IJICTR International Journal of Information & Communication Technology Research

Volume 8- Number 2- Spring 2016 (51-61)

A New Hybrid Collaborative Recommender Using Semantic Web Technology and Demographic data

Faezeh Sadat Gohari
Department of Industrial Engineering, Strategic Intelligence Research Lab
K.N. Toosi University of Technology
Tehran, Iran
faezeh.gohari@gmail.com

Mohammad Jafar Tarokh
Department of Industrial Engineering, Strategic Intelligence Research Lab
K.N. Toosi University of Technology
Tehran, Iran
mjtarokh@kntu.ac.ir

Received: February 5, 2015- Accepted: January 18, 2016

Abstract— **Recommender systems are gaining a great importance with the emergence of E-commerce and business on the internet. Collaborative Filtering (CF) is one of the most promising techniques in recommender systems. It uses the known preferences of a group of users to make recommendations for other users. Regardless of its success in many application domains, CF has main limitations such as sparsity, scalability and new user/item problems. As new direction, semantic-based recommenders have emerged that deal with the semantic information of items. Such systems can improve the performance of classical CF by allowing the recommender system to make inferences based on an additional source of knowledge. Moreover, the incorporation of demographic data in recommender systems can help to improve the quality of recommendations. In this paper, we present a new hybrid CF approach that exploits Semantic Web Technology as well as demographic data to alleviate all the problems mentioned above. The experimental results on the MovieLens dataset verify the effectiveness and efficiency of our approach over other benchmarks.**

Keywords-component; recommender system, collaborative filtering, semantic Web, demographic data, e-commerce.

I. INTRODUCTION

The emergence of the Web and E-commerce has allowed companies to provide customers with more options. Therefore, businesses increase the amount of information that customers must process before they are able to select which items meet. One solution to this information overload problem is the use of recommender systems. These systems learn from customers and recommend products that satisfy their tastes and preferences. Recommender systems enhance E-commerce sales in three ways: converting browsers into buyers; improving cross-sell by suggesting additional products for customers; improving loyalty by creating a value-added relationship between the site and customers [1]. Two basic entities in all recommender systems are: the user and the item. A user provides his opinion about past items, which is usually expressed in the form of ratings. The recommender system applies a filtering algorithm on the input ratings and generates suggestions about new items for that particular user [2].

Collaborative Filtering (CF) is one of the most promising techniques in recommender systems. CF aggregates ratings of items, calculates correlations between users based on their ratings, and generates new recommendations based on inter-user comparisons. Regardless of the success of CF in many application domains, it has main limitations such as sparsity [3],

[4], scalability [5]–[7] and new user/item[8], [9] problems.

CF can take advantages of semantic reasoning to improve the recommendations' quality and cope with the above problems. Actually, the Semantic Web technologies have emerged to represent Web content in a form that is more easily machine-processable. Ontologies, as one of the key Semantic Web technologies, formally represent knowledge as a set of concepts within a domain and the relationships between them. The formal semantics underlying ontology allows the automated reasoners to infer new knowledge [10]. Combining CF with semantic information provides two primary advantages over pure CF. First, the semantic attributes for items allows the system to make inferences based on the underlying reasons for which a user may or may not be interested in a particular item. Secondly, in the case of new item or in very sparse data sets, the system can still use the semantic information to provide reasonable recommendations for users [11]. In recent years many recommender systems have appeared that use Semantic Web technologies for recommending foods [12], experts [13] , cultural heritages [14], news [15], tourism/leisure [16], [17], sound/movie/music [18]–[20], etc.

Combining CF with demographic data is another factor that can improve the quality of collaborative recommendations [2], [21]–[23]. Demographic data refers to information such as the age, the gender and the occupation of the user. Incorporating demographic data alleviates the new user problem of CF. Actually, demographic correlations help to present recommendations to new users before they have provided many ratings [21], [23].

In this paper, we present a new hybrid approach that exploits Semantic Web Technology to reduce dimensionality of the rating matrix for CF. The utilization of the reduced matrix helps to avoid the sparsity and scalability problems of CF. Also, incorporating semantic information reduces the new item problem. Moreover, we further enhance the user neighborhoods by demographic correlations which alleviate the new user problem. The experimental results on the MovieLens dataset show the effectiveness and efficiency of the proposed approach in reducing the main limitations of CF.

The rest of this paper is organized as follows. The following section provides a brief description of the related works. Section 3, describes our proposed approach and Section 4 demonstrates the experimental evaluation and results. Finally, we present our conclusions and outline future lines of research in Section 5.

II. Related Works

A. *Recommendation approaches*

Based on how the recommendations are made, recommender systems are classified into [24]: content-based recommendations (CB), collaborative recommendations and hybrid approaches.

CB stores content information about each item to be recommended and suggests items similar to the ones the user liked in the past [24]. Due to syntactic nature of this approach, it only detects similarity between items that share the same features [25]. Some limitations of this approach are new user, over-specialization and limited content analysis [24].

CF attempts to find groups of people with similar tastes to those of the user and recommend items that they have liked. This approach can be either memory-based, using the entire rating matrix to make recommendations, or model-based, using the collection of ratings to learn a model, which is then used to make rating predictions [24]. Memory-based methods usually fall into two classes: user-based (UB) and item-based (IB) approaches [26]. In UB methods [27], a subset of users is chosen based on their similarity to the active user – commonly called the neighborhood. Then, a weighted combination of neighbors' ratings is used to predict the ratings for the active user. IB methods [28] are similar to UB methods, but IB approaches try to find the similar items for each item [29]. The most extensively used similarity measures are Pearson Correlation Coefficient (PCC) [30] and vector space similarity [27]. UB requires computation that grows linearly with the number of users and items—scalability problem. In contrast, IB can quickly recommend a set of items because item-neighborhood matrix is generated offline. However, there are experiments showing that UB provides more accurate recommendations than IB [31]. Except for scalability problem, UB has another limitation, which provides much poor recommendation if users have many different interests or items have completely different content. To address this issue Li et al. [31] have explored a hybrid collaborative filtering based on item and user. This approach is able to filter dissimilar item to target item and to engender neighbor users of active user based on similar items to target item, which guarantee that target item is consist with the common interest of neighbor users.

Memory-based methods suffer from sparsity problem, which reduces accuracy of predictions. In both cases of UB and IB, only partial information from the data in the user-item matrix is employed to predict unknown ratings. Wang et al. [26] proposed the Similarity Fusion (SF) between the UB and IB methods, using also data from a new source—ratings of similar users on similar items. This model is more robust to data sparsity, because it exploits more of the data available in the user-item matrix.

