International Journal of Industrial Engineering & Production Research (2010) pp. 137-146

September 2010, Volume 21, Number 3

International Journal of Industrial Engineering & Production Research

ISSN: 2008-4889

http://IJIEPR.iust.ac.ir/

Use of Semantic Similarity and Web Usage Mining to Alleviate the Drawbacks of User-Based Collaborative Filtering Recommender Systems

Reza Samizadeh* **& Babak Ghelichkhani**

Reza Samizadeh, *Assistant Professor, Department of Industrial Engineering, Alzahra University, Tehran, IRAN*
Babak Ghelichkhani, *Department of Information Technology, Tarbiat Modares University, Tehran, IRAN,*

KEYWORDS

User-Based Collaborative Filtering, Hybrid Recommender system, Semantic Similarity, Scalability, Sparseness, New-Item Problem.

ABSTRACT

One of the most famous methods for recommendation is user-based Collaborative Filtering (CF). This system compares active user's items rating with historical rating records of other users to find similar users and recommending items which seems interesting to these similar users and have not been rated by the active user. As a way of computing recommendations, the ultimate goal of the user-based collaborative filtering is recommending items with the high accuracy and coverage degree. Nevertheless, some famous limitations are obstacles to meet them. They are Scalability, Sparseness and new item problems. Scalability problem can be handled with the use of Data Mining techniques like clustering. However, use of this technique often leads to the lower recommendation accuracy. Nevertheless, two other problems still remain. Involving Semantic knowledge can increase the performance of recommendation in sparseness and New-Item Problem conditions as well. This paper presents a new approach to deal with the drawbacks of user-based CF systems for web pages recommendation by Combination of Semantic Knowledge with Web Usage Mining (WUM). Semantic knowledge of web pages are extracted and subsequently incorporated into the navigation patterns of each cluster which obtained from clustering the access sessions to get the Semantic Patterns of each cluster. The cluster with the most relevant semantic pattern is chosen with the comparison of semantic representation of the active user session with the semantic patterns and the proper web pages are recommended based on a switching recommendation engine. This engine recommends a list of appropriate recommendations. Results of the implementation of this hybrid web recommender system indicates that this combined approach yields better results in both accuracy and coverage metrics and also has a considerable capability to handle collaborative filtering recommender system for its typical shortcomings.

1. Introduction

Although one of the most important features of the World Wide Web is its huge information space, this feature has leads to some problems for its users. Most of the web structures are huge and complicated and users often become frustrated to find their desired information, services or products. This problem is very crucial for e-commerce websites, because they lose their users (customers) easily. Web Personalization is a key to remedy this problem. The objective of a web personalization system is to provide users with the information they want or need without expecting from them to ask for it explicitly [1]. As a way of Web personalization, Web recommender systems aim to automatically recommend hyperlinks that seem

* **Corresponding author. Reza Samizadeh**
Email: *rsamizadeh@alzahra.ac.ir ghelichkhani@modares.ac.ir*
Paper first received March. 07. 2009, and in revised form August. 27. 2010.

relevant to the users' interests in order to access to the necessary information on a huge website. This method of web personalization is implemented on web server and relies on the data gathered implicitly (history of browsing history which stored in server logs) or explicitly (questioners, explicit user ratings) from users which demonstrate their interests [2].

One of the most widely used approaches used to make recommendation is collaborative filtering. Based on this assumption which the users with similar previous behaviors (browsing history, history of purchases, etc.) have common interests, CF systems recommend items which preferred by the other users with similar interests. This similarity acquired by the comparison between historical records of the previous users ratings with the ratings of the current user (considering WWW, users who are visiting a web site). This user is called active or target user.

Traditional or user-based CF has been widely used in the area of web personalization and recommendation [3]. The goal of this system is specifying a set of users with the most similar item ratings to the active user. This set called its neighborhood. This neighborhood is then used for recommending the items that have the most value in this neighborhood.

Despite of the success of Traditional Collaborative filtering, these systems suffer from some well-known drawbacks [4]:

- *Scalability*: traditional user-based CF mainly uses *k-Nearest-Neighbor* classification approach which must be performed in the online phase of recommendation (recommendation time). This brings to tremendous load of computation in the recommendation time and makes the recommendation engine too slow. To solve this problem, involving web usage mining techniques (such as clustering and association rule mining) can be used to enhance the scalability. These techniques take a huge computation task and thus performed in the offline phase of Recommendation system.
- *Sparseness*: in the real world conditions, that there are many users, only a few items were visited in each user session. Therefore, the matrix that represents the user-item ratings becomes sparse. This sparseness causes items with few ratings get low recommendation value and consequently not be recommended.
- *New Item or First-Rater problem*: sparseness problem in its worst condition is very similar to another problem called new item problem. CF techniques are highly vulnerable to add new items. Because the new item has no rating by the users, has no recommendation value and therefore, it cannot be recommended.

