretical part was the proposed prefix-preserving closure extension of closed patterns, which allowed them to look for all frequent closed patterns in a depth-first manner, in linear time for the number of frequent closed patterns. Their algorithms do not require any storage space for the earlier obtained patterns, while the existing algorithms require it. Performance analysis of LCM with straightforward algorithms illustrated the positive aspects of prefix preserving closure extension.

Lin *et al.* (2008) proposed an algorithm for mining closed frequent sequences. The proposed system was scalable, condensed and lossless structure of complete frequent sequences that was mined from a sequence database. The algorithm, FMCSP (Fast Mining of Closed Sequential Patterns), has employed a number of optimization methods, namely equivalence class, to lessen the needs of searching space and run time. Specifically, one of the main problems in this type of algorithms was the redundant generation of the closed sequences; therefore, they presented an efficient and memory saving methods, diverse from existing works, that does not require the complete set of closed sequences to be residing in the memory.

One important part of web recommendation systems is web page prediction, which is used to prefigure the next page a user might visit. The various techniques proposed are reviewed in the following section.

2.4. WEB PAGE PREDICTION TECHNIQUES

This section presents the studies related to the techniques used for pattern discovery and analysis that can be used to predict next web page for a user.

2.4.1. Preprocessing

Preprocessing is a step in web usage mining which is composed of many steps like data cleaning, user identification and session identification. Several authors have used different techniques to transform the raw web log data into a form that improves the subsequent steps of web usage mining. For example, Etminani *et al.* (2009) applied Kohonen's SOM (Self Organizing Map) to pre-process Web logs of Web server logs to extract patterns. An analysis of the existing methods for preprocessing and how to enhance them for pattern mining and analysis was presented by Hussain *et al.* (2010b). In preprocessing, two areas are considered challenging and have gained considerable interest from the researchers. They are session identification and path completion.

- **User Session Identification**

A user session can be defined as a set of pages visited by the same user within the duration of one particular visit to a web site (Chitraa and Davamani, 2010). A user may be in a single or multiple sessions during a period. Once a user was identified, the click stream of each user can be portioned into logical clusters. The method of portioning into sessions is called as Sessionization or Session Reconstruction. A transaction is defined as a subset of user session having homogenous pages. The methods of sessionization are categorized into three methods, two of which are time oriented and the third is based on navigation details of a web topology.

Time oriented sessionization methods are considered to be the simplest and are based on total session time and on single page stay time. The total session time method uses page viewing time during sessionization. Page viewing time is defined as time taken by a specific user to visit a set of pages. This time is kept as a fixed constant and may vary from 25.5 minutes (Catlegde and Pitkow, 1995) to 24 hours (Spilipoulou *et al.*, 2003), while 30 minutes is the default timeout by

Cooley (Cooley *et al.*, 1997). The second method depends on page stay time which is calculated as the difference between two timestamps. If it exceeds 10 minutes, then it is assumed as a new session. Time based methods are not reliable because users may involve in some other activities after opening the web page and factors such as busy communication line, loading time of components in web page, content size of web pages are not considered. But it has the most important advantage of simple implementation procedure during web using mining process and are very fast in sessionization. Time complex algorithms like clustering and classification normally prefer time based sessionization.

Navigation-Oriented Heuristics uses web topology in graph format by considering the webpage connectivity between two consecutive page requests. If a web page is not connected with previously visited page in a session, then it is considered as a different session. Cooley *et al.* (1997) proposed a referrer based heuristics on the basis of navigation in which referrer URL of a page should exist in the same session. If no referrer is found, then it is considered as a new session.

Both, time-oriented and navigation-oriented heuristic methods have been used in many applications. The referrer-based method and time-oriented heuristics method are combined to accomplish user session identification in Domenech and Lorenzo (2007). A simple algorithm was devised by Zhou *et al.* (2006a) where an access session is created as a pair of URL and the requested time in a sequence of requests with a timestamp. The default time set by author is 30 minutes per session.

Smart Miner, a novel framework devised by Bayir *et al.* (2008, 2009), is a part of Web Analytics Software. The sessions constructed by SMART-SRA contains sequential pages accessed from server-side works in two stages and follows Timestamp Ordering Rule and Topology rule. In the first stage, the data stream is divided into shorter page sequences called candidate sessions, by using

session duration time and page stay time rules. Time constraint is also considered as the difference between two consecutive pages is smaller than 10 minutes.

Another method using Integer Programming was proposed by Dell *et al.* (2008). The advantage of this method is that it constructs all user sessions simultaneously. In his method, each web log entry was considered as a register and registers from the same IP address and agent were grouped to form a session. A binary variable was used to assign a value of 1 or 0 to indicate whether register is assigned a position in a particular session or not.