In model-based algorithms, predictions can be calculated quickly once the model is generated. However, they have the overhead to build and update the model, and they cannot cover as diverse user ranges as the memory-based algorithms do. Model-based recommenders have used a variety of probabilistic models including latent class models [32], [33], regression models [34], clustering models [35], etc. Memory-based and model-based CF approaches, can be combined to leverage the advantages of each one. For example, Xue et al. [36] proposed an accurate and scalable CF using Cluster-Based Smoothing (CBS). CBS approach clusters the user data and applies intra-cluster smoothing to reduce sparsity.

Hybrid recommender systems combine two or more recommendation approaches to avoid certain

limitations of each individual approach. For example, content-boosted CF (CBCF) [37] is a hybrid system which uses a CB predictor to convert a sparse ratings matrix into a full ratings matrix; and then uses CF to provide recommendations. In our previous work [7], we proposed a hybrid collaborative filtering algorithm called CBSF (Cluster-Based Similarity Fusion), which can deal with the sparsity and scalability issues simultaneously. CBSF combines memory-based and model-based approaches. It uses SF as a memory-based algorithm and integrates it with clustering models in order to cope with the scalability problem of SF.

In recent years, some hybrid approaches have appeared that exploit the semantic [16], [18], [20], [38] or demographic [21]–[23] information associated with items and users to enhance collaborative recommendations. Sematic-based systems have proved to be successful in solving the sparsity and new item limitations of CF by allowing the recommender systems to make inferences based on an additional source of knowledge [38]. Such hybrid approaches will be detailed in the next subsection. In addition, the combination of CF with demographic data helps to alleviate the new user problem and improve the quality of recommendations. For example, Gupta and Gadge [22] combined prediction using item-based CF with prediction using demographics based user clusters in an adaptive weighted scheme. Their proposed solution is scalable while successfully addressing new user problem.

Based on this notion, this paper proposes a new hybrid approach that incorporates semantic and demographic information into the traditional CF to achieve better results in terms of efficiency and recommendation accuracy, especially when dealing with data sparsity, new user and new item problems.

B. Semantic-based recommenders

The traditional syntactic-based recommender systems miss a lot of useful knowledge during the recommendation process. Therefore, their recommendations only include items very similar to those the user already knows. Semantic-based recommender systems can overcome this problem by inferring implicit semantic relationships between items [25].

In view of the sparsity and new item problems of CF, researchers have commonly decided to opt for semantic-based recommender systems to tackle such limitations. For example, Lops et al. [39] enhanced CF through semantic user profiles which are learnt by a relevance feedback algorithm from sense-represented documents. Their approach overcomes sparsity problem of CF by computing similarity between users on the ground of their semantic-based profiles. Ceylan and Birturk [40] proposed a hybrid approach that uses semantic similarities between items to convert a sparse ratings matrix into a full ratings matrix; and then uses CF to provide recommendations. Similarly, Hu and Zhou [41] proposed an approach which uses content semantic similarities of items to enhance existing user data, and then provides personalized suggestions through CF. MC-SeCF [42] is a hybrid approach which uses the weighted harmonic mean for integrating the separate predictions from the enhanced multi criteria item-based CF and the item-based semantic filtering module. In the approach presented by Sieg et al. [43], the ontological user profiles are exploited to form semantic neighborhoods. Then, the predictions are computed as the weighted average of deviations from the neighbor's mean using the similarity between profiles as the weight. In some other approaches, a single prediction algorithm is provided by linear combination of semantic and item rating similarity [11]. Ogul and Ekmekciler [44] proposed a novel two-way CF approach based on semantically enhanced data to predict user ratings on new items from previously given ratings by other users. Lu et al. [45] presented a hybrid fuzzy semantic recommendation approach which combines item-based fuzzy semantic similarity and item-based fuzzy CF similarity techniques. Martın-Vicente et al. [46] proposed a new strategy based on semantic reasoning to prevent CF from selecting fake neighborhoods. Gohari and Tarokh [18] presented a hybrid approach that applies semantic similarity fusion as well as biclustering technique to alleviate the main limitations of CF. Al-Hassan et al. [16] proposed a hybrid semantic enhanced recommendation approach by combining a new inferential ontology-based semantic similarity measure and the standard item-based CF approach.

This paper improves the state-of-the-art by presenting a new hybrid system that fuses the semantic and demographic information of items and users within the CF framework to achieve better results in terms of efficiency and effectiveness, especially when dealing with sparsity, new user and new item problems. Our proposed system combines item-based CF and user-based CF based on the idea presented by Li et al. [31] in order to leverage the strengths of each individual approach. Item-based part of our system, which is improved by semantic information of items, refines rating matrix for user-based part of the system. Moreover, in user-based part of the system, the user neighborhoods are further enhanced by the help of demographic correlations.

III. THE PROPOSED APPROACH

In order to avoid the main limitations of CF and improve its performance, we propose a new hybrid approach which consists of two modules: (1) item-based CF using semantic similarity and (2) user-based CF using demographic data. First module combines items' semantic similarity with their rating similarity in order to filter dissimilar items to target item. Second module implements user-based CF based on the output of the previous module. Therefore, user-based CF is implemented based on items that are similar to the target item, not on all items. This leads to improvement in the quality of recommendations. This module computes similarity between two users using two sources of data: (1) the Pearson Correlation Coefficient between their vectors of ratings for similar

