Recommender System Based on Semantic Similarity

Karamollah Bagheri Fard[1], Mehrbakhsh Nilashi[2], Mohsen Rahmani[3], Othman Ibrahim[4]
[1]Dept. of Computer Engineering Islamic Azad University, Yasooj branch, Yasooj, Iran
[2,4]Faculty of Computing, Universiti Teknologi Malaysia, Skudai, Johor, Malaysia
[1,3]Dept. of Computer Engineering, Faculty of Engineering, Arak University, Arak, Iran

Article Info

ABSTRACT

In electronic commerce, in order to help users to find their favourite products, we essentially need a system to classify the products based on the user's interests and needs to recommend them to the users. For the same reason the recommendation systems are designed to help finding information in large websites. They are basically developed to offer products to the customers in an automated fashion to help them to do conveniently their shopping. The developing of such systems is important since there are often a large number of factors involved in purchasing a product that would make it difficult for the customer to make the best decision. Finding relationship among users and relationships among products are important issue in these systems. One of relations is similarity. Measure similarity among users and products is used in the pure methods for calculating similarity degree. In this paper, semantic similarity is used to find a set of k nearest neighbours to the target user, or target item. Thus, because of incorporating semantic similarity in the proposed recommendation system, from the experimental results, the high accuracy was obtained on private building company dataset in comparison with state-of-the-art recommender systems.

Corresponding Author:

Karamollah Bagheri Fard,
Departement of Computer Engineering
Islamic Azad University, Yasooj branch,
Yasooj, Iran
Bfkaramollah2@live.utm.my

1. INTRODUCTION

The recommendation systems have been basically created to recommend products to customers and help them to purchase, because it is unlikely to make an optimal decision in buying [1]. The recommendation systems already presented have lots of problems and this has made the large websites to have difficulty in recommending products to the users. In the past two decades, we have witnessed a significant increase in the number of e-commerce sites that can guide users in the decision making process. In addition to benefiting users, e-commerce sites benefit companies as well, by giving them access to information about user interests and choices, and ultimately increasing their sales and profits. Given the large number of products/items available online, the big challenge that these e-commerce sites face today is how to effectively identify items that users might be interested in purchasing and to recommend such items to users. Recommender systems can help here. The history of recommender systems dates back to the year 1979 with relation to cognitive science [2]. Recommender systems gained prominence among other application areas such as approximation theory [3], information retrieval [4], forecasting theories [5], management science [6] and consumer choice modelling in marketing [7]. In the mid-1990s, recommender systems became active in the research domain when the focus was shifted to recommendation problems by researchers that explicitly rely on user rating structure and also emerged as an independent research area [8-10]. RS's make use of previous user likes and dislikes and statistical methods to extract patterns about users and items. These patterns can be then

employed to suggest items of interest to users. Given the advantages that recommender systems offer, they have become an integral part of many business models and are being used very extensively in many e-commerce websites such as Amazon.com, eBay, Reel.com, etc.

In this paper, semantic similarity is used to find a set of k nearest neighbours to the target user, or target item. The objective of this paper is to incorporate semantic similarity in the developed recommendation system, evaluate its accuracy using the private building company dataset andcompare with state-of-the-art recommender systems.

2.   RECOMMENDATION TECHNIQUES

Recommendation methods have a variety of possible categories [11, 12]. For arranging a first review of the different kinds of RSs, we want to quotation a taxonomy offered by [13] that has become a traditional way of identifying between recommender techniques and mentioning them. Burke [13] differentiates between 6 different classes of recommendation approaches that 3 main of them are explained as follows:

2.1. Content-Based Filtering (CBF)

The content based approach provides recommendations which are based on information on the content of items rather than on other user's opinions. It uses a machine learning algorithm to induce the profile of the user preferences from examples based on a feature description of the content. The content of an item can be structured or unstructured. If we consider the content of a movie as director, writer, cast etc., then each of these attribute can be considered as a feature. But in the case of unstructured items such as text data, deciding on the feature set is more difficult. Content-based recommenders treat suggestions as a user-specific category problem and learn a classifier for the customer's preferences depending on product traits.