Another important contribution to session identification is through the use of graphs. Graphs provide more accurate results for session identification where web pages are represented as vertices, and hyperlinks are represented as edges in a graph. User navigations are modeled as traversals from which frequent patterns can be discovered, i.e., the subtraversals that are contained in a large ratio of traversals (Lee and Park, 2007).

A method proposed by Heydari *et al.* (2009) considered client side data while reconstructing user's session. There were three phases in this method. In the first phase, an interface to monitor user's browsing behavior was designed and events such as session start, end, on page request, on page load, on page focus were recorded in session. In the second phase, a base graph was constructed using web usage data. Browsing time of web pages was indicated as vertices. A database was created with traversals, which is a sequence of consecutive web pages on a base graph. In the third phase, three graph mining method is applied to the database to discover weighted frequent pattern. Weighted frequent pattern is the pattern when weight of traversal is greater than or equal to a given Minimum Browsing Time.

Chen and Liu (2006) proposed a model in which data cleaning and session identification were combined. In this method, the user activity records were

checked for spider record and embedded objects in pages, and were removed. The resultant record is searched in the session record and it is not present, then a new session is established. If the present session ends or exceeds the preset time threshold, the pattern discovery will end and a new session starts. Graph mining methods constructs accurate sessions.

The main disadvantages of navigation oriented solutions are that, even though, they produce accurate results, have implementation and computation complexities, which make then unsuitable for user navigation pattern discovery applications. In these types of applications, results like prediction or recommendation has to be made in a quick and efficient manner, which are provided well by time oriented heuristics.

Losarwar and Joshi (2012) proposed a new method for data cleaning, user identification and session identification using the fields of common log files. The authors assumed that each combination of IPaddress/Agent/Operating system as a single user and if there is a new user there is new session or in one user session, if the referrer page is null, there is a new session or if the time between page requests exceeds a certain limit (30 minutes) It is assumed that user as a new session.

Nasraoui *et al.* (2008) presented a comprehensive structure and findings in mining Web usage patterns by using Web log files of a real Web site that has all the demanding aspects of real-life Web usage mining, together with evolving user profiles and external data describing an ontology of the Web content. Therefore, the authors present a technique for determining and tracing the mounting user profiles. The authors also discuss about how the obtained users profiles can be improved with clear information obtained from search queries of Web log data. Profiles are also enhanced with additional domain-specific information aspects that provide a panoramic view of the discovered mass usage modes. Many

experiments have been done by the authors to assess the excellence of the mined profiles, especially their adaptability in the face of developing user behaviour.

Fang *et al.* (2009) proposed a double algorithm of Web usage mining in accordance with the sequence number that is appropriate for mining several session patterns. The algorithm transforms session pattern of particular user into binary and subsequently uses up and down search approach to double generate candidate frequent itemsets. The algorithm works out support by sequence number dimension with the intention of scanning once session pattern of a particular user, which is dissimilar from conventional double search mining algorithm. In addition to this, the effectiveness of Web usage mining is competently enhanced because of this approach.

- **Path completion**

In web log files, often there will be entries which have missing pages due to proxy servers and caching problems (Li *et al.*, 2008; Li and Feng, 2009). These pages have to be omitted for effective web usage analysis. A normal process begins by checking whether there is a link to the previous page. If no such link is available, then the recent history is analyzed. An entry in the historical log data indicates that the user has pressed the browser's back button to reach a page and the historical data can be analyzed in a repetitive fashion till the main page is reached, which can then be added to the log file, leaving all other pages. The previous page history is normally provided by the referrer entry of a web log data. In cases, where the referrer log is not clear, the site topology can be used for the same effect. If many pages are linked to the requested page, the closest page is the source of new request and is added to the session. To perform such operations, three approaches exist. They are, (i) Reference Length Approach, (ii) Maximal Forward Reference Approach, and (iii) Time Window Approach.

(i) Reference Length approach

This approach is based on the assumption that the amount of time a user spends on a page correlates to whether the page is an auxiliary page or content page for that user. It is expected that the time spent on auxiliary page is small and content page is more. A reference length can be calculated to estimate the cut off between auxiliary and content references. The length of each reference is estimated by taking the difference between the time of the next reference and the current reference. But the last reference has no next reference. So this approach assumes the last one is always an auxiliary reference.

(ii) Maximal Forward Reference

A transaction is considered as the set of pages from the visited page until there is a backward reference. Forward reference pages are considered as content pages and the path is taken as index pages. A new transaction is considered when a backward reference is made.

(iii) Time Window

A time window transaction is framed from triplets of IP address, user identification and time length of each web page up to a limit called time window. If time window is large, each transaction will contain all the page references for each user.

