ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 2, No.8 , March 2014
ISSN : 2322-5157
www.ACSIJ.org

ACSIJ
WWW.ACSIJ.ORG

Applying Web Usage Mining Techniques to Design Effective Web Recommendation Systems: A Case Study

Maryam Jafari [1], Farzad Soleymani Sabzchi [2] and Amir Jalili Irani [3]

[1] Department of Computer, Novin Higher Education Institute
Ardabil, Iran
m123_jafari@yahoo.com

[2] Department of Computer, Novin Higher Education Institute
Ardabil, Iran
f_soleymani63@yahoo.com

[3] Sama technical and vocational training college, Islamic Azad University, Ardebil branch
Ardebil, Iran
amir_jalili_irani@yahoo.com

Abstract

Recommender systems are helpful tools which provide an adaptive Web environment for Web users. Recently, a number of Web page recommender systems have been developed to extract the user behavior from the user's navigational path and predict the next request as he/she visits Web pages. Web Usage Mining (WUM) is a kind of data mining method that can be used to discover this behavior of user and his/her access patterns from Web log data. This paper first presents an overview of the used concepts and techniques of WUM to design Web recommender systems. Then it is shown that how WUM can be applied to Web server logs for discovering access patterns. Afterward, we analyze some of the problems and challenges in deploying recommender systems. Finally, we propose the solutions which address these problems.

Keywords: *Recommender system, Web Usage Mining, Pattern discovery, Web server logs, personalization.*

1. Introduction

Recommender Systems (RS) are useful tools which guaranties that right information are accessible for right users at right time [1]. RSs are useful in different domains, such as Web personalization, information filtering, e-commerce and providing recommendations of books, movies and music. One of the most popular applications of recommender systems is Web environment personalizing by providing a list of items related to user's interests. Usually demographic, content based and collaborative based filtering techniques are employed to generate recommendations. In demographic filtering technique (DMF), users are categorized based on their personal attributes such as their age range and provide recommendation based on these demographic categories. Recommendations produced by these systems are too general and not adaptive with changes in user preferences over time. Content based filtering provides recommendations for users with similar conceptual based on previously evaluated items. Content based filtering methods usually utilize text extraction techniques for building user profiles. These methods have some disadvantages such as mismatch between user profile terms and item profile terms that leads to decreasing the performance [2]. Second type of recommender systems are Knowledge based. Knowledge based filtering technique represents recommendations according to the understanding of user's requirements and features of the item.

In other words, this type of recommender systems needs the knowledge of the system about user and item to generate a list of suggestions. One of the most popular techniques for recommender systems is the collaborative filtering approach that relies on the preferences of items expressed by users, usually under the form of ratings [3]. Collaborative based filtering technique suggests items based on similar users preferences.

Web Usage Mining (WUM) attempts to discover useful knowledge from the secondary data, especially those contained in Web log files. Other sources can be browser logs, user profiles, user sessions, bookmarks, folders and scrolls. These data are obtained from the interactions of the users with the Web. Effective Website management, creating adaptive Websites, business and support services, personalization, and network traffic flow analysis efficiently use WUM for better performance. WUM focuses on the techniques that could predict user's navigational behavior. In [4], there are three main tasks for performing WUM: Preprocessing, Pattern Discovery, and Pattern Analysis, as shown in Fig. 1. Because of the importance of pattern discovery for designing Web recommender systems, this paper focuses on describing

this phase and presents an overview of WUM and also provides a survey of the different techniques of pattern extraction used for WUM.

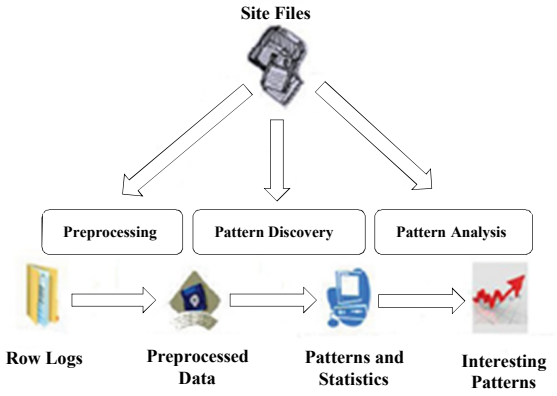

Fig. 1. Three main tasks of WUM [5].

The remainder of this paper organized as follows. In the next section motivation of the paper is presented. Some background on the RS is given in the section 3. In section 4 related works on WUM and designing recommender systems are reviewed. In section 5 an overview on WUM is presented and then different phases of WUM and the related techniques of these phases are examined. Also, applying of WUM techniques to design Web recommendation systems is described. Problems and limitations analysis of recommender systems and suggested solutions are explained in section 6. Finally a conclusion of this work is presented in section 7.

2. Motivation of the Research

Millions of users access to Websites in all over the world. When they access a Websites, a large amount of data generated in log files which is very important because many times user repeatedly access the same type of Web pages and the record is maintained in log files. These series can be considered as a Web access pattern which is helpful to find out the user behavior in order to design recommender systems. Through this behavior information, we can find out the accurate user next request prediction that can reduce the browsing time of Web pages.
In recent years, there has been an increasing number of research works done with regard to WUM. The main motivation of this survey is to know what research has been done on WUM in future request prediction. Also, how we can use various WUM techniques to develop efficient and effective recommendation systems.

3. Recommender Systems

Web recommendation systems have been widely used to help users locate information on the Web. RSs appeared as an independent research area in the mid-1990s when researchers were focusing on recommendations that strongly relied on the rating structure. For example, in a movie recommending application (such as the one at MovieLens [6]), users initially rate a subset of movies that they have already seen. The usual formulation of the problem is then to predict the interest of an active user to an unseen item. This estimation could be used to recommend items to the user that she/he has the highest interest degree to them. Many different approaches to the RS have been published [7,8], using methods from machine learning, approximation theory and various heuristics. RSs are usually classified into the following categories, independently of the technique being used, based on how the recommendations are made.