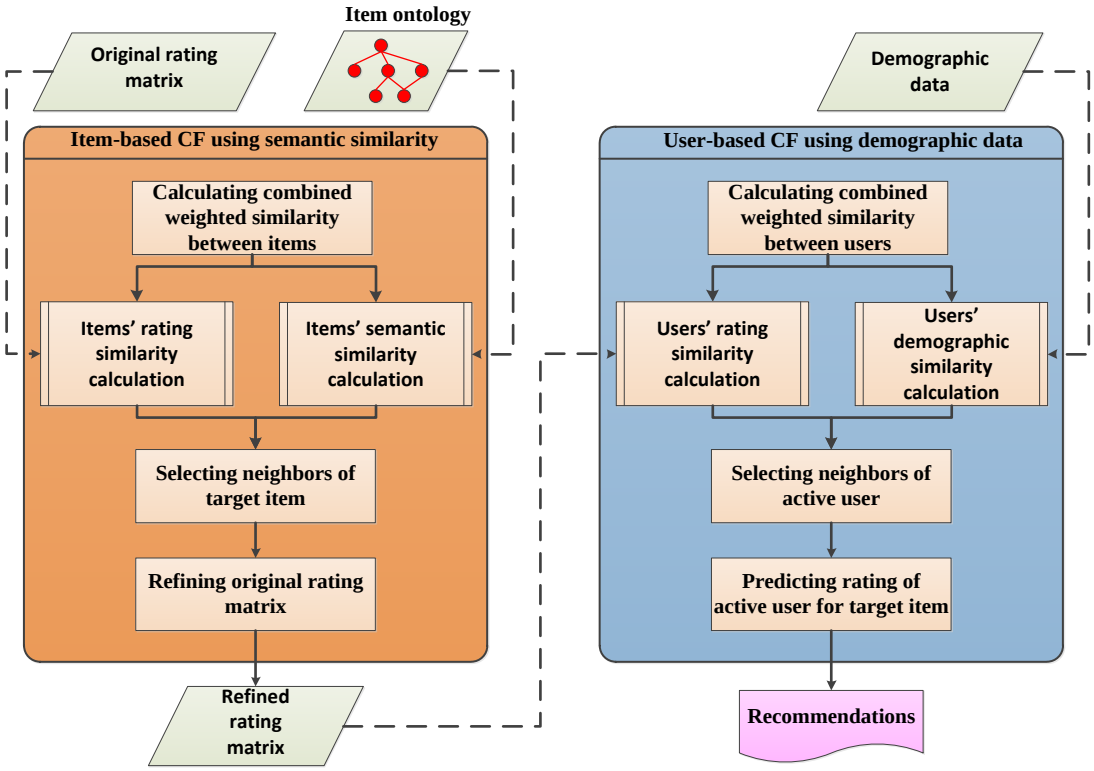

Fig. 1. The proposed approach

items to target item, and (2) demographic correlation between these users. Our proposed approach is shown in Fig. 1.

A. Item-based CF using semantic similarity

The aim of this module is to refine original rating matrix by filtering items that are dissimilar to the target item. In the first step, combined weighted similarities between target item and other items are calculated. In the next step, neighbors of the target item are selected. Finally, items that do not belong to the neighborhood of the target item are removed from the original rating matrix. This refined matrix is used as an input feature to the next module.

1) Calculating combined weighted similarity between items

To calculate the similarity between items, two types of similarity are combined: (1) semantic similarity, and (2) rating similarity.

(1) Semantic similarity: In order to integrate the semantic aspect in our recommender system, we use an item ontology and calculate similarities between ontology entities. Maedche and Zacharias [47] calculate the similarity between two ontology instances from three dimensions:

Taxonomy Similarity (TS), which computes the similarity between two instances based on their corresponding concepts' positions in concept taxonomy. TS between two concepts Ci and Cj is based on the concept match between them. Concept match is the depth of the most specific common subsumer of Ci and Cj, divided by the union of the concepts from Ci and Cj to the root.

Attribute Similarity *(AS)*, which computes the similarity between two instances *Ii* and I_j based on the similarity of their associated literals. *AS* considers the set of numeric attributes that are attributes of both *Ii* and I_j; and translates the numeric difference between their associated literals into a similarity value that is between 0 and 1.

Relation Similarity *(RS)*, which computes the similarity between *Ii* and I_j based on the similarity of the instances they have relations to. *RS* considers two type of relation: relations allowing I_i and I_j as range, and those that allow I_i and I_j as domain. The similarity of the referred instances is once again calculated using semantic similarity. So, the process of calculating similarities is recursive and a maximum recursion depth is defined to prevent infinite cycles. After reaching this maximum depth, the arithmetic mean of *TS* and *AS* is returned. In our work, we use the value 3 as the maximum depth, because the similarity remains almost constant after this value.

Semantic similarity between items (instances) *i* and *j* is calculated by the weighted arithmetic mean of *TS*, *RS* and *AS*:

$$SSim_{i,j} = \frac{W_T \times TS_{i,j} + W_A \times AS_{i,j} + W_R \times RS_{i,j}}{W_T + W_A + W_R} \quad (1)$$

where W_T , W_A and W_R are the weights of the semantic similarities. The overall similarity value between two instances is between 0 and 1, and the more similarity should result in a similarity value close to 1.

We validate our system in the movie domain and use Movie Ontology [1] which has been developed according to the OWL standard by the University of Zurich. The Movie Ontology provides a controlled vocabulary to semantically describe movie related concepts. Through this ontology it is possible to link, hierarchically and semantically, elements belonging to the movies domain. For instantiating this ontology, we gather required data from the IMDB website using a crawler. Based on the instantiated Movie Ontology and equation (1), we calculate the item-based semantic similarity values.

(2) Rating similarity: In order to compute rating similarity between two items, we use the Pearson Correlation Coefficient between their vectors of ratings. In this case, similarity between items *i* and *j* is computed as follows [48]:

$$IRSim_{i,j} = \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_i)(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U}(r_{u,i} - \bar{r}_i)^2}\sqrt{\sum_{u \in U}(r_{u,j} - \bar{r}_j)^2}} \quad (2)$$

where U is the set of all users who have rated both items *i* and *j*, $r_{u,i}$ is the rating of user *u* on item *i*, and $\bar{r}_i$ is the average rating of the *i*-th item across users.

Now, the item-based rating similarity and the item-based semantic similarity values are combined via a weighted sum to get the final weighted similarity value by:

$$IWSim_{i,j} = \alpha * SSim_{i,j} + (1-\alpha) \times IRSim_{i,j} \quad (3)$$

where $IWSim$ means the weighted similarity between items, and α is a weighted combination parameter indicating the weights of each similarity measure in the final combined measure. Selecting a proper value for α is usually highly dependent on the characteristics of the data used. When $\alpha = 1$, then only semantic similarity among items is used, and when $\alpha = 0$, only rating similarity is used. We choose the proper value for α by performing sensitivity analysis for MovieLens dataset in our experimental section.

According to the static nature and stability of item-item similarities, the expensive item-item weighted similarity matrix is created off-line.

2) Selecting neighbors of target item

In this step, the most nearest neighbors to the target are selected based on the item-item weighted similarity matrix. For this purpose, we apply best-n-neighbors [49] method and take n items with greatest similarity as the neighbors.