Several researchers have used the above methods during preprocessing of web log files. An optimal algorithm was devised by Arumugam and Suguna (2009) to generate accurate path sequences using two way hashed structure based on access history list to frame a complete path. A tree structure was used to store the server page data. During path completion, the tree was traversed and unused pages were identified using backward reference process. The main disadvantage of this method was that it was time consuming.

To overcome this, the same authors proposed another Session Identification algorithm based on data structures such as Array List to represent Web Logs and User Access List, a Hash table to represent server pages, a two-way hashed structure instead of a tree structure, in the same paper. To solve the time consumption only visited pages were stored in access history list and the rest were ignored or not considered. Using a single search in history list, the page sequences could be directly located. When pages are referred from other servers, directly start from the current page and not from root. If the page is not available in present sessions, a new session is started. This method generates correct complete path than maximal forward and reference length methods.

The demerits of the above algorithms were that they were tested only with a very small log file and only the standard log file format with six fields were used. However, the reference length approach and time stamp approach are more beneficial in terms of complexity and scalability to any size of log file.

2.4.2. Pattern Discovery and Analysis Techniques

Predicting a user's next access on a web site has attracted a lot of research work lately due to the positive impact of such prediction on different areas of web based applications (Khalil *et al.*, 2007). In all of these applications the goal is the development of an effective and accurate prediction model. The most successful prediction algorithms use historical access data from web access logs, which records the information about all the visits by different users to different web sites and web pages. By having this information, many researchers have designed action systems that use the predictions from a learned model and have developed methods for dealing with specific aspects of web usage mining, like automatically discovering web personalization (Nasraoui and Petenes, 2003), recommender systems (Khalil *et al.*, 2008), web prefetching (Yang *et al.*, 2003b; Alexandros *et al.*, 2003), web presenting (Li, 2001), and design of adaptive web sites (Zhu *et al.*, 2002).

Pattern discovery from web data is the key component of web mining and it converges algorithms and techniques from several research areas. Catledge and Pitkow (1995) say that the Web is a kind of open, highly dynamic and collaborative hypermedia system, a "dynamic information ecology" including two main types of user strategies: search and navigation. Cove and Walsh (1988) add a third strategy, "serendipitous browsing", when the users randomly walks through Web pages. These strategies are not excluding, one user shifting its focus between them. Web designers must be aware of these strategies when planning a Web site, since there are different needs associated to each one. The risk of users becoming "lost in cyberspace" (Nielsen, 1990), when these needs are insufficiently mitigated also exist.

According to Kimble and Kudenko (2007), abundant information can be uncovered, if they are properly analyzed. Recently, several Web Usage Mining (WUM) systems have been proposed to predicting user's behaviour, preferences and their navigation behaviors. Techniques that have been successfully exploited in pattern discovery fall under statistical pattern mining, association rules mining, clustering and classification.

- **Statistical Pattern Mining**

Statistical techniques are the most powerful tools in extracting knowledge about visitors to a Web site. The analysts may perform different kinds of descriptive statistical analyses based on different variables when analyzing the session file. By analyzing the statistical information contained in the periodic Web system report, the extracted report 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 models for pattern matching and knowledge discovery is a technique that has been by many researchers. Borges and Levene (1998 and

Levene and Loizou (1999) used statistical models to represent user navigation. In their works, WWW is considered to be a database of pages, described as a directed graph whose nodes are pages and the arcs are hyperlinks between pages. Through association of states to pages and probabilities to links, one can build Markov chain models to represent the navigation process, since this process has a strong regularity from a statistical point of view. A survey of various techniques on machine learning and statistical pattern mining for analyzing hypertext is provided by Chakrabarti *et al.* (1998).

Spiliopoulou *et al.* (1999) proposed the exploitation of mining technology to discover access patterns with "interesting" statistical properties and presented Web Utilization Miner (WUM) – a tool designed for the purpose. The mining model of WUM is in two aspects. First, it predicts that the "importance" indicators in user behavior go far beyond than frequent access to some pages, such that the pattern discovery can be done in the statistical domain, but also supports the subjective specification. Second, by processing aggregated sequences and applying optimization steps during the mining process, the high performance can be achieved.

Ali and Ghorbani (2004) described an improved statistical-based time oriented heuristics for the reconstruction of user sessions from a server log. Jain *et al.* (1996) presents transaction data models for various web mining tasks such as the discovery of association rules and sequential patterns from the Web data.

Mobasher and Moore (1998) proposed a framework for web mining using various web mining task and implemented a prototype namely WEBMINER by applying the framework as proposed. The problem has already pointed out from the system is to perform cluster analysis on association rules and sequential pattern discovery. Recently, AlMurtadha *et al.* (2010) used statistical techniques for mining web navigation profiles for recommendation system.