This paper presents a hybrid web recommender system which addresses mentioned problems combining the features of content - based with collaborative recommendation. The problem of scalability is settled using web usage mining in collaborative component, and with utilizing the semantic knowledge extracted from items (web pages), the problems of new item and sparseness efficiently alleviated. This hybrid system clusters the user sessions extracted from the log files and generates the navigational patterns of users which demonstrate the common user navigational behaviors. The semantic features which extracted from domain specific anthologies subsequently combined with the navigation patterns to produce semantically enhanced navigational patterns which are called semantic patterns. Then, the recommendations are made with the use of navigational and semantic patterns with a switching engine. The results of the implementation of this hybrid recommender system show a significant improvement for foregoing drawbacks, and consequently improve the accuracy and coverage of the recommendation engine output.

The Following sections are organized as follows. In section 2, the necessary concepts of the collaborative filtering recommendation systems are presented. This section also include an overview of the important researches that attempted to solve the shortcomings of recommendation systems with the use of web usage mining and combination of usage and content mining with the use of keywords or semantic features of the contents of the web pages. In section 3, the architecture of the proposed hybrid recommender system is presented. Every component of this system is described in detail. In section 4, the results of the implementation of the system are evaluated and finally, in section 5, conclusions are presented and the possible future works are introduced.

2. Related Works

Formally, a collaborative filtering system is defined as follows: Given a set of all users $U = \{u_1, u_2, ..., u_m\}$ and a set of all items $I = \{i_1, i_2, ..., i_n\}$. Each user rates some items of set *I.* This record of item rating can be displayed by an n-dimensional vector. All of these vectors can be represented in the form of a matrix $M_{m \times n}$ which each of its rows represents one user's record and each column represents one particular item. Thus, each cell of the matrix *M,* for instance, $M_{p,q}$ is the value of item i_q which has been rated by the user u_p. This rating can be explicit or implicit. In the context of Web, each user session (each row of matrix *M*) is represented as a user transaction with a website in a predefined limited time. The ratings of usage data are generally implicit and determined as binary values (existence or non-existence of web pages in a session) or real number values based on the combination of factors like time spent on certain web pages and their sizes and the other more complicated factors [5]. For a given active user u_a, also called the target user (the user that the recommendation will be made for), the

task of a user-based collaborative recommendation engine is recommending a set of items which seems interesting to the active user, that is the prediction of the value of $M_{a,t}$ for items which have not been rated (have no weight) by the active user. In user based collaborative filtering, this recommendation is made with the comparison of the active (target) user with *k* users (in the domain of Web, it means *k* user sessions) with the most similar item rating behavior and subsequently use some methods to determine items with the most value from the items which these k users have rated. These items are then recommended in order, according to their determined recommendation values. This method of collaborative filtering is called *k-nearest neighborhood* or *traditional collaborative filtering*[6].

First attempts to solve foregoing shortcomings, which focused on solving the scalability problem, presented collaborative recommendation systems based on a web usage mining technique [7-9] or combination of several web usage mining techniques [10, 11] These systems generally consist of two phases: offline pattern extraction and online recommendation [12]. In offline recommendation phase, the web usage mining techniques are applied to reveal the hidden navigation patterns of users that stored in the web server logs. In the online phase, the current session of active user is compared with these navigational patterns with some similarity measures and consequently recommends items are determined. Although computation of extracting the navigational patterns with the use of web usage mining adds a heavy burden of computing to offline phase and consequently makes it slow, due to the comparison of active user current session with relatively a few number of navigational patterns, the online phase of recommendation is much faster than the traditional CF with the use of finding k-Nearest-Neighbors In the online phase[13]. Although the scalability problem can be handled by the use of web usage mining, other problems approximately remain.

To address the new item and sparseness problems, some research activities proposed hybrid recommender systems. In some of these systems, in addition to usage data, the sources of knowledge extracted from content of web page[14-17], structure of web sites[12], are combined to alleviate the sparseness, new-item and cold-start[1] problem. Most of these systems proposed a content-collaborative recommender system which combines content mining with web usage mining. Generally, in these works, two approaches are used. One approach which is known as feature combination method, important information of content in various forms (e.g. keywords, character N-grams and word N-grams) are extracted from the content of web pages and combined into the patterns or profiles concluded from web usage mining results to construct one recommendation set. Another approach constructs one set of content profiles and one set for content profiles. These two sets subsequently compared with the user active session and items recommended in a switching manner.

Melville et al. trained the content based prediction on the rows of the actual user-item rating matrix (which is a sparse matrix) and consequently made a full matrix (all of the cells are non-zero). Then, they use collaborative filtering on this pseudo user-ratings matrix to make recommendations. The results of their experiments show significant improved predictions for a CF-based recommender system by alleviating the sparseness and first-rater problems[16].

In newer work, [14] incorporate the content features of web pages in the form of character N-grams with the navigation patterns generated by clustering users' session. This technique yields better results of classification of users' sessions into extracted clusters and produced better prediction of future requests of active users in comparison of using just clustering data mining technique.

A hybrid recommender system which combined the usage, content and structure data was presented by[12].They clustered web pages based on their contents and used these clusters to deconstruct each user session to sub-sessions with the relevant content called missions which represent a consistent goal. These missions are subsequently clustered to generate navigational patterns, and augmented with their linked neighborhood and ranked based on resource connectivity. These new augmented navigational patterns are used for the recommendation engine. The session of the active user is compared with these patterns and consequently the recommendations are made with the most relevant pages in the matched cluster which is represented with the corresponding pattern.