3) Refining original rating matrix

In this step, items that do not belong to the neighborhood of the target item are filtered. Therefore, the refined rating matrix only contains items that are similar in terms of semantic features and users' preferences. This refined matrix is the output of the first module.

B. User-based CF using demographic data

When items are quite different in terms of semantic features or users' preferences, user-based CF cannot make accurate recommendations. In fact, in such cases, neighbors of active user are selected based on their common interest for items that are not similar to the target item. Therefore, the predicted rating is based on items that are not related to the target item, so the prediction is not accurate [31]. To avoid this problem, we use the refined matrix obtained from the previous module as an input for user-based CF. So, rating similarities between the active user and other users are calculated based on similar items to the target item, not on all items. The refined rating matrix also helps to avoid the sparsity and scalability problems. The reason is that unrepresentative or insignificant items are removed and therefore the dimensionality of the original matrix is reduced directly. Moreover, in our proposed approach, the user neighborhoods are further enhanced by the help of demographic correlations. Incorporating demographic data helps to avoid new user problem. Finally, after predicting the ratings for the active user, items with the highest predicted rating are recommended.

1) Calculating combined weighted similarity between users

To calculate the similarity between users, two types of similarity are combined: (1) rating similarity, and (2) demographic similarity.

(1) Rating similarity: In order to compute rating similarity between two users, we use the Pearson Correlation Coefficient between their vectors of ratings in the refined matrix. In this case, similarity between users *a* and *u* is computed as follows [48]:

$$URSim_{a,u} = \frac{\sum_{i \in I}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I}(r_{a,i} - \bar{r}_a)^2}\sqrt{\sum_{i \in I}(r_{u,i} - \bar{r}_u)^2}} \quad (4)$$

where I is the set of items in the refined matrix rated by both users, $r_{u,i}$ is the rating given to item *i* by user *u*, and $\bar{r}_u$ is the mean rating given by user *u*.

(2) Demographic similarity: In order to calculate demographic similarity between users, we have to construct a demographic vector for each user. In our work, we use demographic data which are provided by MovieLens dataset and construct users' demographic vectors same as vectors used in [2]. In MovieLens dataset, demographic data are: age, gender, occupation and zip code. The gender can be either 'M', for male, or 'F', for female. The occupation takes a value from a list of 21 distinct possibilities. An actual sample from the demographic information about the users, which are included in the MovieLens data set, can be found

[1] http://www.movieontology.org/

in [50]. The demographic correlation between two users a and u is calculated by computing their corresponding vector similarities [50]:

$$DSim_{a,u} = \cos(d\vec{v}_a, d\vec{v}_u) = \frac{d\vec{v}_a \cdot d\vec{v}_u}{\|d\vec{v}_a\|2 \times \|d\vec{v}_u\|2} \quad (5)$$

where $d\vec{v}_a$ and $d\vec{v}_u$ are demographic vectors of user a and u, respectively.

Now, the user-based rating similarity and the user-based demographic similarity values are combined via a weighted sum to get the final weighted similarity value by:

$$UWSim_{a,u} = \beta * URSim_{a,u} + (1-\beta) \times DSim_{a,u} \quad (6)$$

where $UWSim$ means the weighted similarity between users, and β is a weighted combination parameter indicating the weights of each similarity measure in the final combined measure. If $\beta = 1$, the user-based CF similarity value is then considered as the final weighted similarity value for predictions. Whereas, if $\beta = 0$, then only the user-based demographic similarity value is used for predictions. We choose the proper value for β by performing sensitivity analysis for MovieLens dataset in our experimental section.

2) Selecting neighbors of active user

Based on the user-user weighted similarity matrix, we take n users with greatest similarity as the neighbors of the active user.

3) Predicting rating of active user for target item

Based on the results from the previous steps, the prediction value is computed as the weighted average of deviations from the neighbor's mean, as in [48]:

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in K} (r_{u,i} - \bar{r}_u) \times UWSim_{a,u}}{\sum_{u \in K} UWSim_{a,u}} \quad (7)$$

where $p_{a,i}$ is the prediction for the active user a for target item i, and K is the neighborhood or set of most similar users to the active user.

At the end of the recommendation process, the system recommends a number of unrated items that have the highest predicted rating to the active user.

IV. EXPERIMENTAL EVALUATIONS

In this section, we examine the performance of the proposed approach. The testing methodology adopted in this study is same as the one used in our previous work [7]. We use MovieLens 100K dataset which consists of 100,000 ratings, with the scale of one to five, from 943 users on 1,682 movies. This dataset has some inconsistencies (for example duplicate or unknown movies and duplicate ratings). We correct these inconsistencies and then remove users having less than 20 ratings. We partition the users into test users and train users using 10-fold cross-validation. The ratings withheld in the test set are randomly chosen based on *Given* 5, *Given* 10 and *Given* 20 experimental protocols [27].

In order to evaluate accuracy of predicted ratings, we use Mean Absolute Error (MAE) metric [27]. MAE measures the average absolute deviation between a predicted rating and the user's true rating. The Mean Absolute Error for each test user u is defined as:

$$MAE = \frac{\sum_{\forall i \in I_u} |p_{u,i} - r_{u,i}|}{|I_u|} \quad (8)$$

where I_u is the set of items rated by user u, and $p_{u,i}$ is the predicted rating for user u on item i. In our experiments, we compute the MAE on the test set for each user, and then average over the set of test users. The lower the MAE is, the more accurately the recommender system predicts user ratings.

Several parameters for our experiments are the following: weights of the semantic similarities (W_T, W_A and W_R), neighborhood size of the target item (N_i, default value 50), item-based weighted combination parameter (α, default value 0.5), neighborhood size of the active user (N_a, default value 30) and user-based weighted combination parameter (β, default value 0.5).

A. Parameters tuning

In this section we find the most appropriate values for W_T, W_A, W_R, N_i, α, N_a and β parameters, respectively.

- *Semantic similarity weights:* In this experiment, W_T, W_A and W_R parameters are set to interval [0,1] with one decimal place under the constraint of $W_T + W_A + W_R = 1$. We set N_i, α, N_a and β parameters to their default values and examine the accuracy of predictions against different values of W_T, W_A and W_R. The best result (lowest MAE) is obtained by a configuration in which the values of W_T, W_A and W_R are 0.3, 0.2 and 0.5, respectively. Therefore, in the following, these values are kept as default weights.