- **Classification and Clustering Techniques**

Classification is the technique to map a data item into one of several predefined classes. In the Web domain, Web master or marketer use this technique to establish a profile of users belonging to a particular class or category. This requires extraction and selection of features that best describe the properties of a given class or category. The classification can be done by using supervised inductive learning algorithms such as decision tree classifiers, naïve Bayesian classifiers, k-nearest neighbor classifier, Support Vector Machines, etc. **Clustering analysis** is a technique to group together users or data items (pages) with the similar characteristics. Clustering of user information or pages can facilitate the development and execution of future marketing strategies. Clustering of users will help to discover the group of users, who have similar navigation pattern. This section presents a literature review regarding both these fields.

Clustering assumes consolidating perspective customers/visitors containing similar behaviour and characteristics naturally into groups (Jain *et al.*, 1997; Xiao and Zhang, 2001; Kanth Ravi and Ravada, 2002). These groups of related clients are called clusters. For example, a cluster is found of the most valuable customers. It contains attributes of clients having income more than £25,000 per year and age between 30 and 45. Another example of the cluster is web pages occurrences. For example, pages A, B, and C are the most visited from Mondays to Fridays. The similarity between cluster objects (data samples) can be measured utilizing Euclidean distance (Duda *et al.*, 2000).

Clustering is suitable for huge datasets and is the first data mining/web mining technique to examine the structure and patterns in data (Hand *et al.*, 2001). Dai *et al.* (2000) put forward techniques based on clustering users' transactions to discover overlapping profiles which are later used by recommendation systems for real-time personalization.

Liu and Keselj (2007) proposed the automatic classification of web user navigation patterns and proposed a novel approach to classifying user navigation patterns and predicting users' future requests. The approach is based on the combined mining of Web server logs and the contents of the retrieved web pages. They used character N-grams to represent the contents of web pages and combined them with user navigation patterns by building user navigation profiles composed of a collection of N-grams

Later, Cadez *et al.* (2000) present a tool called WebCANVAS that displays clusters of users with similar navigation behavior. Prasetyo *et al.* (2002) introduce "Naviz", an interactive web log visualization tool that is designed to display the user browsing pattern on the web site at a global level and then display each browsing path on the pattern displayed earlier in an incremental manner. The support of each traversal is represented by the thickness of the edge between the pages. Such a tool is very useful in analyzing user behavior and improving web sites. Khosravi and Tarokh, (2010) proposed a technique based on naive Bayesian method for modeling and predicting users' navigation behavior.

Gangrade *et al.* (2009) discussed about the techniques for privacy preserving classification under multi-party environment. Further, the two approaches, the classification model and secure multi-party computation algorithms have also been reviewed. The performance analysis of the algorithms has been concentrated in connection with the classification.

Baraglia and Palmerini (2002) proposed a WUM system called SUGGEST that provides useful information to make easier the web user navigation and to optimize the web server performance. The main goal of SUGGEST is to find useful information from the user access data collected in web server logs. SUGGEST adopts a two levels architecture composed by an offline creation of historical knowledge and an online engine that understands user's behavior. After

a pre-processing of the data recorded in the web server log files, SUGGEST creates clusters of related pages based on users past activity and then classifies new users by comparing pages in their active sessions with pages inside the clusters created. A set of suggestions is then obtained for each request. The main disadvantages of this system are: Online component and offline component work separately, how to maintain and update the knowledge extracted in the offline phase and how the system can exactly understand the differences between index page and content page.

In the new architecture of SUGGEST they put together the previous two components into a single online module performing the same operation (Silvestri *et al.*, 2004; Baraglia and Silvestri, 2004). As the requests arrive at this system module it incrementally updates a graph representation of the Web site based on the active user sessions and classifies the active session using a graph partitioning algorithm. This architecture was designed to be usable on Web sites made up of pages statically generated, i.e. Web sites with a fixed number of pages. A list containing all the information describing a Web site pages was required as input by this architecture at its start-up time. Potential limitation of this architecture might be:

a) The memory required to store Web server pages is quadratic in the number of pages. This might be a severe limitation in large sites made up of millions of pages, and
b) It does not permit us to manage Web sites made up of pages dynamically generated.

The last contribution of SUGGEST architecture is proposed by Baraglia and Palmerini (2002). This version of SUGGEST introduces a novel solution to implement WP (Web Personalization) as a single online module that performs user profiling, model updating and recommendation building. It is designed to

dynamically generate personalized contents of potential interest for users of large Web sites made up of pages dynamically generated. It is based on an incremental personalization procedure tightly coupled with the Web server. It is able to update incrementally and automatically the knowledge base obtained from historical usage data and to dynamically generate a list of page links (suggestions). The suggestions are used to personalize the HTML page requested on-the-fly. The adoption of a LRU-based (Least Recently Used) algorithm handling the knowledge base makes it possible for SUGGEST to manage large Web sites. But in this system quality of recommendations is not better than previous version of this system.