According to Ziegler [14], techniques applying a content-based recommendation strategy evaluate a set of documents and/or details of products previously ranked by a user, and develop a model or user profile of user passions depending on the features of the things rated by that user. Content-based RS's can be used in a variety of domains ranges i.e., recommending web pages, news articles, jobs , television programs, and products for sale.

2.2. Collaborative Filtering

Based on the genuine and ordinary of this strategy [15] the items that other users with similar tastes liked in the past are recommended to the target user. The likeness in taste of two customers is computed with regards to the likeness in the rating history of the users.

All collaborative filtering methods share a capability to utilize the past ratings of users in order to predict or recommend new content that an individual user will like [16]. The actual assumption is highly based in the idea of likeness between users or between products, with the similarity being expressed as a function of agreement between past ratings or preferences. Two basic variants of collaborative filtering approach can be classified as user-based and item-based.

2.3. Hybrid Recommender Systems

Hybrid RS's can be obtained from a combination of mentioned techniques by blending two or more techniques that tries to fix disadvantages of them. A hybrid approaches more have been used by combing collaborative and content-based methods, which tries to eliminate shortcomings of both [13, 17, 18]. Moreover, a combination for developing hybrid recommender system is depending on the domain and data characteristics. Seven categories of hybrid recommendation systems, weighted, switching, mixed, feature combination, feature augmentation, cascade, and meta-level have been introduced by [19].

3.   SIMILARITY METRICS

One crucial step in the collaborative filtering algorithm is to calculate the similarity between items and users and finally to choose a group of nearest neighbours as recommendation partners for an active user. After establishing a set of profiles by the recommender system, it is possible to reason about the similarities between users or items, and finally chooses a group of nearest neighbours as recommendation partners for an active user. Because of importance of similarity matrices, some of the popular similarity metrics that used in collaborative filtering will be examined in detail.

3.1. Cosine Similarity

Usually cosine similarity metric is used for estimate the similarity between two instance a and b in information retrieval that the objects are in the shape of vector x_a and vector x_b[20, 21] and calculating the Cosine Vector (CV) (or Vector Space) similarity between these vectors indicate the distance of them to each other [22, 23]:

$$\cos(X_a, X_b) = \frac{X_a . X_b}{\| X_a \|^2 * \| X_b \|^2} \tag{1}$$

In the context of item recommendation, for computing user similarities, this measure can be employed in which a user u indicates vector $x_u \in R|I|$ where $x_{ui} = r_{ui}$ if user u has rated item i and for unrated item considers 0. The similarity between two users u and v would then be calculated as:

$$CV(u,v) = \cos(X_a, X_b) = \frac{\sum_{i \in I_{uv}} r_{ui} r_{vi}}{\sqrt{\sum_{i \in I_u} r_{ui}^2} \sqrt{\sum_{j \in I_v} r_{vi}^2}} \tag{2}$$

Where r_{uv} once more indicates the items rated by both u and v. A shortcoming of this measure is that it does not examine the differences in the mean and variance of the ratings made by users u and v.

Cosine similarity is calculated on a scale between -1 and +1, where -1 implies the objects are completely dissimilar, +1 implies they are completely similar and 0 implies that the objects do not have any relationship to each other. In prior researches, vector similarity has been proven to work well in information retrieval [4] but it has not been found to carry out as well as Pearson's for user-based CF [24].

3.2. Pearson Correlation

Pearson Correlation (PC) is a well-known metric that compares ratings where the effects of mean and variance have been eliminated is the Pearson Correlation (PC) similarity [25, 26]:

$$PC(u,v) = \frac{\sum_{i \in I_{uv}} (r_{u,i} - \overline{r}_u)(r_{vi} - \overline{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{u,i} - \overline{\overline{r}_u})^2 \sum_{i \in I_{uv}} (r_{vi} - \overline{r}_v)^2}} \tag{3}$$

Also, for acquiring the similarity between two items i and j the ratings given by users that have rated both of these items is compared:

$$PC(i,j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \overline{r}_i)(r_{uj} - \overline{r}_j)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \overline{r}_i)^2 \sum_{u \in U_{ij}} (r_{uj} - \overline{r}_j)^2}} \tag{4}$$