In spite of the utility of these hybrid recommender systems, use of keyword based content mining has some drawbacks. They don't gain an understanding of the relationship or properties of the objects (text, picture, movie, etc.) of web pages which can be extracted by the semantic information of that objects. Using keyword based methods have some drawbacks: This is not a good approach for make relation between non-text objects. These systems make a plenty of noise and this problem becomes more if the focus of content mining is not based on appropriate tags. In addition, the representation of documents based on keywords doesn't have ability to compare documents at a deeper semantic level. For instance, words glad and happy are no alphabetically similar, but are almost semantically same[18].

A few efforts provided the hybrid content-collaborative recommendation systems which combined usage data

[1] Cold start problem refers to the condition that an active user has not rated any item and consequently, the pure collaborative filtering system can not recommend any item.

with the semantic knowledge extracted from the contents of web pages. Paulakis et al. [19] presented a hybrid web personalization system called SEWeP. Subsequently they applied document clustering with semantic similarity measures for clustering documents. The result of these clustering is finding the similar documents with similar semantic contents. Furthermore, they applied association rule mining on the original records of web server logs. In the online phase of recommendation, in addition to recommendation based on the matching of the active user's navigational behavior with the rules extracted in the offline phase, the similar pages with similar semantic contents will be recommended to that user. The primary innovation of their work was C-logs. They enhanced each record of web server logs with keywords from taxonomy and generated C-logs (concept-logs). Data mining techniques w ere described above performed on the C-logs.

Xin Jin et al. proposed a hybrid web recommender systems based on item-based collaborative filtering to handle sparseness and first rater problem [20]. They used a combined similarity measures which computed both user ratings and document semantic similarity to recommend the most relevant web pages to the current user. Experiments in the paper prove ability to make recommendations with high quality.

In [15], since the similarity between users' item rating is computable if they have similar item ratings (considered as a shortcoming), a hybrid content-collaborative recommendation system was proposed that instead of using similarity between users' item rating, semantic similarity between semantic profiles of users are computed for users clustering in order to neighborhood generation. Their system was implemented on a movie recommendation scenario and the results showed its effectiveness.

3. System Architecture

In this work, a collaborative system has been designed to recommend the proper web pages to the users of a web site which has a significant performance in the sparseness and new-item problem conditions, while has the ability to handle he scalability problem. This system is comprised of two main phases: an offline phase which include web log preprocessing and use of web usage mining and semantic knowledge to generate the semantic patterns and an online phase which a recommendation engine is used that takes the output of offline phase as input and produces a relevant recommended list of web pages to the active user.

Fig. 1 illustrates the overall design of the system. One of the innovative features of the system is the incorporation of semantic features with navigational patterns to construct semantic patterns used for comparing with semantic representation of active user's current session. Each pattern demonstrates the common contents (in a semantic level) which are interesting to a set of users with similar navigational pattern (similar web page visits). In addition, to handle the new added item problem (provide an opportunity for new added web pages of the web site to be able to be recommended by the recommendation engine), all new web pages are classified into the extracted semantic patterns using *1-Nearest-Neighbor* classification approach(each new added web pages is classified into one semantic pattern). Another innovation (in the online phase of recommendation) is the use of switching recommendation engine based on the weights of the web pages in the mean vector of matched cluster (the cluster that is represented with the semantic pattern which has the most semantic similarity with the semantic representation of the current session) which might be recommended to the active user (will be described in section 3.4).