- *Neighborhood size of the target item (N_i):* In this step, we find the most appropriate value of N_i. For this purpose, the determined values of parameters in previous experiment are used and remaining parameters are set to their default values. We vary the neighborhood size from 10 to 200 and compute MAE. Fig. 2 shows the performance of our recommender for varying N_i. We can observe that the size of

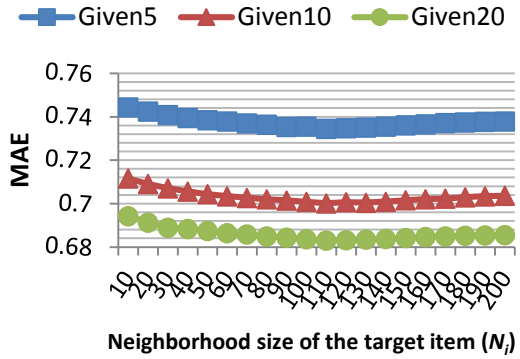

Fig. 2. MAE against different values of N_i

neighborhood affects the quality of prediction. As shown, the performance initially improves as we increase N_i from 10 to 110, after that it shows decrease in prediction quality with increased number of neighbors. The observed results imply that an appropriate neighborhood size achieves the best recommendation performance. Therefore, choosing a large number of neighbors will increase the computation complexity and reduce the quality of recommendations. According to the observed results, we select 110 as our optimal choice of N_i.

• *Item-based weighted combination parameter (α):* This parameter determines the degrees to which the semantic and rating similarities are used in the generation of neighbor items. The value of α is varied from 0 to 1. When setting α to 0, the algorithm only uses the rated information for similarity computation between items. When α is set to 1, the algorithm just use semantic information for similarity computation. For determining the most proper value for α, the best values of parameters in previous steps are used and remaining parameters are set to their default values. Fig. 3 shows the impact of α on the MAE. As shown, the optimum values of α for *Given* 5, *Given* 10 and *Given* 20 are about 0.8, 0.7 and 0.6, respectively.

When α is too large (e.g., $\alpha \geq 0.9$) which means that we rely heavily on the semantic information, the performance will decrease slightly. When α is too small (e.g., $\alpha \leq 0.3$), which means that we rely less on the semantic information, the rating data sparseness will cause the lower performance. Since for all protocols the value of α is higher than 0.5, we can conclude that the sematic information is more important than rating information for computing similarity between items. This is due to the sparsity of rating information which leads to poorer performance.

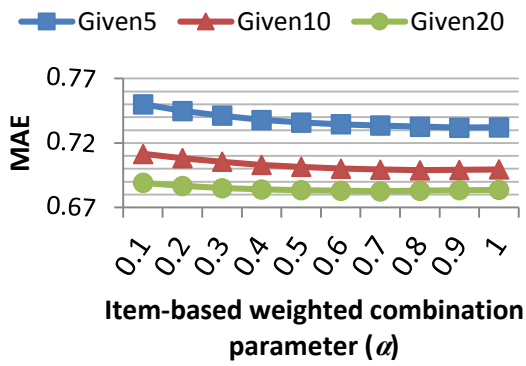

Fig. 3. MAE against different values of α

• *Neighborhood size of the active user (N_a):* Based on the best values of parameters in previous experiments, the best value of N_a is determined. We set β to its default value and examine the accuracy for varying N_a from 10 to 150 (Fig. 4). As shown, the prediction accuracy increase as the size of N_a increases. However, after a certain point, the improvement gain diminishes and the quality becomes worse. These results are coincident with those obtained from our previous work [7]. Based on the observations, the lowest MAE for *Given* 5, *Given* 10 and *Given* 20 is obtained for N_a equals 30, 40 and 70, respectively.

User-based weighted combination parameter (β): This parameter determines the degrees to which the rating and demographic similarities are used in the generation of neighbor users. The value of β is varied from 0 to 1. When setting β to 0, the algorithm only uses demographic information for similarity computation between users. When β is set to 1, the algorithm just use the rating information for similarity computation. We evaluate the impact of β on the performance of our recommender by setting the remaining parameters to their best values (Fig. 5). As shown, the optimum value of β for *Given* 20 is about 0.7 and for other protocols is about 0.5.

In the extreme case, if we employ a very large value for β, the algorithm almost forgets that demographic information exists for users and only utilizes the user-item rating matrix for similarity computation. On the other hand, a very small

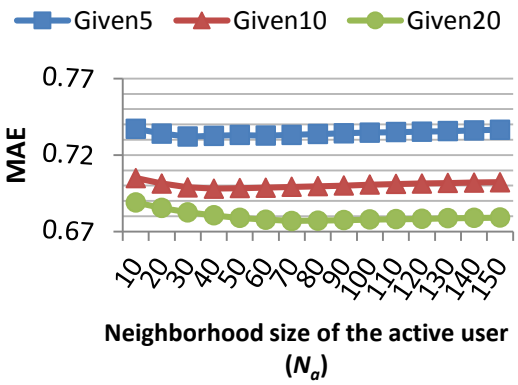

Fig. 4. MAE against different values of N_a

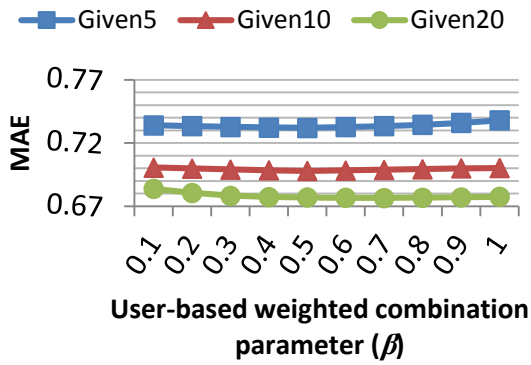

Fig. 5. MAE against different values of β

value for β, demographic information will dominate the recommendation process, leading to poorer performance.

B. Experimental results

In this section, we compare the performance of our approach against the following baseline methods: User-Based collaborative filtering using PCC (UB-PCC), Item-Based collaborative filtering using PCC (IB-PCC), Similarity Fusion (SF), Cluster-Based Smoothing (CBS), Cluster-Based Similarity Fusion (CBSF), and content-boosted CF (CBCF).

In order to evaluate the performance of above baseline methods, we tuned their related parameters to get the best results for these methods.

Initially, we measure MAE for all the examined algorithms and compare the overall performance of our approach with other methods. Then, we compare the performance of all the examined algorithms in dealing with the sparsity, scalability and new item/user problems.

1) Overall performance

In this section, we evaluate the overall performance of our approach and the benchmark algorithms in terms of MAE. For each method, parameters are set to the best values. The results are summarized in Table 1.