3.1 Content-based Recommender System

Content-based RS is an outgrowth and continuation of information filtering research [9]. This system stores content information about each item to be recommended. This information will be used to recommend items similar to the ones the user preferred in the past. This task is done based on how certain items are similar to each other or the similarity with respect to user preferences. In a content-based system, the objects of interest are defined by their associated features. For example, text recommendation systems like the newsgroup filtering system NewsWeeder [10] uses the words of their texts as features. The main problems with content-based RS are firstly, the difficulty of making accurate recommendations to users with very few ratings, and secondly, overspecialization since the system should recommend similar items to the users according to the previous users' rating. Decision trees, neural nets, and vector-based representations have all been used. As in the collaborative case, content-based user profiles are long term models and updated as more evidence about user preferences is observed.

3.2 Knowledge-based Recommender System

Knowledge-based RS is a type of recommender systems that uses knowledge about users and products to follow a knowledge-based approach to generating a suggestion, reasoning about what products meet the user's requirements. Knowledge-based recommendation attempts to suggest objects based on inferences about a user's needs and preferences. Also this system can reason about the relationship between a need and a possible recommendation. The user profile can be any knowledge structure that supports this inference. In the simplest case,

as in Google search engine, it may simply be the query that the user has formulated. In other systems, it may be a more detailed representation of the user's requirements using explicit user models [11]. An example of this kind of systems is the restaurant recommender Entrée [12,13]. This recommender system makes its recommendations by finding restaurants in a new city similar to restaurants the user knows and likes. The system allows users to navigate by expressing their preferences with respect to a given restaurant, thereby refining their search criteria. The Entree system and several other systems for example, [14] employ techniques from case-based reasoning for knowledge-based recommendation. Schafer, Konstan & Riedl call knowledge-based recommendation the "Editor's choice" method.

3.3 Collaborative Filtering Recommender System

Collaborative filtering RS is one of the most successful approaches to building recommender systems. This approach uses the known preferences of a group of users to make recommendations or predictions of the unknown preferences for other users. In the other word, this system attempts to identify groups of people with similar interests to those of the user and recommend items that they have liked. CF techniques use a database of users' preferences about items to predict products that a new user might like. In a typical CF scenario, there is a list of m users $\{u_1, u_2, \dots, u_m\}$ and a list of n items $\{i_1, i_2, \dots, i_n\}$, and each user, u_i, has a list of items, Iu_i, which the user has rated explicitly, or about which their preferences have been reasoned through their behaviors implicitly [15].
Collaborative RSs have their own limitations. First, there is the new user problem since it is necessary for a set of items to be rated in order to perform similarity analysis, and the greater the number of ratings performed by a user, the more accurate the assignment to a group of similar users. Second limitation is the new item problem since an item which has not been rated previously cannot be recommended.
According to [16], collaborative RS can be grouped into memory-based and model-based approaches. On the one hand, memory-based algorithms use the whole rating matrix to make recommendations. In order to do so, they use some kind of aggregation measure considering the ratings of other (most similar) users for the same item. Different models can be obtained by considering different similarity measures and different aggregation criteria. The most common approach that is introduced in [17] is:

$$r_{a,j} = k \sum_{U_i \in U_a^*} sim(U_a, U_i) \times r_{i,j} \qquad (1)$$

where $r_{i,j}$ denotes the rating given by user U_i to the item I_j, k is a normalizing factor, sim is a similarity (distance) measure between users and U_a^* denotes the set of users that are most similar to U_a. On the other hand, in model-based algorithms the predictions are made by building an explicit model of the relationships between items in offline phase. This model is then used to finally recommend the product to the users in online phase. In this approach, the predictions are therefore not based on any ad hoc heuristic, but rather on a model learnt from the underlying data using statistical and machine learning techniques.

3.4 Hybrid Recommender System

Hybrid RS is a combination of several techniques to present recommendation, including content-based, collaborative, knowledge-based and other techniques to improve performance. Different combination methods could be used from running alternatives separately and combining their prediction to construct a general unifying model which incorporates combined models.

4. Related Works

During the recent years, many researches are done for personalizing Websites using Variety of techniques. In the field of swarm intelligence algorithms, Ujjin and Bently have presented a recommender system based on particle swarm optimization algorithm [18]. They proved that the results obtained from their PSO recommender system are more accurate than the genetic and Pearson algorithm. In another work a fuzzy genetic recommender system with the accuracy of memory based CF and the scalability of model based CF. their novel user model helps achieving complexity and sparsity reduction in system. The performance of their method is proved by comparing the results with Pearson and fuzzy recommender system [19]. Sobecki used ant colony metaphor for selecting optimal solutions in his hybrid recommendation method and Bedi also, presented a recommender system based on collaborative behavior of ants. He used collaborative filtering approach and generated recommendations for Jester dataset [20]. Clustering is an important step in all recommendation systems, choosing an appropriate clustering algorithm leads to producing more qualified recommendations. The c-means and k-means algorithms are most well-known clustering algorithms [21].
Fuzzy c-means is a method of clustering which allows one piece of data to belong to two or more clusters. This method developed by Dunn in 1973 and improved by Bezdek in 1981. Swarm algorithms are also used for clustering items. Ant based clustering has been introduced by Deneuborg, in this algorithm ants discriminate between different kinds of items and spatially arrange them according to their properties. This algorithm is modeled of the real behavior of ants in nature. The proposed approach by Kanade and Hall (2003), presents the combination of ant based clustering and FCM [22]. Their model is