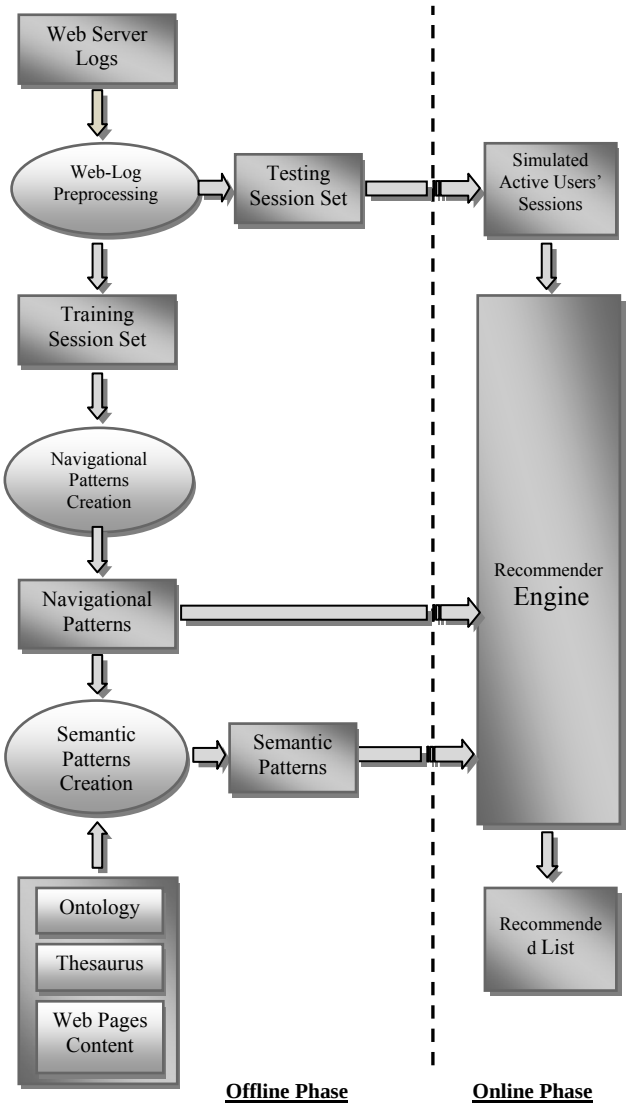

Fig. 1. System architecture

The details of each component of this system are given in the following.

3.1. Web Log Preprocessing

In general, all the requests of each user are recorded in the web server logs, in the form of a text file. The purpose of Web log Preprocessing is reformatting the web server logs to indentify all users' access sessions. For the purpose of Web log preprocessing, the entries of the web server logs must be cleaned, the users must be differentiated and the sessions must be identified. The preprocessing method used in [21], with session duration threshold of 30 minutes, was chosen for the proposed recommendation system. The outcome of this component can be represented as a matrix *UP* which each row of this matrix represents a user who visits some pages in a session time and each column represents a particular web page. This set is subsequently divided to training and testing sets. The training set is used in offline phase for extracting navigational patterns of users and constructing the semantic patterns. The testing set is used for constructing simulated active user sessions which used in the online phase.

3.2. Navigational Patterns Creation

In the next step, the preprocessed web logs are used as the input data for clustering data mining technique. This technique can discover important user segments that demonstrate common navigational behavior (page visits) among a set of users. A typical method for clustering user sessions is vectorization. Given a set of all web pages visited by all users in web server logs, $P = \{p_1, p_2, ..., p_n\}$ (each page can be represented by its associated URL), each session *s* can be represented as an *n*-dimensional vector over the space of all the web pages, $s = \{w(p_1, s), w(p_2, s), ..., w(p_n, s)\}$, where $w(p_i, s)$ is the weight of the (*i*)th web page visited in session *s*. This representation can facilitate the clustering operation of users' sessions. The weight $w(p_i, s)$ can be determined binary (value 0 for non-existence or 1 for existence) or real value which is a combination of some other metrics[5]. In this paper, $w(p_i, s)$ is defined as the interest degree of a particular user to the page p_i, which is the harmonic mean of Frequency and Duration to represent this interest, and shown as Eq. (1) [14]:

$$Interest(p_i) = \frac{2 \times Frequency(p_i) \times Duration(p_i)}{Frequency(p_i) + Duration(p_i)} \quad (1)$$

The *K*-means algorithm is applied as the method for clustering the vectored sessions. WEKA machine learning toolkit [22] is used to perform the K-means algorithm and the Euclidean distance is adopted as the distance measure used for clustering[23]. The optimal number of clusters are determined with two measures of cluster compactness and cluster separation proposed in [24]. The result of session clustering is a set of clusters which contain similar sessions.

As mentioned before, use of the clustering technique in the offline phase of the recommendation system decreases the load of computing in the online phase and consequently improves the scalability of the collaborative recommender system. In the scenario of the clustering sessions for collaborative filtering (in pure clustering based recommendation system), the result of session clustering is a set of clusters $C = \{c_1, c_2, ..., c_k\}$. Each $c_i (1 \leqslant i \leqslant k)$ contains a fraction of all the training sessions set. Considering a cluster *c*, a mean vector m_c is generated for this cluster (each member of this vector is the average weight of each web page across all sessions of cluster *c*). The vector m_c represents a user navigational pattern which demonstrates web pages that tend to be visited based on the common interests or needs. A weight threshold $w_{\min}$ is determined for removing the web pages that have not such a value to be recommended to the active user (recommendation of these web pages often leads to low accuracy and coverage). The web pages which have a mean weight higher than $w_{\min}$, formed the navigational pattern of each cluster. subsequently, an active session is compared with the navigational patterns, instead of all users' sessions and a set of web pages that form the navigational pattern are chosen for recommended list based on their recommendation score[25].

It is obvious that the pure clustering based recommendation system doesn't recommend web pages which have mean weights lower than $w_{\min}$. The proposed system in this paper has ability to recommend these pages efficiently (see section 3.4).

3.3. Semantic Profiles Creation

Semantic profiles creation process aims at incorporating the semantic features of web pages contents into the navigational patterns. This is done by the extraction of the most important keywords of each page. As mentioned before, the semantic similarity of the semantic profiles with the semantic representation session of active users is calculated to determine the most relevant semantic pattern. From the cluster which the semantic pattern represents, the most relevant web pages to the active user are recommended. The method of recommending web pages will be described in section 3.4. At the first step, content of each web page must be represented with keywords (from ontology) that characterize that web page in the best manner. To extract these keywords, the web page content must be cleaned that includes removing HTML, XML, SGML tags and all punctuations. In addition, since a few web pages which are used for experiments have a mixture content of both English and Persian languages (Since

the web site used for experiments are accessible in both English and Persian, English version of the rest of the web pages are in access.), the Persian content of these web pages must be translated into English before keyword extraction. It must be stressed here that due to the use of semantic similarity, instead of keyword based similarity, this translation doesn't affect the accuracy of semantic similarity as described in section 2.