Table. 1. Comparison between different algorithms in terms of MAE

	MAE		
Algorithms	**Given5**	**Given10**	**Given20**
Our approach	**0.732**	**0.698**	**0.676**
CBCF	0.766	0.735	0.718
CBSF	0.784	0.741	0.734
SF	0.774	0.739	0.730
CBS	0.800	0.785	0.751
UB-PCC	0.830	0.806	0.794
IB-PCC	0.852	0.820	0.801

Clearly, our approach outperforms other methods in all configurations. This is due to: (1) the utilization of the semantic-based pre-filtering method as a dimensionality reduction technique for user-based CF, and (2) the utilization of the demographic correlations for enhancing the user neighborhoods.

2) Impact on the sparsity problem

We compare the performance of our approach against the benchmark algorithms on different sparsity levels. In each level, 10000 ratings are reduced from train set. For each method, parameters are set to the best values and Given 5 protocol is used. The results against the different sparsity levels are presented in Fig. 6. As expected, with increasing the sparsity level, the performance downgrades for all methods. This is due to reduction in the train set size. Fig. 6 shows that our approach outperforms the other methods. The reason is that the first module in our approach, removes unrepresentative or insignificant items to reduce the dimensionality of the original user-item matrix directly. User-based CF is implemented on the reduced matrix and therefore the sparsity problem is alleviated.

3) Impact on the scalability problem

For comparing the scalability of different methods, the run time of their online parts are measured. In our proposed approach, the online part is the second module. We measure the average time (ms) that takes to provide recommendations to a test user (runtime per user). For each method, parameters are set to the best values and Given 20 protocol is used. The results against the different size of the train set are presented in Fig. 7. The training sets are created using *k*-fold cross-validation (k = 2, 3, 4, 5, 10). As expected, with increasing the size of the train set, the runtime increases for all methods. As shown, the runtime of UB-PCC, CBCF, and SF grows linearly with the size of the train set. This is due to the online computation of similarity between users in these approaches. In contrast, the runtime of IB-PCC is stable because it creates the expensive similar-items table offline. Our approach is stable but it needs a little more time than IB-PCC. The reason is that our approach contains an online module (i.e., user-based CF using demographic data) which implements user-based filtering based on

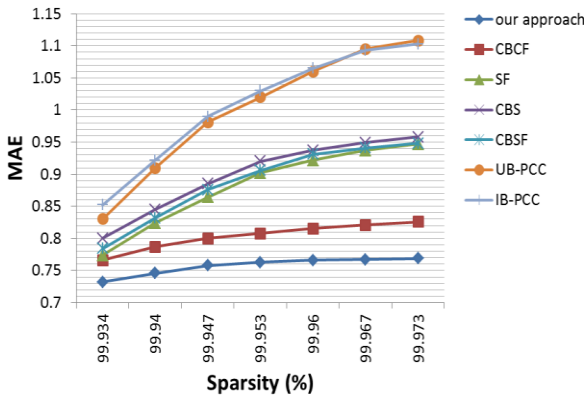

Fig. 6. Impact on the sparsity problem

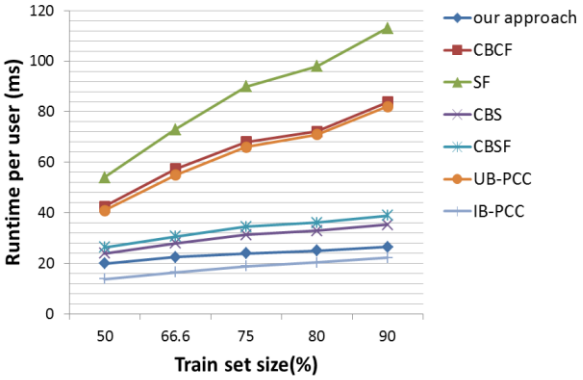

Fig. 7. Impact on the scalability problem

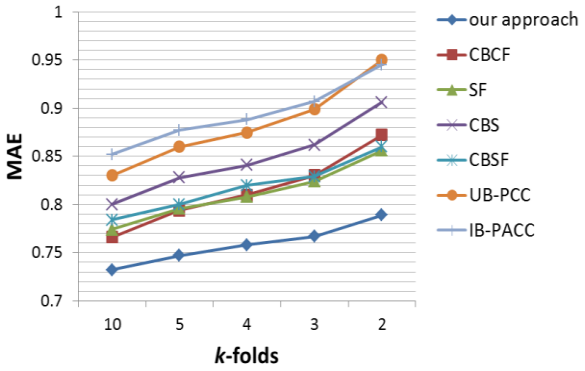

Fig. 9. Impact on the new user problem

the results of the item-based filtering. As mentioned before, in the proposed approach, the online module computes similarities between users based on a refined rating matrix. By removing the unrepresentative or insignificant items, the dimensionality of the original matrix is reduced, resulting in better scalability. CBS and CBSF algorithms are almost stable because clusters are also created offline. However, our approach needs a little less time because the dimensionality of the rating matrix in our approach is lower than CBS or CBSF.

4) Impact on the new item problem

For testing the new item problem, we split items into test and train items using *k*-fold cross-validation. The items that have less than 5 ratings are considered as new items (test items). So, for each test item, we use *Given* 5 and randomly select 5 ratings as the observed ratings in the train set. Then, the accuracy of predictions is measured for each method by setting their related parameters to the best values. Fig. 8 illustrates MAE against different *k*-fold validation. With decreasing *k*, the number of new items increases and thus the MAE increases for all methods. As shown, our approach has the highest performance under the all *k*-fold validation. Actually, in the case of new items, our system can still use the semantic information to provide reasonable recommendations for users. Therefore, it can be concluded that this approach is a significant improvement on alleviating the new item problem in comparison to the benchmark algorithms.