Another study towards Web Usage Mining proposes cluster visitors of a website based on the page requests taking place on the sessions belonging to them. The aim of this study presented in Farzan (2004) is to discover the groups of pages that are visited together by many visitors. This information can then be used by the Web master in redesigning the Web Site or updating it with extra links between these pages

Park *et al.* (2008) proposed a general sequence based clustering method in association with Markov models for user, web page clustering. A new, fuzzy ART-enhanced K-means algorithm is also developed and its superior performance is demonstrated.

Mobasher *et al.* (2000) and Nakagawa and Mobasher (2003) presented a WebPersonalizer system which provides dynamic recommendations, as a list of hypertext links, to users. The analysis is based on anonymous usage data combined with the structure formed by the hyperlinks of the site. Data mining techniques (i.e. clustering, association rules and sequential pattern discovery) are used in the preprocessing phase in order to obtain aggregate usage profiles. In this phase, Web server logs are converted in clusters made up of sequences of visited

pages and cluster made up of set of pages with common usage characteristics. The online phase considers the active user session in order to find matches among the user's activities and the discovered usage profiles. Matching entries are then used to compute a set of recommendations which will be inserted into the last requested page as a list of hypertext links. WebPersonalizer is a good example of two-tier architecture for Personalization systems.

Mobasher *et al.* (2000) developed a recommendation system, termed Yoda that is designed to support large-scale Web-based applications requiring highly accurate recommendations in real-time. With Yoda, they introduced a hybrid approach that combines collaborative filtering (CF) and content-based querying to achieve higher accuracy. Yoda is structured as a tunable model that is trained online and employed for real-time recommendation on-line. The on-line process benefits from an optimized aggregation function with low complexity that allows real time weighted aggregation of the soft classification of active users to predefined recommendation sets.

Jespersen *et al.* (2002) proposed a hybrid approach for analyzing the visitor click sequences. A combination of hypertext probabilistic grammar and click fact table approach is used to mine web logs which could be also used for general sequence mining tasks.

Analog (Yan *et al.*, 1996) is one of the first WUM systems. It is structured according to an off-line and an online component. The off-line component builds session clusters by analyzing past users activity recorded in server log files. Then the online component builds active user sessions which are then classified according to the generated model. The classification allows identification of pages related to the ones in the active session and to return the requested page with a list of suggestions. The geometrical approach used for clustering is affected by several limitations, related to scalability and to the effectiveness of the results found.

Nevertheless, the architectural solution introduced was maintained in several other more recent projects.

Cooley *et al.* (2000) used a full spectrum of data mining algorithms for web personalization, based on transaction clustering, usage clustering and association rule discovery. Their proposed approach web personalization is based on examining past users activities. This information later is used for online recommendations.

Jalali *et al.* (2008a, 2008b) proposed a recommender system for navigation pattern mining through Web usage mining to predict user future movements. The approach is based on the graph partitioning clustering algorithm to model user navigation patterns for the navigation patterns mining phase. Furthermore, in the recommender phase, longest common subsequence algorithm is utilized to classify current user activities to foresee user next movement.

Graph partitioning theoretic approach is presented by Perkowitz and Etzioni (2000b), who have developed a system that helps in making Web sites adaptive, i.e., automatically improving their organization and presentation by mining usage logs. The core element of this system is a new clustering method, called cluster mining, which is implemented in the PageGather algorithm. PageGather receives user sessions as input, represented as sets of pages that have been visited. Using these data, the algorithm creates a graph, as signing pages to nodes. An edge is added between two nodes if the corresponding pages co-occur in more than a certain number of sessions. Clusters are defined either in terms of cliques, or connected components. Clusters defined as cliques prove to be more coherent, while connected component clusters are larger, but faster to compute and easier to find. A new index page is created from each cluster with hyperlinks to all the pages in the cluster. The main advantage of PageGather is that it creates overlapping clusters. Furthermore, in contrast to the other clustering methods, the

clusters generated by this method group together characteristic features of the users directly. Thus, each cluster is a behavioral pattern, associating pages in a Web site. However, being a graph based algorithm, it is rather computationally expensive, especially in the case where cliques are computed.

Cadez *et al.* (2000) in the Web CANVAS tool proposed a partitioning clustering method, which visualizes user navigation paths in each cluster. In this system, user sessions are represented using categories of general topics for Web pages. A number of predefined categories are used as a bias and URLs from the Web server log files are assigned to them, constructing the user sessions. The Expectation-Maximization (EM) algorithm (Mustapha *et al.*, 2009), based on mixtures of Markov chains is used for clustering user sessions. Each Markov chain represents the behavior of a particular subgroup. EM is a memory efficient and easy to implement algorithm, with a profound probabilistic background.