A famous technique for representing each document is the representation in the bag-of-word (BOW) format[18]. A bag of word is a set of weighted terms that characterize document in an optimal manner. So, the similarity between two documents can be computed based on their BOWs. A number of similarity measures[26-28] have been proposed to compute the similarity. Some recent works were on the semantic similarity relatedness of two terms or documents based on the domain specific-ontology. In this paper, the notions of extracting the semantic features of documents and measuring the semantic similarity between two sets of words will not be discussed in detail. In [29] the concept of bag-on-concept was introduced for extracting the critical keywords of each web page, instead of bag-of-word concept. Using this concept, WordNet [30] as the ontology (an ontology of university is accessible in [32] and the methods used for extracting semantic features and semantic similarity measures which were presented in [18], our semantic patterns are created and the semantic similarity between semantic patterns and semantic representation of active sessions are computed.

3.4. The Recommendation Engine

The task of this component is receiving the user's recent access session s_a and producing a recommended list for the active user u_a.

The recommendation priority is determined based on the score calculated (using recommendation engine) for each web page that have not been visited by the active user. It is so clear that the inability of cluster (generally, web usage mining) based CF system for recommendation the web pages with mean value lower than $w_{\min}$, causes a lower performance for this type of recommendation system. These pages are either web pages with the mean weight under the threshold $w_{\min}$ or the web pages which added newly to the web site which has no weight. The hybrid recommendation engine proposed in this paper uses a novel switching technique, based on the mean weight of the web pages in a cluster (weight of the web page in the mean vector m_c of the cluster). This technique is based on the following assumptions. The pure clustering(in general, web usage mining) based collaborative filtering is highly effective for a high dense user-item matrix [4] (most of the cells of this matrix is non-zero, and consequently, most of web pages in each cluster have a mean weight which causes to be recommended by the recommendation engine). For web pages which are not visited frequently or added newly to the web site, most of the cells of their corresponding column is zero or have low values which causes these web pages are not recommended to the active user. The recommendation engine in this system, uses a switching technique based on the weight of web page in the mean vector of the matched cluster and its relation to the threshold $w_{\min}$. Using this approach, the web pages which cannot be recommended by the pure web usage mining based CF recommendation system, have chance to be recommended .

The process of the recommendation engine of the system is in the following order. Given an active session *as*, the semantic pattern of active session sp_a is compared with all semantic patterns *SP* and the matched semantic pattern (the most similar pattern) is selected. The web pages included in mean vector m_c of the cluster *c* which is represented by the matched semantic pattern sp_c are used as the input of the recommender engine (in contrast to pure cluster based recommendation system in which the web pages included in navigational pattern are used). Given $P = \{p_1, p_2, ..., p_n\}$ as the set of all web pages that are currently in a web site, and P' as a subset of *P* ($P(|P'| = m \leq n)$ which are web pages included in training set (exist in web logs), considering the matched cluster *c* and its $w_{\min}$, each web page *p* belongs to one of the following sets:

$$R = \{p_i \varepsilon P' (1 \leq i \leq m); mw(p_i, c) \geq w_{\min}\}$$
$$R' = \{p_i \varepsilon P' (1 \leq i \leq m); 0 < mw(p_i, c) < w_{\min}\}$$
$$R'' = \{p_i \varepsilon (P - P') (1 \leq i \leq n)\}$$

In above sets, $mw(p_i, c)$ is the mean weight of web page p_i in the mean vector m_c.

Thus, given an active session *as*, *Rs (p)*, the recommendation score for each web page *p* in the matched cluster *c* is computed as Eq. (2):

$$Rs(p) = \begin{cases} mw(p,c) & p \varepsilon R \\ \beta \times mw(p,c) + (1 - \beta) \times ss(p,as) & p \varepsilon R' \\ ss(p,as) & p \varepsilon R'' \end{cases} \quad (2)$$

In the equation above, $mw(p,c)$ is the mean weight of web page *p* in the mean vector m_c and the weight $ss(p,as)$ is the semantic similariry measure between semantic representation of page *p* and semantic representation of active session *as*. This score is computed for the web pages that have not been visited

(rated) by the active user (there are not in the current session). Apparently, for the visited web pages by the active user, this score is 0. In the equation above, for each $p \varepsilon R$, the recommendation score is determined as the pure clustering based recommendation method. For each $p \varepsilon R'$, a linear combination measure is computed to give score to web page *p*. In this condition, the parameter β is used to give weight to both $mean\_weight(p,c)$ and $sem\_sim(p,as)$ in a linear manner. For each $p \varepsilon R''$, since there is no item-rating for these web pages, the pure similarity measure is computed for determining the score of web page *p*. It must be stressed that the priority of web pages recommendation is based on the score determined with this equation. The optimal values of $w_{\min}$ and β will b determined in experiments (section 4.2).