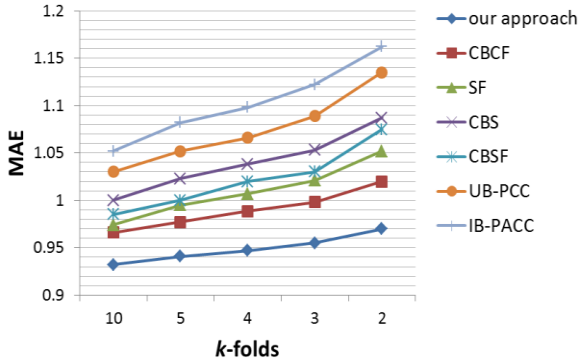

Fig. 8. Impact on the new item problem

5) Impact on the new user problem

Testing the new user problem is similar to new item problem. Here, users are split using *k*-fold cross-validation and *Given 5* is used for each test user. Then, the MAE is measured for each method. Fig. 9 illustrates MAE against different *k*-fold validation. With decreasing *k*, the number of new users increases and thus MAE increases for all methods. The results show the proposed approach outperforms other counterparts for cold start new users in terms of the prediction accuracy. Actually, traditional CF algorithms cannot produce reliable recommendations for new users who have not yet provided sufficient information about their preferences. In such cases, the utilization of demographic data instead of rating history helps to tackle the new user problem in some extent. Thus, compared to other approaches, the proposed approach can alleviate the new user problem by applying demographic information during the similarity calculation process in the second module.

V. Conclusions and Future Works

In this paper, we proposed a new approach which exploits Semantic Web Technology as well as demographic data for improving CF recommender in E-commerce. Our approach consists of two modules: First module identifies similar items to the target item in terms of semantic features and users' preferences. Then, the original rating matrix is refined by removing items that do not belong to the neighborhood of the target item. Second module implements user-based CF using the refined matrix from the previous module. Moreover, in the second module, the user neighborhoods are further enhanced by demographic correlations. The experimental results verify the effectiveness and efficiency of the proposed approach in dealing with the main limitations of CF. In future work, we would like to apply our proposed approach to the real applications to test its performance. Furthermore, we would like to use word senses similarity in order to capture the semantic similarity more precisely.

REFERENCES

[1] J. B. Schafer, J. Konstan, and J. Riedl, "Recommender systems in e-commerce," in *Proceedings of the 1st ACM conference on Electronic commerce*, 1999, pp. 158–166.

[2] M. G. Vozalis and K. G. Margaritis, "Using SVD and demographic data for the enhancement of generalized Collaborative Filtering," *Inf. Sci.*, vol. 177, no. 15, pp. 3017–3037, Aug. 2007.

[3] A. Kumar and A. Sharma, "Alleviating sparsity and scalability issues in collaborative filtering based recommender systems," in *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA)*, 2013, pp. 103–112.

[4] L. Sun, G. Hao, J. Li, and J. Lv, "Cluster-based smoothing and linear-function fusion for collaborative filtering," in *Foundations of Intelligent Systems*, Springer, 2014, pp. 681–692.

[5] S. Bakshi, A. K. Jagadev, S. Dehuri, and G.-N. Wang, "Enhancing scalability and accuracy of recommendation systems using unsupervised learning and particle swarm optimization," *Appl. Soft Comput.*, vol. 15, pp. 21–29, 2014.

[6] N. Koenigstein and Y. Koren, "Towards scalable and accurate item-oriented recommendations," in *Proceedings of the 7th ACM conference on Recommender systems*, 2013, pp. 419–422.

[7] F. S. GOHARI and M. J. TAROKH, "A CLUSTER-BASED SIMILARITY FUSION APPROACH FOR SCALING-UP COLLABORATIVE FILTERING RECOMMENDER SYSTEM," 2014.

[8] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades, "Facing the cold start problem in recommender systems," *Expert Syst. Appl.*, vol. 41, no. 4, Part 2, pp. 2065–2073, Mar. 2014.

[9] H. Liu, Z. Hu, A. Mian, H. Tian, and X. Zhu, "A new user similarity model to improve the accuracy of collaborative filtering," *Knowl.-Based Syst.*, vol. 56, pp. 156–166, Jan. 2014.

[10] W. Carrer-Neto, M. L. Hernández-Alcaraz, R. Valencia-García, and F. García-Sánchez, "Social knowledge-based recommender system. Application to the movies domain," *Expert Syst. Appl.*, vol. 39, no. 12, pp. 10990–11000, Sep. 2012.

[11] B. Mobasher, X. Jin, and Y. Zhou, "Semantically enhanced collaborative filtering on the web," in *Web Mining: From Web to Semantic Web*, Springer, 2004, pp. 57–76.

[12] M. A. El-Dosuky, M. Z. Rashad, T. T. Hamza, and A. H. El-Bassiouny, "Food Recommendation using Ontology and Heuristics," *ArXiv Prepr. ArXiv13121448*, 2013.

[13] E. Davoodi, K. Kianmehr, and M. Afsharchi, "A semantic social network-based expert recommender system," *Appl. Intell.*, pp. 1–13, 2013.

[14] T. Ruotsalo, K. Haav, A. Stoyanov, S. Roche, E. Fani, R. Deliai, E. Mäkelä, T. Kauppinen, and E. Hyvönen, "SMARTMUSEUM: A mobile recommender system for the Web of Data," *Web Semant. Sci. Serv. Agents World Wide Web*, 2013.

[15] M. Capelle, F. Hogenboom, A. Hogenboom, and F. Frasincar, "Semantic News Recommendation Using Wordnet and Bing Similarities," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, New York, NY, USA, 2013, pp. 296–302.

[16] M. Al-Hassan, H. Lu, and J. Lu, "A semantic enhanced hybrid recommendation approach: A case study of e-Government tourism service recommendation system," *Decis. Support Syst.*, vol. 72, pp. 97–109, Apr. 2015.

[17] T. Ruotsalo, K. Haav, A. Stoyanov, S. Roche, E. Fani, R. Deliai, E. Mäkelä, T. Kauppinen, and E. Hyvönen, "SMARTMUSEUM: A mobile recommender system for the Web of Data," *Web Semant. Sci. Serv. Agents World Wide Web*, vol. 20, pp. 50–67, May 2013.

[18] F. S. Gohari and M. J. Tarokh, "New Recommender Framework: Combining Semantic Similarity Fusion and Bicluster Collaborative Filtering," *Comput. Intell.*, 2015.

[19] D. Bogdanov, M. Haro, F. Fuhrmann, A. Xambó, E. Gómez, and P. Herrera, "Semantic audio content-based music recommendation and visualization based on user preference examples," *Inf. Process. Manag.*, vol. 49, no. 1, pp. 13–33, Jan. 2013.

[20] V. C. Ostuni, T. Di Noia, E. Di Sciascio, S. Oramas, and X. Serra, "A Semantic Hybrid Approach for Sound Recommendation," in *Proceedings of the 24th International Conference on World Wide Web Companion*, Republic and Canton of Geneva, Switzerland, 2015, pp. 85–86.