However, there are cases where it has a very slow linear convergence and may therefore become computationally expensive, although in the results in Cadez *et al.* (2000), it is shown empirically that the algorithm scales linearly in all aspects of the problem. The EM algorithm is also employed by Anderson *et al.* (2001) in two clustering scenarios, for the construction of predictive Web usage models. In the first scenario, user navigation paths are considered members of one or more clusters and the EM algorithm is used to calculate the model parameters for each cluster. The probability of visiting a certain page is estimated by calculating its conditional probability for each cluster. The resulting mixture model is named Naive Bayes mixture model, since it is based on the assumption that pages in a navigation path are independent, given the cluster. The second scenario uses a similar approach to (Cadez *et al.*, 2000). Markov chains that represent the navigation paths of users are clustered using the EM algorithm, in order to predict subsequent pages. Improving quality of clustering is the main objective in all previous works mentioned above. These works attempt to find

architecture and algorithm for this purpose, but the quality still does not meet satisfaction.

Mustapha *et al.* (2009) proposed clustering, using EM algorithm, for improving accuracy of user navigation clustering. The EM algorithm was used to find the maximum likelihood estimate of parameters in probabilistic models, where the model depends on unobserved latent variables. The experimental results represent that, by decreasing the number of clusters, the log likelihood converges toward lower values and at the same improved the visit-coherence (accuracy) of navigation pattern mining. Although the likelihood converged towards lower values, it still has provision to improve this system.

Raghavendra *et al.* (2010) modeled user behaviour as a vector of the time, the particular user spends at each URL and additionally categorize a new user access pattern. The clustering and classification methods of k-means with non-Euclidean similarity measure and artificial neural networks with consistent inputs were implemented and evaluated. Despite recognizing user behavior, this model can also be utilized as a prediction system which can be used to identify deviational behaviour.

Hussain *et al.* (2010a) proposed a structure for web session clustering at initial level of web usage mining. The structure will cover the data preprocessing phase to organize the web log data and transform the categorical web log data into numerical data. A session vector is acquired, so that suitable comparison and swarm optimization possibly will be applied to cluster the web log data. The hierarchical cluster based technique improves over the existing web session techniques for additional structured information about the user sessions.

- **Associative Rule Mining**

In the web domain, the pages, which are most often referenced together, can be put in one single server session by applying the association rule generation.

Association rule mining techniques can be used to discover unordered correlation between items found in a database of transactions. The problem of deriving Association Rules from data was first formulated in (Agrawal *et al.*, 1993) and is called the "market-basket problem". The problem is that we are given a set of items and a large collection of transactions which are sets (baskets) of items. The task is to find relationships among the containments of various items within those baskets.

There are many other examples where association rules have been used, for example users' visits of WWW pages which the structure and its content can be optimized. Xue *et al.* (2002) have used re-ranking method and generalized Association Rules to extract access patterns of the web sites pattern usage.

Mannila *et al.* (1994) use page accesses from a Web server log as events for discovering frequent episodes. The major data mining technique used in their research work was association rules. Chen *et al.* (1996b) introduce the concept of using the maximal forward references in order to break down user sessions into transactions for the mining of traversal patterns. Zhou *et al.* (2006a) used association knowledge to discover knowledge from web logs and recommended their system for online applications such as web recommendation and personalization. Their experiments showed that the rules generated are comparable in quality.

Association rule generation can be used to relate pages that are most often referenced together in a single server session. In the context of Web Usage Mining, association rules refer to sets of pages that are accessed together with a support value exceeding some specified threshold. These pages may not be directly connected to one another via hyperlinks. For example, association rule discovery using, the Apriori algorithm (or one of its variants), may reveal a correlation between users who visited a page containing electronic products to those who access a page about sporting equipment.

Aside from being applicable for business and marketing applications, the presence or absence of such rules can help Web designers to restructure their Web site. The association rules may also serve as a heuristic for prefetching documents in order to reduce user-perceived latency when loading a page from a remote site.

This research work proposes the use of associative rule mining algorithm for predicting users' future page requirement. This section presents the frequently used associative algorithms along with studies that use them for web usage mining.

(i) Algorithms

The most frequently used basic frequent itemset mining methodologies are Apriori, FP-growth and Eclat. As there are usually a large number of distinct single items in a typical transaction database and their combinations may form a very huge number of itemsets, it is challenging to develop scalable methods for mining frequent itemsets in a large transaction database. Agrawal and Srikant (1994) observed an interesting downward closure property, called Apriori, among frequent kitemsets: A k-itemset is frequent only if all of its sub-itemsets are frequent. This implies that frequent itemsets can be mined by first scanning the database to find the frequent 1-itemsets, then using the frequent 1-itemsets to generate candidate frequent 2-itemsets and check against the database to obtain the frequent 2-itemsets. This process iterates until no more frequent k-itemsets can be generated for some k. This is the essence of the Apriori algorithm and its alternative (Mannila *et al.* 1994).