4. Experimental Results and Evaluation

To evaluate the performance of the recommender system, the framework proposed in this paper was implemented on the Website of Tarbiat Modares University of Tehran (TMU). This selection for the experiment provided the data set that allows analyzing both web logs and web pages contents. Entries of two months of the log of TMU were used as experimental data. The first month which produced a 251 MB file with 946076 access entries including 728 web pages, was used as the training set and the second which produced a 203 MB with 765153 access entries including 691 web pages, was used as the testing set. The applied clustering method described earlier was applied on the testing sessions set extracted from the web logs and produced 23 clusters.

4.1. Methodology

As mentioned before, half of the web logs were chosen for simulating the active sessions as testing set. Due to the absence of the new added web pages in the testing set (not including in the user-rating matrix), another approach must be chosen to evaluate the performance of the recommendation system when new added web pages are recommended to the active user. For this purpose, after implementing the system on the training and testing set, the system implemented again on altered training and testing sets. To construct this altered sets, some items from all the training and testing sessions of web logs were removed (value of all cells in some columns were set zero) and the recommendation process was made on these altered sessions set.

Evaluation method of the system is as follows. Each testing session (*ts)* of testing set is divided into two parts. The first *n* web pages of session *ts* is used as the input of the recommendation engine and the second part is simulated as the future requests (page visits) which are compared with the output of the recommendation system. Number *n* determines the maximum window size *w* ($w \le n$) as the input. According to [13], the size 4 is set for *w* (*w*=4). This size represents the last *w* pages in the first part of session that called active session window (*asw*). This window is used as the input of the recommender engine. For testing session *ts* with size $n \le 4$, active session window *asw* with size n-1 were chosen. The recommendation engine takes *asw* and the recommendation threshold μ as the input and generates a recommended list which denoted by $R(asw,\mu)$ This list is compared with the second part of session *ts* (*es*) with two metrics, accuracy and coverage [13], which are defined as Eq. (3) and Eq. (4):

$$accuracy(Rasw,\mu)) = \frac{|R(asw,\mu \cap es|}{|R(asw,\mu)|} \quad (3)$$

$$\text{cov}erage(Rasw,\mu)) = \frac{|R(asw,\mu \cap es|}{|es|} \quad (4)$$

Accuracy measures the ability of recommender system to produce accurate recommendations and coverage metric determines the ability for recommending all web pages that will be likely visited by the active user. As you can see, both of the metrics are between 0 and 1.

In order to evaluate the performance of the system, experiments carried out on two different original and altered sets and both accuracy and coverage of the recommender engine for recommendation threshold μ from 0.1 to 0.9 were measured.

4.2. Results

As consider Eq. (2), used in the recommendation engine (section 3.4), the parameters $w_{\min}$ and β must be determined in order to lead our system to the best performance ($w_{\min}$ and $\beta \in \{0,1,0,2,...,0.9\}$). To determine the most optimal $w_{\min}$ (considering both recommendation accuracy and coverage), the pure clustering based recommendation system was implemented according to [31]. This implementation determined $w_{\min} = 0.5$ as the optimal value for sum of accuracy and coverage. (Although the recommendation accuracy becomes better at higher $w_{\min}$, the recommendation coverage decreases). To find the optimal β, all the experiments for recommendation accuracy and coverage on the thresholds μ were implemented for each β from 0.1 to 0.9 as well. The consequent results showed the optimal $\beta = 0.3$.

It is mentioned earlier that the recommendation system is implemented on two sets of original and altered

session sets. Fig. 2 and Fig. 3 show the accuracy and coverage of the system for the original set. furthermore, association rule based recommender systems were also implemented (because of wide range of use for recommendation systems and having the best performance in overall accuracy and coverage metrics in web usage mining based recommender systems [31] with the same data sets(original training and testing sets). The results were also depicted in Fig. 2 and Fig. 3 for comparison.

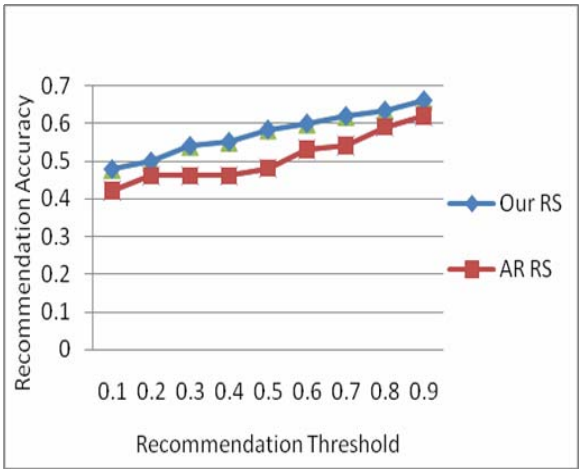

Fig. 2. Recommendation accuracy comparison: Our system vs. association rule-based system ($w_{\min} = 0.5 and \beta = 0.3$)