employed in this study for clustering Web users based on their accesses to Web pages.
Jalali et al., [23] presented a system for extracting user's navigational behavior using a graph partitioning model. An undirected graph based on connectivity between each pair of Web pages was considered and also proposed a new formula for allocating weights to edges of the graph.
In [24] the prediction of user's navigation patterns is proposed using clustering and classification from Web log data. First phase of this method focuses on separating users in Web log data, and in the second phase clustering process is used to group the users with similar preferences and in the third phase the results of classification and clustering are used to predict the users' next requests.
Emine Tug et al., [25] found sequential accesses from Web log files, using Genetic Algorithm (GA) that called Automatic Log Mining via Genetic (ALMG). In their work, GA based on evolutionary approach for pattern extraction was used to found best solutions for time consuming problem to discover sequential accesses from Web log data.
Kim and Zhang use a genetic algorithm to learn the importance factors of HTML tags which are used to re-rank the documents retrieved by standard weighting schemes for Web structure mining [26]. Picarougne et al. present a genetic search strategy for a search engine [27]. Abraham and Ramos propose an ant clustering algorithm to discover Web usage patterns (data clusters) and a linear genetic programming approach to analysis the visitor trends [28].
Some systems have been developed based on Web mining for automatic personalization [29-31]. They generally consist of two major processes: off-line mining and on-line recommendation. In the off-line mining process, all the access activities of users in a Website are recorded into the log files by the Web server. Then, some Web mining processes are applied to the server logs to mine the hidden navigation models of users. In the on-line recommendation process, user's requests from his current active session are recorded. By comparing these requests with the models obtained from the off-line mining, appropriate personalized recommendations are generated. Mobasher et al. [32] made an attempt to integrate both usage and content attributes of a site into a Web mining framework for Web personalization. A ''post-mining'' type approach was implemented to obtain the uniform representation for both site usage and site content profiles to facilitate the real-time personalization. However, the techniques proposed in [28] were limited to the use of clustering to separately build site usage and content profiles.
In [33], Sarukkai has discussed about link prediction and path analysis for better user navigations. He proposes a Markov chain model to predict the user access pattern based on the user access logs previously collected. Chen et al. [34] introduce the concept of using the maximal forward references in order to break down user sessions into transactions for the mining of traversal patterns. A maximal forward reference is the last page requested by a user before backtracking occurs, where the user requests a page previously viewed during that particular user session.

5. Web Usage Mining

Web usage mining is the process of applying data mining techniques to the discovery of behavior patterns based on Web data, for various applications. In the advance of ecommerce, the importance of WUM grows larger than before. The overall process of WUM is generally divided into two main tasks; data preparation and pattern discovery. The data preparation tasks build a server session file where each session is a sequence of requests of different types made by single user during a single visit to a site. Paper [35] presented a detailed description of data preparation methods for mining Web browsing patterns. The pattern discovery tasks involve the discovery of association rules, sequential patterns, usage clusters, page clusters, user classifications or any other pattern discovery method [32,36]. Usage pattern extracted from Web data can be applied to a wide range of applications such as Web personalization, system improvement, site modification, business intelligence discovery, usage characterization, and so on [4].
Web usage data captures Web-browsing behavior of users from a Web site. The task of modeling and predicting a user's navigational behavior on a Web site or on a Web domain can be useful in quite many Web applications such as Web server caching that provides an interface between a single Web server and all of its users. It reduces the number of requests the server must handle, and then helps load balancing, scalability and availability [37,38]. Web page recommender systems that help people make decisions in complex information space where the volume of information available to them is huge [39,40].
Web search engines that usually help users locate information based on the textual similarity of a query and potential documents [41,42] and Web search personalization that its goal is to tailor search results to a particular user based on that user's interests and preferences [43]. WUM has other several applications [44] such as: business intelligence, e-Learning, e-Business, e-Commerce, e-Newspapers, e-Government and Digital Libraries. Most of the WUM techniques are based on association rules, sequential patterns and clustering [45].

5.1 Preprocessing

The information available in the Web is heterogeneous and unstructured. Therefore, the preprocessing phase is a prerequisite for discovering patterns. The goal of preprocessing is to transform the raw click stream data into

a set of user profiles [46]. Data preprocessing presents a number of unique challenges which led to a variety of algorithms and heuristic techniques for preprocessing tasks such as merging and cleaning, user and session identification etc. [47]. Various research works are carried in this preprocessing area for grouping sessions and transactions, which is used to discover user behavior patterns.

The information available in the Web is heterogeneous and unstructured. Therefore, data preprocessing is predominantly significant phase in WUM. The goal of preprocessing is to transform the raw collected data into a set of user profiles [48]. This phase is often the most time consuming and computationally intensive step in WUM but it is necessary to have a successful analysis of Web usage patterns.

Every log entry of Web server log contains the traversal time from one page to another, the IP address or domain name, time and type of request (GET and POST, etc.), address of the page being accessed and other data [49].

Preprocessing removes many entries from the data files that are considered uninteresting for pattern discovery. Various research works are carried in this area for grouping sessions and transactions, which is used to discover user's navigation patterns. In brief, the whole process deals with the conversion of raw Web server logs into a formatted user session file in order to perform effective pattern discovery and analysis phases.

Generally, data preprocessing has four main tasks that are called data cleaning, user identification, session identification and path completion, as shown in Figure 2.

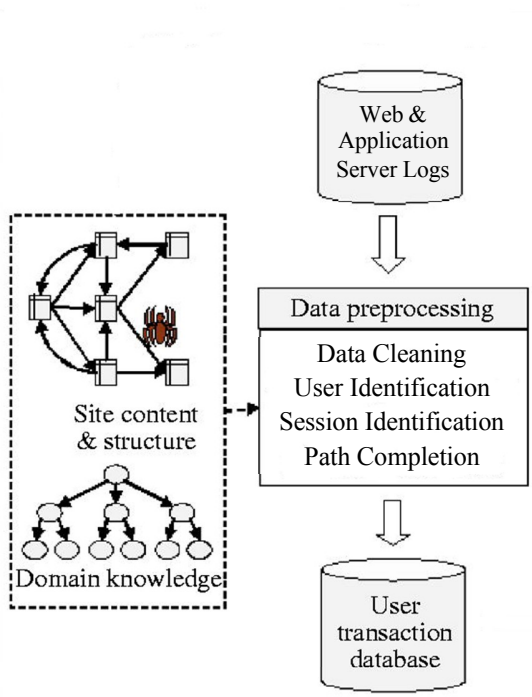

Fig. 2. Steps in data preprocessing for WUM.

5.1.1 Data Cleaning

In this task the server log is examined to remove the irrelevant and redundant items for the mining process.

There are three kinds of irrelevant or redundant data needed to clean:

- Accessorial resources embedded in HTML file: A user's request to view a favorite page often records in several log file entries since file requests that the user did not explicitly request such as graphic files and scripts add entries in Web log file. Therefore, all log entries with an extension such as gif, jpeg, GIF, JPEG, jpg, JPG, css, cgi and map in their filename should be removed.
- Robots' requests: Search engines such as Google periodically use Web robots (also called spiders) to navigate on Web and do accurate searches on Websites update their search indexes [50]. Therefore, it is required to try to rid the Web log file of these types of automatic access behavior.
- Error requests: Erroneous files are useless for WUM and can be removed by examining the HTTP status codes. For example, if the status code is 404 it means that the requested resource is not existence, so this log entry can be removed. Finally, log entries with status codes between 200 and 299 that give successful response are kept and entries which have other status codes are removed.