From the point of introduction, there have been extensive studies on the improvements or extensions of Apriori, like hashing technique (Park *et al.*, 1995), partitioning technique (Savasere *et al.*, 1995), sampling approach (Toivonen, 1996), dynamic itemset counting (Brin *et al.*, 1997), incremental mining (Cheung *et al.*, 1996), parallel and distributed mining (Agrawal and Shafer, 1996;

Zaki *et al.*, 1997) and integrating mining with relational database systems (Sarawagi *et al.*, 1998). Geerts *et al.* (2001) derived a tight upper bound of the number of candidate patterns that can be generated in the level-wise mining approach. This result is effective at reducing the number of database scans.

In many cases, the Apriori algorithm significantly reduces the size of candidate sets using the Apriori principle. However, it can suffer from two-nontrivial costs: (1) Generating a huge number of candidate sets and (2) Repeatedly scanning the database and checking the candidates by pattern matching. Han *et al.* (2000a) devised an FP-growth method that mines the complete set of frequent itemsets without candidate generation.

FP-growth works in a divide-and-conquer way. The first scan of the database derives a list of frequent items in which items are ordered by frequency descending order. According to the frequency-descending list, the database is compressed into a frequent-pattern tree, or FP-tree, which retains the itemset association information. The FP-tree is mined by starting from each frequent length-1 pattern (as an initial suffix pattern), constructing its conditional pattern base (a "subdatabase", which consists of the set of prefix paths in the FP-tree co-occurring with the suffix pattern), then constructing its conditional FP-tree and performing mining recursively on such a tree. The pattern growth is achieved by the concatenation of the suffix pattern with the frequent patterns generated from a conditional FP-tree.

The FP-growth algorithm transforms the problem of finding long frequent patterns to searching for shorter ones recursively and then concatenating the suffix. It uses the least frequent items as a suffix, offering good selectivity. Performance studies demonstrate that the method substantially reduces search time.

There are many alternatives and extensions to the FP-growth approach, including depth-first generation of frequent itemsets. They are H-Mine, by Pei *et al.* (2001) which explores a hyper-structure mining of frequent patterns; building alternative trees, exploring top-down and bottom-up traversal of such trees in pattern-growth mining by Liu *et al.* (2002; 2003), and an array-based implementation of prefix-tree-structure for efficient pattern growth mining by Grahne and Zhu (2003).

Both the Apriori and FP-growth methods mine frequent patterns from a set of transactions in horizontal data format (i.e., {TID: itemset}), where TID is a transaction-id and itemset is the set of items bought in transaction TID. Alternatively, mining can also be performed with data presented in vertical data format (i.e., {item:TID_set}).

Zaki (2001) proposed EquivalenceCLAssTransformation (Eclat) algorithm by exploring the vertical data format. The first scan of the database builds the TID_set of each single item. Starting with a single item (k = 1), the frequent (k+1)-itemsets grown from a previous k-itemset can be generated according to the Apriori property, with a depth-first computation order similar to FP-growth (Han *et al.*, 2000a). The computation is done by intersection of the TID_sets of the frequent k-itemsets to compute the TID_sets of the corresponding (k+1)-itemsets. This process repeats, until no frequent itemsets or no candidate itemsets can be found.

Besides taking advantage of the Apriori property in the generation of candidate (k + 1)-itemset from frequent k-itemsets, another merit of this method is that there is no need to scan the database to find the support of (k+1)-itemsets (for $k \geq 1$). This is because the TID_set of each k-itemset carries the complete information required for counting such support.

Another related work which mines the frequent itemsets with the vertical data format is by Holsheimer *et al.* 1995. This work demonstrated that, though impressive results have been achieved for some data mining problems using highly specialized and clever data structures, one could also explore the potential of solving data mining problems using the general purpose database management systems (DBMS).

- **Web Usage Mining and Associative Algorithms**

In Web usage mining, association rules are used in order to discover the pages which are visited together even if they are not directly connected, which can reveal associations between group of users with specific interest. This information can be used for example for restructuring Web sites by adding links between those pages which are visited together. Association rules in Web logs are discovered by Chen *et al.* (1996a), Punin *et al.* (2001), Batista *et al.* (2002), Zaiane *et al.* (1998) and Shen *et al.* (2000).