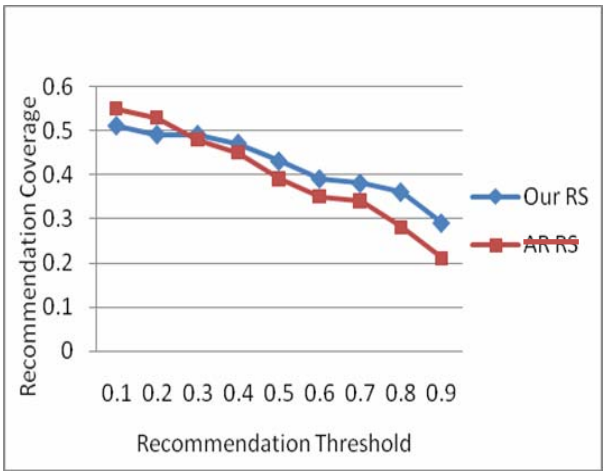

Fig. 3. Recommendation coverage comparison: our system vs. association rule-based system ($w_{\min} = 0.5 and \beta = 0.3$)

As it is obvious in Fig. 2, our recommender system results an overall better accuracy(in all of the threshold values), but Fig. 3 depicts better results for recommendation coverage of association rule based recommendation system for threshold values of 0.1 and 0.2 and better results for our system in another thresholds (0.3 to 0.9). According to [31], it is notable that the association rule mining based recommendation system has the best performance in coverage metric among the web usage mining based CF recommendation systems.

Fig. 4 and Fig. 5 show the accuracy and coverage of our system for the new added web pages condition (new web pages have been added to the website and can be recommended to the active user). As described earlier, these results are implementation of our system on the altered training and testing session sets. As it is depicted in Fig. 4 and Fig. 5, the recommendation accuracy and coverage of our system have slightly decreased in comparison to Fig. 2 and Fig. 3. This result is because of involving new added web pages to the output of the recommendation engine.

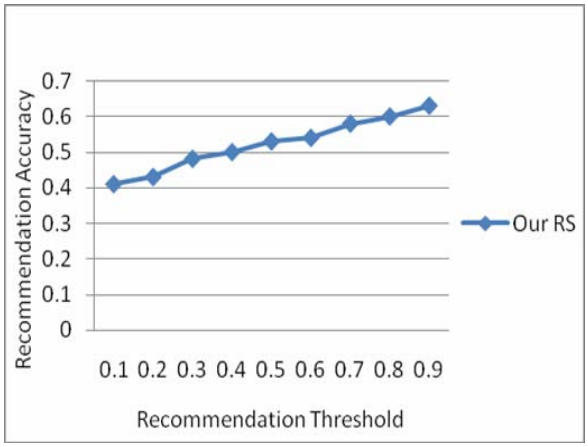

Fig. 4. Recommendation accuracy for new added web pages condition ($w_{\min} = 0.5 and \beta = 0.3$)

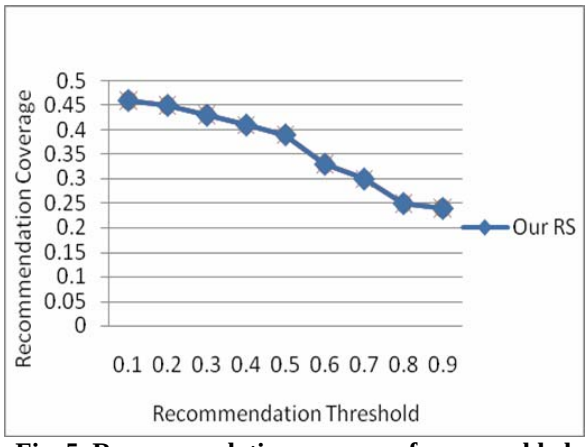

Fig. 5. Recommendation accuracy for new added web pages condition ($w_{\min} = 0.5 and \beta = 0.3$)

5. Conclusions

With the use of clustering of the sessions which have been extracted from the web logs, the user navigational patterns are constructed which represent visitors segments based on common interests. The semantic knowledge of the web pages is extracted and subsequently incorporated to these patterns and form semantic patterns. These patterns are used in the online

phase as the input of the recommendation engine which uses an innovative switching technique for generating a recommended list to the active user. With the use of these semantic patterns and the active sessions, the recommender engine determines the most relevant cluster to the active session with comparing all the semantic patterns with the semantic representation of active user session. Subsequently, based on the weight of the web page in the mean vector of the matched cluster represented by the corresponding semantic pattern, Recommendation engine chooses different semantically enhanced combined measures for scoring the possible future web page visits of the active user and recommend those web pages that satisfies a threshold score as a recommended list. The results of the implementation of this system show that the proposed hybrid recommender system improves the accuracy and coverage of recommendation. This improvement is more considerable when the new item problem addressed.

6. Future Works

More research may be done in the following as ways to improve the proposed hybrid system. Applying association rule mining on the navigational patterns captures togetherness of the web pages in each cluster, one of the future work could be the detection of association rules among web pages to see whether can improve the recommendation accuracy and coverage. Another area of future work is improving the semantic similarity measure for computing the similarity between patterns and recommendation of web pages. In this paper, the extracted semantics was incorporated into the navigational patterns to generate the semantic patterns. Another ways can be utilized to integrate semantic knowledge with web usage mining. One of them is constructing a full-rated matrix with the combination of semantic knowledge and web usage mining techniques, which may yield better results in the sparseness and new item problem conditions.