[21] B. Yapriady and A. L. Uitdenbogerd, "Combining demographic data with collaborative filtering for automatic music recommendation," in *Knowledge-Based Intelligent Information and Engineering Systems*, 2005, pp. 201–207.

[22] J. Gupta and J. Gadge, "Performance analysis of recommendation system based on collaborative filtering and demographics," in *Communication, Information & Computing Technology (ICCICT), 2015 International Conference on*, 2015, pp. 1–6.

[23] T. Chen and L. He, "Collaborative filtering based on demographic attribute vector," in *Future Computer and Communication, 2009. FCC'09. International Conference on*, 2009, pp. 225–229.

[24] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *Knowl. Data Eng. IEEE Trans. On*, vol. 17, no. 6, pp. 734–749, 2005.

[25] Y. Blanco-Fernández, J. J. Pazos-Arias, A. Gil-Solla, M. Ramos-Cabrer, M. López-Nores, J. García-Duque, A. Fernández-Vilas, R. P. Díaz-Redondo, and J. Bermejo-Muñoz, "A flexible semantic inference methodology to reason about user preferences in knowledge-based recommender systems," *Knowl.-Based Syst.*, vol. 21, no. 4, pp. 305–320, 2008.

[26] J. Wang, A. P. De Vries, and M. J. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006, pp. 501–508.

[27] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 1998, pp. 43–52.

[28] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Trans. Inf. Syst. TOIS*, vol. 22, no. 1, pp. 143–177, 2004.

[29] H. Ma, I. King, and M. R. Lyu, "Effective missing data prediction for collaborative filtering," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 39–46.

[30] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994, pp. 175–186.

[31] Y. Li, L. Lu, and L. Xuefeng, "A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce," *Expert Syst. Appl.*, vol. 28, no. 1, pp. 67–77, 2005.

[32] H. Langseth and T. D. Nielsen, "Scalable learning of probabilistic latent models for collaborative filtering," *Decis. Support Syst.*, vol. 74, pp. 1–11, 2015.

[33] T. Zhao, J. McAuley, and I. King, "Improving Latent Factor Models via Personalized Feature Projection for One Class Recommendation," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015, pp. 821–830.

[34] M. Nilashi, D. Jannach, O. bin Ibrahim, and N. Ithnin, "Clustering-and regression-based multi-criteria collaborative filtering with incremental updates," *Inf. Sci.*, vol. 293, pp. 235–250, 2015.

[35] X. Wang, D. He, D. Chen, and J. Xu, "Clustering-Based Collaborative Filtering for Link Prediction," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[36] G.-R. Xue, C. Lin, Q. Yang, W. Xi, H.-J. Zeng, Y. Yu, and Z. Chen, "Scalable collaborative filtering using cluster-based smoothing," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 114–121.

[37] P. Melville, R. J. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations," in *AAAI/IAAI*, 2002, pp. 187–192.

[38] Q. Shambour and J. Lu, "A trust-semantic fusion-based recommendation approach for e-business applications," *Decis. Support Syst.*, vol. 54, no. 1, pp. 768–780, 2012.

[39] P. Lops, M. Degemmis, and G. Semeraro, "Improving social filtering techniques through wordnet-based user profiles," in *User Modeling 2007*, Springer, 2007, pp. 268–277.

[40] U. Ceylan and A. Birturk, "Combining feature weighting and semantic similarity measure for a hybrid movie recommender system," in *The 5th SNA-KDD Workshop'11*, 2011.

[41] B. Hu and Y. Zhou, "Content semantic similarity boosted collaborative filtering," in *Computational Intelligence and Security, 2008. CIS'08. International Conference on*, 2008, vol. 2, pp. 7–11.

[42] Q. Shambour and J. Lu, "A hybrid multi-criteria semantic-enhanced collaborative filtering approach for personalized recommendations," in *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, 2011, pp. 71–78.

[43] A. Sieg, B. Mobasher, and R. Burke, "Improving the effectiveness of collaborative recommendation with ontology-based user profiles," in *proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, 2010, pp. 39–46.

[44] H. Ogul and E. Ekmekciler, "Two-way collaborative filtering on semantically enhanced movie ratings," in *Proceedings of the ITI 2012 34th International Conference on Information Technology Interfaces (ITI)*, 2012, pp. 361–366.

[45] J. Lu, Q. Shambour, Y. Xu, Q. Lin, and G. Zhang, "A WEB-BASED PERSONALIZED BUSINESS PARTNER RECOMMENDATION SYSTEM USING FUZZY SEMANTIC TECHNIQUES," *Comput. Intell.*, vol. 29, no. 1, pp. 37–69, 2013.

[46] M. I. Martín-Vicente, A. Gil-Solla, M. Ramos-Cabrer, J. J. Pazos-Arias, Y. Blanco-Fernández, and M. López-Nores, "A semantic approach to improve neighborhood formation in collaborative recommender systems," *Expert Syst. Appl.*, vol. 41, no. 17, pp. 7776–7788, 2014.

[47] A. Maedche and V. Zacharias, "Clustering ontology-based metadata in the semantic web," in *Principles of Data Mining and Knowledge Discovery*, Springer, 2002, pp. 348–360.

[48] P. Melville and V. Sindhwani, "Recommender systems," in *Encyclopedia of machine learning*, Springer, 2010, pp. 829–838.

[49] J. Herlocker, J. A. Konstan, and J. Riedl, "An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms," *Inf. Retr.*, vol. 5, no. 4, pp. 287–310, 2002.

[50] M. Vozalis and K. G. Margaritis, "On the enhancement of collaborative filtering by demographic data," *Web Intell. Agent Syst.*, vol. 4, no. 2, pp. 117–138, 2006.

Faezeh S. Gohari received her B.Sc. degree in Computer Engineering from Elm & Farhang University, Tehran, Iran. In late 2013, she received her M.Sc. degree in Electronic Commerce from K.N. Toosi University of Technology, Tehran, Iran. Her research interests include recommender systems, semantic web, soft computing and data mining.

Mohammad J. Tarokh associate professor in the department of Industrial Engineering at K.N. Toosi University of Technology, Tehran, Iran. He received his B.Sc. degree from Sharif University of Technology in Tehran, M.Sc. degree from University of Dundee in UK and Ph.D. from University of Bradford, UK. Recently, he has established the Strategic Intelligence Research Laboratory at K.N. Toosi University of Technology, Tehran, Iran. His main research interests are in knowledge management, business intelligence, customer relationship management and supply chain management.