5.1.2 User Identification

This step focuses on separating the Web users from others. User Identification means identifying Unique users considering their IP address. Following heuristics are used to identify unique users:

- If there is a new IP address, then there is a new user.
- For more logs, if the IP address is same, but the operating system or browsing software are different, a reasonable assumption is that each different agent type for an IP address represents a different user.

Existence of local caches, corporate firewalls and proxy servers greatly complicate user identification task. The WUM methods that rely on user cooperation are the easiest ways to deal with this problem. However, it's difficult because of security and privacy.

5.1.3 Session Identification

Visited pages in a user's navigation browsing must be divided into individual sessions. A session means a set of Web pages viewed by a particular user for a particular purpose. At present, the methods to identify user session include timeout mechanism and maximal forward reference mainly [35].

The following rules are used to identify a session:

- For any new IP address in Web log file, a new user and also a new session will be created.
- In one user session, if the refer page in an entry of Web log file is null, a new session will be considered.
- If the time between page requests is more than 25.5 or 30 minutes, it is assumed that the user is starting a new session.

5.1.4 Path Completion

Many of important page accesses are missed in the Web log file due to the existence of local cache and proxy server. The task of path completion is to fill in these missing page references and makes certain, where the request came from and what all pages are involved in the path from the start till the end.

5.2 Pattern Discovery

Pattern discovery is a phase which extracts the user behavioral patterns from the formatted data. For this reason the data have to be converted in the preprocessing phase such that the output of the conversion can be used as the input of this phase. In pattern discovery phase, several data mining techniques are applied to obtain hidden patterns reflecting the typical behavior of users. Some important techniques for this phase are: path analysis, standard statistical analysis, clustering algorithms, association rules, classification algorithms, and sequential patterns. In the following, some of these techniques are described.

5.2.1 Statistical Analysis

Statistical analysis is the most common form of analysis to extract knowledge about visitors' behavior. By analyzing the obtained session file from Web log, useful statistical information such as frequency, mean, median, etc. can be resulted. This statistical information is used to produce a periodic report from the site such as information about users' popular pages, average visit time of a page, average time of users' browsing through a site, average length of a navigational path through a site, common entry and exit pages and high-traffic days of site.
According to these reports, it is clear that used statistical technique for pattern discovery perform a sketchy analysis on preprocessed data but obtained knowledge can be useful. For instance detecting entry points which are not home page or finding the most common invalid URL lead to enhance system performance and security and also facilitate the site topology modification task. The useful statistical information discovered from Web logs is shown in Table1.

Table 1: Important Statistical Information

Statistics	Detailed Information
Website Activity Statistics	Total number of visits Main 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

5.2.2 Sequential Patterns

The technique of sequential pattern discovery is to find inter-session patterns such that the presence of a set of pages is followed by another page in the time-stamp ordered session set. This mining is trying to find the relationships between sequential visits, to find if there exists any specific time order of the occurrences. The goal of this technique is to discover time ordered sequences of URLs followed by past users, in order to predict future pages. This prediction helps Web marketers to target advertising aimed at groups of users based on these patterns. An example of Web server access logs analysis by using the Web mining system can show temporal relationships discovering 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.
- 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.

5.2.3 Classification

Classification is to build automatically a model that can classify a set of pages that is the task of mapping a page into one of several predefined classes [51]. In the Web domain, classification techniques allow one developing a profile of users which are belonging to a particular class or category and access particular server files. This requires extraction and selection of features that based on demographic information available on these users, or based on their access patterns. This technique has two steps. The first step is based on the collection of training

data set and a model is constructed to describe the features of a set of data classes. In this step, data classes are predefined so it is known as supervised learning. In the second step, the constructed model is used to predict the classes of future data. For example, classification on server access logs may lead to the discovery of interesting patterns such as the following:

- Users from state or government agencies who visit the site tend to be interested in the page /company/lic.html.
- 60% of users, who placed an online order in /company/products /Music, were in the 18-25 age groups and lived in Chandigarh.

5.2.4 Clustering

Clustering is another mining technique similar to classification however unlike classification there are no predefined classes therefore, this technique is an unsupervised learning process. This technique is used to group together users or data items that have similar characteristics, so that members within the same cluster must be similar to some extent, also they should be dissimilar to those members in other clusters.
In the WUM domain, clustering techniques are mainly used to discover two kinds of interesting clusters: user clusters and page clusters. Clustering of users is to cluster users with similar preference, habits and behavioral patterns. Such knowledge is especially used for automated return mail to users falling within a certain cluster, or dynamically changing a particular site for a user, on a return visit, based on past classification of that user (provide personalized Web content to the users). On the other hand, clusters of Web pages contain pages that seem to be conceptually related according to the users' perception. The knowledge that is obtained from clustering in WUM is useful for performing market segmentation in ecommerce, designing adaptive Web sites and designing recommender systems.

5.2.5 Association Rule Mining

In the context of WUM, once sessions have been identified association rules can be used to relate pages that are most often referenced together in a single 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. 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. The typical rule mined from database is formatted as (2):

$$X \rightarrow Y\,[\text{Support}, \text{Confidence}] \qquad (2)$$

It means the presence of item (page) X leads to the presence of item (page) Y, with [Support]% occurrence of [X,Y] in the whole database, and [Confidence]% occurrence of [Y] in set of records where [X] occurred.

$$\text{Support} = P(A \cap B) = \frac{\text{number of sessions that contain A and B}}{\text{total number of sessions}} \qquad (3)$$

$$\text{Confidence}(X \rightarrow Y) = \frac{\text{Support}(X \cap Y)}{\text{support}(X)} \qquad (4)$$

Many algorithms can be used to mine association rules from the data available; one of the most used and famous is the Apriori algorithm proposed and detailed by Agrawal and Srikant in 1994 [52]. This algorithm, given the minimum support and confidence levels, is able to quickly give back rules from a set of data through the discovery of the so-called large item set.
For example, if one discovers that 80% of the users 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 users 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. Discovery of such rules for organizations engaged in electronic commerce can help in the development of effective marketing strategies. 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 pre-fetching of Web data.
This type of result is for instance produced by [53] using a modification of the Apriori algorithm [54]. Reference [55] proposes and evaluates measures of interest to evaluate the association rules mined from Web usage data. Reference [56] exploits a mixed technique of association rules and fuzzy logic to extract fuzzy association rules from Web logs.