References

[1] Mulvenna, M.D., Anand, S.S., Büchner, A.G., *Personalization on the Net using Web Mining: introduction*. 2000, ACM New York, NY, USA.

[2] Nasraoui, O., *World Wide Web Personalization*. 2005, Citeseer.

[3] Schafer, J.B., et al., *Collaborative Filtering Recommender Systems*. Springer 2007, P. 291.

[4] Burke, R., *Hybrid Web Recommender Systems*. Springer 2007, P. 377.

[5] Kim, H., Chan, P.K., *Implicit Indicator for Interesting Web pages*. 2005, Citeseer. pp. 270-277.

[6] Anand, S.S., Mobasher, B., *Intelligent Techniques for Web Personalization*. Springer 2005, pp. 1-36.

[7] Mobasher, B., Cooley, R., Srivastava, J., *Automatic Personalization Based on Web Usage Mining*. 2000, ACM New York, NY, USA.

[8] Perkowitz, M., Etzioni, O., *Towards Adaptive Web Sites: Conceptual Framework and Case Study*. 2000, Elsevier. pp. 245-275.

[9] Spiliopoulou, M., *Web Usage Mining for Web Site Evaluation*. 2000, ACM. pp. 127-134.

[10] Plasse, M., et al., *Combined use of Association Rules Mining and Clustering Methods to Find Relevant Links Between Binary Rare Attributes in a Large Data Set*. 2007, Elsevier. pp. 596-613.

[11] Lazcorreta, E., Botella, F., A. Fernandez-Caballero, *Towards Personalized Recommendation by Two-Step Modified Apriori Data Mining Algorithm*. 2008, Elsevier. pp. 1422-1429.

[12] Li, J., Zaiane, O.R., *Combining usage, Content, and Structure Data to Improve Web Site Recommendation*. Springer 2004, pp. 305-315.

[13] Mobasher, B., et al., *Effective Personalization Based on Association Rule Discovery from Web Usage Data*. 2001, ACM New York, NY, USA. pp. 9-15.

[14] Liu, H., Keselj, V., *Combined Mining of Web Server Logs and Web Contents for Classifying user Navigation Patterns and Predicting Users' Future Requests*. 2007, Elsevier. pp. 304-330.

[15] Lops, P., Degemmis, M., Semeraro, G., *Improving Social Filtering Techniques Through WordNet-Based user Profiles.* Springer 2007. P.268

[16] Melville, P., Mooney, R.J., Nagarajan, R., *Content-Boosted Collaborative Filtering for Improved Recommendations*. 2002, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. pp. 187-192.

[17] Kazienko, P., Adamski, M., *AdROSA - Adaptive Personalization of Web Advertising*. 2007, Elsevier. pp. 2269-2295.

[18] Thiagarajan, R., Manjunath, G., Stumptner, M., *Computing Semantic Similarity Using Ontologies*. 2008, HP Labs Technical Report HPL-2008-87.

[19] Paulakis, S., et al., *SEWeP: a Web Mining System Supporting Semantic Personalization*. Springer 2007, pp. 552-554.

[20] Mobasher, B., Jin, X., Zhou, Y., *Semantically Eenhanced Collaborative Filtering on the Web*. Springer 2004, pp. 57-76.

[21] Baglioni, M., et al., *Preprocessing and Mining Web Log Data for Web Personalization*. Springer 2003. pp. 237-249.

[22] Bouckaert, R.R., et al., *WEKA Manual for Version 3-6-0*. 2008.

[23] Meng, X.M., Chen, H.P., Zhang, T., *Study to Web Transactions Clustering Algorithm on WEKA*. 2009, China Aerospace Science & Industry Corporation, P. O. Box 142 Beijing 100854 China. pp. 1332-1334.

[24] He, J., et al., *On Quantitative Evaluation of Clustering Systems*. 2003. P. 134.

[25] Mobasher, B., *Data Mining for Web Personalization*. Springer 2007, P. 90.

[26] Song, W., Li, C.H., Park, S.C., *Genetic Algorithm for Yext Clustering Using Ontology and Evaluating the Validity of Various Semantic Similarity Measures*. 2009, Elsevier. pp. 9095-9104.

[27] Basu, T., Murthy, C.A., *Semantic Relation Between Words with the Web as Information Source*. Springer 2009, pp. 267-272.

[28] Gad, W.K., Kamel, M.S., *Enhancing Text Clustering Performance Using Semantic Similarity*. Springer 2009, P. 325.

[29] Gabrilovich, E., Markovitch, S., *Computing Semantic Relatedness using Wikipedia-Based Explicit Semantic Analysis*. 2007. pp. 6–12.

[30] WordNet, A Lexical Database for English, http://wordnet.princeton.edu, (accessed Feb 2010).

[31] Mobasher, B., *Web usage Mining and Personalization*. Practical Handbook of Internet Computing 2005.

[32] SWOOGLE, Semantic Web Search Engine, http://swoogle.umbc.edu, (accessed Feb 2010)