Association rule generation can be used to relate pages that are most often referenced together in a single server session. In the context of Web Usage Mining, association rules refer to sets of pages that are accessed together with a support value exceeding some specified threshold. These pages may not be directly connected to one another via hyperlinks. For example, association rule discovery using the Apriori algorithm (or one of its variants) may reveal a correlation between users who visited a page containing electronic products to those who access a page about sporting equipment. Aside from being applicable for business and marketing applications, the presence or absence of such rules can help Web designers to restructure their Web site. The association rules may also serve as a heuristic for prefetching documents in order to reduce user-perceived latency when loading a page from a remote site.

There are many other examples where association rules have been used. For example in users' visits of WWW pages the structure and its content can be optimized. Baowen (2001) have used re-ranking method and generalized Association Rules to extract access patterns of the Web sites pattern usage. Mannila *et al.* (1994) use page accesses from a Web server log as events for discovering frequent episodes. The major data mining technique used in their research work was association rules. Chen *et al.* (1996b) introduce the concept of using the maximal forward references in order to break down user sessions into transactions for the mining of traversal patterns.

Mary and Malarvizhi (2010) used association rules to improve the prediction accuracy and claim that the method proposed is better than Streaming Association Rule (SAR) model. They enhanced the existing SAR mining model with Apriori-like algorithm and Dynamic programming approach. An enhanced pruning rule method for eliminating the redundancy was also introduced in the preprocessing phase. This pruning of rules leads to better prediction accuracy.

Batista and Silva (2001) perform mining process for online newspaper Web access logs by using Apriori algorithm. Apriori was the first scalable algorithm designed for association-rule mining algorithm. Apriori is an improvement over the AIS and SETM algorithms (Agrawal and Srikant, 1994). The Apriori algorithm searches for large itemsets during its initial database pass and uses its result as the seed for discovering other large datasets during subsequent passes. Rules having a support level above the minimum are called large or frequent itemsets and those below are called small itemsets (Chen *et al.*, 1996a).

Another approach to associative web log mining is the use of FP-growth algorithm. According to Sun and Zhang (2004) mining frequent patterns from Web logs is an important data mining task and candidate-generation-and-test and FP-growth are two representative frequent pattern mining approaches. They

conducted extensive experiments on real world Web log data to analyse the characteristics of Web logs and the behaviour of these two approaches on Web logs. To improve the performance of FP-Growth algorithm on mining Web logs, they proposed a new algorithm, namely, Combined Frequent Pattern Mining (CFPM) to cater for Web log data specifically.

Dong *et al.* (2005) used a combination of neural networks and FP-Growth algorithm to discover frequent patterns from web log files. In the same period, Li *et al.* (2005) used FP-growth algorithm to discover the path traversal patterns over web long click sequences. Romero *et al.* (2009) applied FP-growth algorithm for personalizing hyperlinks in Web-based adaptive educational systems.

Similarly, Sun and Zhao (2009) designed and implemented an E-learning model based on web usage mining Techniques built on FP-Growth. Recently, Kumar and Rukmani (2010) discovered web usage patterns using Apriori and FP-Growth algorithms. Yu *et al.* (2001) proposed a novel incremental mining algorithm using FP-growth algorithm for identifying frequent patterns of web usage mining. Their results were comparable with the existing standards and had the potential to a web prototype of Web Log Analyzer in web usage mining.

Wang *et al.* (2004) proposed a method that can discover users' frequent access patterns underlying users' browsing Web behaviors using association rules. They proposed a technique which used a revised algorithm (FAP-Mining) based on the FP-tree algorithm to mine frequent access patterns. The algorithm is accurate and scalable for mining frequent access patterns with different lengths. A detailed review of frequent pattern discovery in web log data is discussed by Iváncsy and Vajk (2006). Dong *et al.* (2007) presented a novel incremental mining algorithm of frequent patterns for web usage mining which used FP-Growth algorithm during frequent pattern mining.

2.5. CONCLUSION

In the current scenario, many industries have changed their focus from product orientation to customer orientation to retain regular frequent customers for the improvement of customer relationship management. From the review study, it can be understood that several studies have focused on providing valuable information for web designer to quickly respond to their individual needs. In almost all of these studies, the web log files play a significant role. Further, it can also be understood that, these proposals have concentrated on two main issues, namely, the accuracy of prediction and speed with which the results are delivered. In spite of the various researches addressing these issues, the field has several scopes for improvement.

In the present research work, to improve the performance of WARS for predicting the next page access in terms of accuracy and speed, an associative classification-based algorithm is proposed. This framework is optimized in three different manners. The first optimization operation analyzes and discovers a technique that best identifies the user and session, while the second optimization operation tries to reduce the size of web log data by focusing only on potential users in a session. The third optimization operation enhances the work of association rule mining method for predicting users' next page. The techniques used for each optimization operation are explained in the next Chapter, **Methodology**.