5.3 Pattern Analysis

Pattern analysis is the last step in the overall WUM process and the aim of this process is to extract the interesting rules, patterns or statistics from the output of

the pattern discovery process by filtering the irrelative rules or statistics. The exact analysis methodology is usually governed by the application for which Web mining is done. The most common form of pattern analysis is combining WUM tools with a knowledge query mechanism such as SQL. Visualization techniques, such as graphing patterns or assigning colors to different values, can often highlight overall patterns or trends in the data. Content and structure information can be used to filter out patterns containing pages of a certain usage type, content type, or pages that match a certain hyperlink structure.

6. Problems and Limitations Analysis of Recommender Systems

In this section, we present some of the common problems in deploying recommender systems, as well as some research directions that address them.

6.1 Sparsity

Stated simply, most users do not rate most items and hence the consumer-product interaction matrix is typically very sparse. A serious limitation of the collaborative filtering approach is the sparsity problem, referring to the situation where insufficient historical transactions are available for inferring reliable consumer similarities. This problem often occurs when a system has a very high item-to-user ratio, or the system is in the initial stages of use. When such systems have access only to a small number of past transaction records (relative to the total numbers of the books and consumers), however, determining which consumers are similar to each other and what their interests are becomes fundamentally difficult. The dimensionality reduction approach addresses the sparsity problem by removing unrepresentative or insignificant consumers or products to condense the user ratings matrix. However, potentially useful information might be lost during this reduction process.
This issue can be mitigated by using additional domain information or making assumptions about the data generation process that allows for high-quality imputation.

6.2 The Cold-start Problem

New items and new users pose a significant challenge to recommender systems. Cold-start [57] refers to the situation in which an item cannot be recommended unless it has been rated by a substantial number of users. The first of these problems arises in Collaborative Filtering systems, where an item cannot be recommended unless some users have rated it before.
This problem applies to new and obscure items and is particularly detrimental to users with eclectic taste. As such the new-item problem is also often referred to as the first-rater problem. Likewise, a new user has to rate a sufficient number of items before the recommendation algorithm be able to provide reliable and accurate recommendations.
As such, research in this area has primarily focused on effectively selecting items to be rated by a user so as to rapidly improve recommendation performance with the least user feedback. In this setting, classical techniques from active learning can be leveraged to address the task of item selection. Also, we can utilize social networks' information to fill the existing gap in cold-start problem and find similarities between users [58].

6.3 Fraud

Because of economic benefits to the retailers and service providers, recommender systems are being increasingly adopted by commercial Websites. This has led to many unscrupulous vendors engaging in different forms of fraud to game recommender systems for their benefit. There are two concepts in fraud problem that are named push attacks and nuke attacks.
Push attacks typically happens when vendors attempt to inflate the perceived desirability of their own products and nuke attacks is the lower ratings of their competitors. These types of attack have been broadly studied as shilling attacks [59] or profile injection attacks [60]. Such attacks usually involve setting up dummy profiles, and assume different amounts of knowledge about the system. Another attack is the average attack [59] that assumes knowledge of the average rating for each item and the attacker assigns values randomly distributed around this average, along with a high rating for the item being pushed. Studies have shown that such attacks can be quite detrimental to predicted ratings, though item-based Collaborative Filtering tends to be more robust to these attacks [59].
Obviously, content-based methods, which only rely on a user past ratings, are unaffected by profile injection attacks. While pure content-based methods avoid some of the traps discussed above, Collaborative Filtering still has some major advantages over them. The first advantage is that CF can perform in domains where there is not much content associated with items, or where the content is continuously changing or it is difficult for a computer to analyze, such as ideas, opinions, etc. Second advantage is that a CF system has the ability to provide opportune recommendations, i.e. it can recommend items that are relevant to the user, but do not contain content from the user's profile.

6.4 Scalability

When the population of existing users and items grow tremendously, a typical web-based recommender system running on an existing algorithm will suffer seriously the scalability problem. Therefore, there are demands for a

new approach that can quickly produce high quality predictions and can resolve the very large-scale problem. Specially, recommender systems based on nearest neighbor algorithms require computation that grows with both the number of users and the number of items.
One of approaches that we can use is clustering techniques. Once clustering is done, performance can be quite good, since the size of a cluster that must be analyzed is much smaller. Therefore, the clustering-based method can solve the very large-scale problem in recommender systems [61]. However, Clustering techniques may lead to worst prediction accuracy than other methods.

6.5 Synonymy

When a number of the same or very similar items to have different names or entries. Most recommender systems are not able to discover this hidden association and thus treat these items differently. When a user whose opinions do not consistently correlate in agreement or disagreement with any group of people and thus not benefit from the system. For instance, "kids film" and "children film" which are the apparently different items, they are actual the same item, but memory-based Collaborative Filtering (CF) systems would find no match between them to compute similarity. By prevalence of synonyms reduces the performance of recommendation which recommender systems make. These situations lead to a problem that is called Gray Sheep and Black Sheep. The users in the group of gray sheep problem are also responsible for increased error rate in collaborative filtering recommender systems [62], which often result in failure of recommender systems. Black sheep are those users who have no or very few people who they correlate with. This situation makes it very difficult to make recommendation for them [63].
Latent Semantic Indexing (LSI) method which is one of the Singular Value Decomposition (SVD) techniques is able to cope with the synonymy problems. A large matrix of term-document association data is captured by SVD and a semantic space would be created where terms and documents that have closely association are located to each other. The arrangement of the space could be done by SVD which display the most significant associative items and users from the data. This also denies the smaller and the less essential items and users. The performance of LSI in dealing with the synonymy problem is impressive at higher recall levels where precision is ordinarily quite low, thus representing large proportional improvements.
However, the performance of the LSI approach at the lowest levels of recall is weak [29]. The LSI method provides just a partial result to the synonymy problem, which refers to the fact that most words have more than one distinct meaning [29].

6.6 Algorithms

Typically, recommendation algorithms rely only on user information (collaborative features) such as navigational history or rating data, and additional information such as content features of the items, which may provide valuable source of complementary knowledge about user's activities, is usually ignored. By incorporating content information with a user's navigation or rating behavior, we may be able to gain a deeper understanding of her underlying interests.
In order to make use of available data sources, different combination methods are tried to make recommendations more effective and interpretable. Generally, an integrated approach is preferred during the mining or model learning phase to avoid subjective or ad hoc ways of combining evidence. Various WUM techniques used in designing recommender systems which are developed work well for Websites which do not have a complex structure, but experiments on complex, highly interconnected Websites show that the storage space and runtime requirements of these techniques increase due to the large number of patterns for sequential pattern and association rules, and the large number of states for Markov models. It may be possible to prune the rule space, enabling faster on-line prediction.
All recommender systems based on WUM techniques have strengths and weaknesses. Experimental results of the previous studies show that using a recommender model as a module of hybrid recommender system, which has a lower accuracy comparing to the other modules of the hybrid model, decreases the final recommendation accuracy. Therefore, the need for hybrid approaches that combine the benefits of multiple algorithms has been introduced [64].
However, most of the hybrid recommender systems switch between recommendation algorithms which work independently, or combining different algorithms in one algorithm. In recent years, there has been an increasing interest in applying Web content mining techniques to build Web recommender systems. However, the Web content mining techniques are unable to handle constantly changing Websites, such as news sites, and dynamically created Web pages. Thus, using Web content mining techniques in a recommender model leads to update the model frequently.

Table 2: Different challenges of recommender systems and their solutions

Problem	Description	Solution
Sparsity	-Not rating most items by users -The lack of sufficient historical transactions for inferring reliable similarities between users	-The dimensionality reduction approach for insignificant users or items -Condensing the user ratings matrix -Using additional domain information -Making assumptions about the data generation process for high-quality imputation
Cold-start	-The problem of recommending a new item to users which is not already rated (cold-start of the items). -The problem of recommending items to the users who have not rated a sufficient number of items already (cold-start of the users).	-Utilizing social networks' information to fill the existing gap in cold-start problem -Using classical techniques from active learning
Fraud	- Using recommender systems by unscrupulous vendors to inflate the perceived desirability of their own products	- Applying collaborative filtering to counter shilling attacks (push attacks and nuke attacks)
Scalability	-A dense population of users and items	- Using clustering techniques
Synonymy	- The tendency of the same or similar items to have different names - Treating recommender systems dealing with these items differently - Gray Sheep and Black Sheep	- Latent Semantic Indexing (LSI) method from Singular Value Decomposition (SVD) techniques
Algorithms	-The large number of discovered patterns through sequential pattern mining and association rule mining algorithms - Weak performance of algorithms in complex and highly interconnected Websites (increasing the storage space and runtime) -Decreasing the accuracy of hybrid recommender due to applying non-precision approach -Lack of ability in handle constantly changing Websites in recommender systems based on web content mining	- Pruning the rule space -Appropriate using of precision methods to build hybrid recommender systems -Updating the model frequently in content based models
Lack of Data	- The need of a lot of data to efficiently make recommendations	-Utilizing different data in Web such as Web usage data, Web content data and Web structure data

6.7 Lack of Data

Perhaps the most important challenge encountering Web recommender systems is that they require a lot of data to efficiently make recommendations. Companies such as Google, Amazon, Netflix, Last.fm are identified because of having excellent recommendations based on a lot of consumer user data. Having more data about items and users causes a recommender system can have the stronger chances of making good recommendations. So an effective recommender system should obtain a lot of data for the recommendations. Not only this data could gain from Web usage data, but also it can obtain from Web content data and Web structure data. Then it must capture and analyze user navigational patterns.
Table.2 shows a summary of various challenges of recommender systems and the solutions related to each one which is explained above.

7. Conclusion

Recommender systems are a powerful new technology for extracting useful information from users' behavioral databases and guiding them better in Web. These systems help users find items, Web pages or products which they are interested. Conversely, they help the business companies, online shops and search engines to have more interaction with Web users. Recommender systems are quickly becoming an important tool in predicting users' behavior on the Web. Recommender systems are being stressed by the large amount of users' data in existing corporate databases, and will be stressed even more by the increasing amount of users' data accessible on the Web. New technologies are required that can efficiently improve the challenges facing the recommender systems. One of these techniques is applying WUM to design effective Web recommendation systems. According to that nowadays discovering hidden information from large

amount of Web log data collected by Web servers is very difficult, pattern discovery has become one of the most important phases in WUM. This paper presented a complete introduction to WUM and focused on methods that can be used for the task of pattern extraction from Web log files. After discovering patterns, the result will be used for pattern analysis phase. Analyzing of the Web users' navigational patterns can help understand the user behaviors for constructing Web recommender systems. Therefore, the design of these Web applications will be improved.

In summary, recommender systems have been extensively explored in several Web fields. However, the quality of recommendations and the user satisfaction with such systems are still not optimal.

References

[1] M. Göksedef Ş. Gündüz, and Öğüdücü, " A Consensus Recommender for Web Users ", ADMA '07 Proceedings of the 3rd international conference on Advanced Data Mining and Applications, pp. 287-299.

[2] P. Shoval, V. Maidel, B. Shapira, and M. Taieb-Maimon, "Ontological content-based filtering for personalized newspapers: A method and its evaluation", Online Information Review, 2010, Vol. 34, No. 5, pp. 729-756.

[3] A. Brun, S. Castagnos, and A. Boyer, "From Community Detection to Mentor Selection in Rating-Free Collaborative Filtering", Advances in Multimedia, 2011, pp.1-19.

[4] J. Srivastava, "Web usage mining: discovery and applications of usage patterns from Web data", ACM SIGKDD Explorations Newsletter, 2000, Vol. 1, No. 2, pp. 12-23.

[5] L.K. Joshila Grace, V. Maheswari, and D. Nagamalai, "Analysis of Web Logs and Web User in Web Mining", International Journal of Network Security & Its Applications (IJNSA), 2013, Vol. 62, No. 5, pp. 199-202.

[6] B. Miller, I. Albert, S. Lam, J. Konstan, and J. Rield, "Movielens unplugged: experiences with an occasionally connected recommender systems", IUI '03 Proceedings of the 8th international conference on Intelligent user interfaces, 2002, Vol. 8, pp. 263–266.

[7] J. Ben Schafer, D. Frankowski, J. Herlocker, and Sh. Sen, "Collaborative filtering and recommendation systems", Springer-Verlag Berlin, 2007, pp. 291-324.

[8] G. Andomavicius, and A. Tuzhilin, "Toward the next generation of recommender system: A survey of the state-of the-art and possible extensions", IEEE Transactions on Knowledge and Data Engineering, 2005, Vol. 17 Issue 6, pp. 734-749.

[9] N.J. Belkin and W.B. Croft, "Information filtering and information retrieval: Two sides of the same coin?", Communications of the ACM, 1992, pp. 29-38.

[10] K. Lang, "Newsweeder: Learning to filter news", 12th International Conference on Machine Learning, 1995, pp. 331-339.

[11] B. Towle and C. Quinn, "Knowledge Based Recommender Systems Using Explicit User Models", In Knowledge-Based Electronic Markets, 2000, pp.74-77.

[12] R. Burke, K. Hammond. and E. Cooper, "Knowledge-based navigation of complex information spaces", In Proceedings of the 13th National Conference on Artificial Intelligence, 1996, pp. 462-468.

[13] R. Burke, K. Hammond, and B. Young, "the Find Me Approach to Assisted Browsing", IEEE Expert, 1997, pp. 32-40.

[14] S. Schmitt and R. Bergmann, "Applying case-based reasoning technology for product selection and customization in electronic commerce environments", 12th Bled Electronic Commerce Conference, 1999, pp. 7-9.

[15] X. Su and T.M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques", Advances in Artificial Intelligence, 2009, pp. 1-19.

[16] J.S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering", 14th Conference on Uncertainty in Artificial Intelligence, 1998, pp. 43–52.

[17] M. Luis, J. de Campos, M. Fernandez-Luna, J.F. Huete, "A collaborative recommender system based on probabilistic inference from fuzzy observations", Fuzzy Sets and Systems, 2008, pp. 1554 – 1576.

[18] S. Ujjin and P.J. Bentley, "Particle swarm optimization recommender system", Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE, pp. 124-131.

[19] H. Mohammad Yahya, K. Al-Shamiri, and K. Bharadwaj, "Fuzzy Genetic Approach to Recommender Systems based on a Novel Hybrid User Model", Journal Expert Systems with Applications: An International Journal, 2008, Vol. 35 Issue 3, pp. 1386-1399.

[20] J. Sobecki, "Web-Based System User Interface Hybrid Recommendation Using Ant Colony Metaphor", Knowledge-Based Intelligent Information and Engineering Systems, 2008, Vol. 4694, pp. 1033-1040.

[21] M.N. Vathatis, B. Boutsinas, P. Alevizos and G. Pavlides, "The new K-windows algorithm for improving the K-means clustering algorithm", Journal of Complexity, 2002, Vol. 18, Issue 1, pp. 375–391.

[22] P.M. Kanade and L.O. Hall, "Fuzzy ants as a clustering concept", 22nd International Conference of the North American Fuzzy Information Processing Society, 2003, pp. 227-232.

[23] M. Jalali, "A new clustering approach based on graph partitioning for navigation patterns mining", International Conference on Pattern Recognition, 2008 Vol. 19, pp. 1-4.

[24] V. Sujatha and M. Punithavalli, "Improved User Navigation Pattern Prediction Technique from Web Log Data", International Conference on Communication Technology and System Design, 2012, Vol. 30, pp. 92–99.

[25] E. Tug, M. Sakiroglu, and A. Arslan, "Automatic discovery of the sequential accesses from Web log data files via a genetic algorithm", Knowledge-Based Systems, 2006, Vol. 19, Issue 3, pp. 180–186.

[26] S. Kim and B. Zhang, "Genetic mining of HTML structures for effective Web-document retrieval", Journal Applied Intelligence, 2003, Vol. 18 Issue 3, pp. 243 - 256.

[27] N. Picarougne, " Geniminer: Web Mining with Genetic-Based Algorithm ", NEC Research Institute CiteSeer, 2002.

[28] A. Abraham and V. Ramos, "Web usage mining using artificial ant colony clustering and genetic programming", Evolutionary Computation, 2003, Vol. 2, pp. 1384-1391.

[29] B. Zhou, S.C. Hui, and K. Chang, "An Intelligent recommender system using sequential Web access patterns",

IEEE Conference on Cybernetics and Intelligent Systems, 2004 Vol. 1, pp. 393-398.
[30] R. Burke, "Hybrid recommender systems: survey and experiments", User Modeling and User-Adapted Interaction, 2002, Vol. 12, Issue 4, pp. 331 – 370.
[31] H. Ishikawa, "An intelligent Web recommendation system: A Web usage mining approach", Proceeding ISMIS '02 Proceedings of the 13th International Symposium on Foundations of Intelligent Systems, 2002, Vol. 13, pp. 342-350.
[32] B. Mobasher, H. Dai, T. Luo, Y. Sun, and J. Zhu, "Integrating Web usage and content mining for more effective personalization", in First International Conference on Electronic Commerce and Web Technologies, 2000, Vol. 1875, pp. 165-176.
[33] R. R. Sarukkai, "Link prediction and path analysis using Markov chains", Computer Networks, 2000, Vol. 33, Issues 1–6, pp. 377–386.
[34] M. S. Chen, J. S. Park, and P. S. Yu, "Data mining for path traversal patterns in a Web environment", Distributed Computing Systems, Proceedings of the 16th International Conference on, 1996, pp. 365-392.
[35] R. Cooley, B. Mobasher, and J. Srivastava, "Data preparation for mining world wide Web browsing patterns", Journal of Knowledge and Information Systems, 1999, Vol. 1, Issue 1 , pp. 5-32.
[36] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic personalization based on Web usage mining", Communications of the ACM, 2000, Vol. 43 Issue 8. pp. 142-151.
[37] F. Bonchi, "Web log data warehousing and mining for intelligent Web caching", Data and knowledge engineering, 2001, Vol. 39, Issue 2, pp. 165–189.
[38] S. Schechter, M. Krishnan, and M. D. Smith, "Using path profiles to predict HTTP requests", Proceedings of the Seventh International World Wide Web Conference, 1998, Vol. 30, Issues 1–7, pp. 457–467.
[39] J. Dean, and M. R. Henzinger, "Finding related pages in the world wide Web", WWW '99 Proceedings of the eighth international conference on World Wide Web, 1999, Vol. 31 Issue 11-16, pp. 1467-1479.
[40] M. Chen, A. S. LaPaugh, and J. P. Singh, "Predicting category accesses for a user in a structured information space", in 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2002, Vol. 25, pp. 65-72.
[41] S. Brin, and L. Page, "The anatomy of a large-scale hypertextual Web search engine", Seventh International Conference on World Wide Web, 1998, Vol. 30 Issue 1-7, pp. 107-117.
[42] F. Qiu, and J. Cho, "Automatic identification of user interest for personalized search", in 15th International Conference on World Wide Web, 2006, Vol.15, pp. 727–736.
[43] M. Eirinaki, and M. Vazirgianis, "Web mining for Web personalization", Journal ACM Transactions on Internet Technology (TOIT), 2003, Vol. 3 Issue 1, pp. 1-27.
[44] K. R. Suneetha, and R. Krishnamoorthi, "Identifying User Behavior by Analyzing Web Server access log file", IJCSNS International Journal of Computer Science and Network Security,2009, Vol.9 No.4, pp. 327-332.
[45] F. M. Facca, and P. L. Lanzi, "Mining interesting knowledge from Weblogs: a survey", Data & Knowledge Engineering, 2005, Vol. 53, Issue 3, pp. 225–241.
[46] D. Dong, "Exploration on Web Usage Mining and its Application", International Workshop on Intelligent Systems and Applications, 2009, pp.1 - 4.
[47] G. T. Raju, and P. Sathyanarayana, "Knowledge discovery from Web Usage Data: Complete Preprocessing Methodology", International Journal of Computer Science and Network Security (IJCSNS), 2008, Vol.8 No.1, pp. 179-186.
[48] B. Liu, "Web Data Mining Exploring Hyperlinks, Contents, and Usage Data", Springer Series on Data-Centric Systems and Application, 2011.
[49] D. Tanasa and B. Trousse, "Advanced data preprocessing for inter sites Web usage mining", IEEE Intelligent Systems, 2004, Vol. 1912, pp. 59-65.
[50] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery: An overview", ACM KDD, 1994.
[51] T. Nguyen, D. Phung, B. Adams, T. Tran, and S. Venkatesh, "Classification and Pattern Discovery of Mood in Weblogs", Advances in Knowledge Discovery and Data Mining, 2010, Vol. 6119, pp. 283-290.
[52] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", 20th VLDB Conference, 1994.
[53] K.P. Joshi, A. Joshi, and Y. Yesha, "On using a warehouse to analyze Web logs", in Distributed and Parallel Databases, 2003, Vol. 13, Issue 2, pp. 161-180.
[54] J. Han and M. Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann Series in Data Management Systems, 2001.
[55] X. Huang and N. Cercone, "Comparison of interestingness functions for learning Web usage patterns", CIKM '02 Proceedings of the eleventh international conference on Information and knowledge management, 2000, pp. 617-620.
[56] M. Martínez, G. Vargas, A. Dorado, and M. Millán, "Mining fuzzy association rules for Web access case adaptation", AWIC'03 Proceedings of the 1st international Atlantic web intelligence conference on Advances in web intelligence, 2003, pp. 73-82.
[57] A.I. Schein, A. Popescul, and L.H. Ungar, "Methods and Metrics for Cold-Start Recommendations", SIGIR '02 Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, 2002, pp. 253-260.
[58] S. Sahebi, and W. Cohen, "Community-Based Recommendations: a Solution to the Cold Start Problem ", In: Workshop on Recommender Systems and the Social Web (RSWEB), held in conjunction with ACM RecSys'11, 2011.
[59] K. Shyong, and J. Riedl, "Shilling recommender systems for fun and profit", 13th international conference on World Wide Web, 2000, pp. 393–402.
[60] R. Burke, B. Mobasher, R. Bhaumik, and C. Williams, "Segment-based injection attacks against collaborative filtering recommender systems", Fifth IEEE International Conference on Data Mining, 2005, pp. 577–580.
[61] T. H. Kim, and S.B. Yang, "An Effective Recommendation Algorithm for Clustering-Based Recommender Systems", Springer-Verlag Berlin Heidelberg, 2005, pp. 1150-1153.
[62] M. A. Ghazanfar, and A. Prugel-Bennett, "Fulfilling the Needs of Gray-Sheep Users in Recommender Systems, A

Clustering Solution", International conference on information systems and computational intelligence, 2011, pp. 1-6.
[63] M.d. Gemmis, "Preference Learning in Recommender Systems", European Conference on Machine Learning and Principles and Practice of knowledge Discovery in Databases, 2009, pp. 41-55.
[64] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis", Journal of the American Society for Information Science, 1990, vol.41, no. 6, pp. 391–407.

Maryam Jafari received her M.Sc. degree in computer engineering at Islamic Azad University, Zanjan Branch, Zanjan, Iran in 2013. She is a lecturer in Department of Computer, Novin Higher Education Institute, Ardabil, Iran. Her research interest spans the area of Web Mining issues, especially in the field of web usage mining and pattern discovery. She published some papers in International Journals and a book in this domain.

Farzad Soleymani Sabzchi graduated in M.Sc. degree in computer engineering at Islamic Azad University, Zanjan Branch, Zanjan, Iran in 2013. He works as a lecturer in Department of Computer, Novin Higher Education Institute, Ardabil, Iran. His primary research interest is the area of Web Mining issues, especially in the field of web usage mining, Web recommender systems and personalization. He has some papers in International Journals and published a book in this domain.

Amir Jalili Irani graduated in M.Sc. degree in computer engineering at Islamic Azad University, science and research branch, tehran, Iran in 2011. He works as a faculty in Sama technical and vocational training college, Ardebil Branch, Islamic Azad University, Ardebil, Iran. His primary research interest is the area of intelligent system.