3.3. Spearman's Correlation Coefficient

Spearman's correlation coefficient is a rank coefficient that independent of the actual item rating values, estimates the difference in the ranking of the items in the profiles [27]. First user's list of ratings is turned into a list of ranks, where the user's highest rating takes the rank of 1, and tied ratings take the average of the ranks for their spot [28, 29]. Herlocker [29] showed that Spearman's performs similarly to Pearson's for user-based CF.

$$SRC(i,j) = \frac{\sum_{i \in I} (r_{a,i} - \overline{r}_a)(r_{b,i} - \overline{r}_b)}{\sqrt{\sum_{i \in I} (r_{a,i} - \overline{r}_a)^2 \sum_{i \in I} (r_{b,i} - r_b)^2}} \tag{5}$$

The Spearman Correlation Coefficient for user-user similarity between two users a and b have been represented in Equation 5. It is declared regarding the set of all co-rated items (I) that $r_{a,i}$ and $r_{b,i}$ indicate rank each user gave to each item i and $\overline{r}_a$ and $\overline{r}_b$ finally indicate each user's average rank. Once again, the

correlation is measured on a scale between -1 to +1 where , -1 implies the objects are completely dissimilar, +1 implies they are completely similar and 0 implies that the objects do not have any relationship to each other.

4.   SEMANTIC SIMILARITY

There are three types of semantic similarity measures used in calculating the similarities between items serving as ontology-based metadata instances that are defined as three types of Taxonomy Similarity (TS), Attribute Similarity (AS) and Relation Similarity (RS).For each pair of item, the above semantic similarity measures are used by obtaining the weighted values of these measures [30]. The semantic similarity between instance I_i and I_j is denoted by SS (I_i,I_j) and TS, RS, and AS is calculated for weighted arithmetic mean.

$$SS\left(I_1, I_2\right) = \frac{a \times TS\left(I_1, I_2\right) + b \times RS\left(I_1, I_2\right) + c \times AS\left(I_1, I_2\right)}{a + b + c} \tag{6}$$

4.1. Taxonomy Similarity

Taxonomy Similarity (TS) between two instances is determined according to their corresponding concepts' places in concept hierarchy (H^c) that specified in ontology model [31]. Mainly, in TS the closer concepts in taxonomy indicates the strong similarity between them. After computing similarities between concepts in ontology, it is possible to calculate similarity between two instances by considering the similarities between relative concepts of these instances. To do taxonomy similarity calculation between two concepts, 4 different measures $TSC_{Wu\&Palmer}$, TSC_{CM}, TSC_{Lin} and $TSC_{Mcleancan}$ be used.

According to Maedche and Zacharias [32] TSC_{CM} or taxonomy similarity between concepts using concept match is used to calculate TSC. In ontology, it is defined based on distance between two concepts. Concept Match (CM) between two concepts uses TSCCM and is determined as:

$$CM\left(C_i, C_j\right) = \frac{\left|UC\left(C_i, H^c\right) \cap UC\left(C_j, H^c\right)\right|}{\left|UC\left(C_i, H^c\right) \cup UC\left(C_j, H^c\right)\right|} \tag{7}$$

where UC (Upwards cotopy) is determined as :

$$UC\left(C_i, H^c\right) = \left\{C_j \in \tilde{C} \,\middle|\, H^c\left(C_i, C_j\right) \vee C_i = C_j\right\} \tag{8}$$

A set of concepts that make a path from a given concept given concept to the root of a given concept hierarchy is determined by UC. Subsequently, TSC_{CM} can be defined as follow:

$$TSC_{CM}\left(C_i, C_j\right) = \begin{cases} 1, & \text{if } C_i = C_j \\ \dfrac{CM\left(C_i, C_j\right)}{2}, & \text{otherwise} \end{cases} \tag{9}$$

$TSC_{Wu\&Palmer}$is as second measure that was proposed by Wu and Palmer [33] .Wu and Palmer's measure that is used for similarity between concepts is defined as following:

$$TSC_{Wu\&palmer}\left(C_i, C_j\right) = \begin{cases} 1, & \text{if } C_i = C_j \\ \dfrac{2 \cdot N_3}{N_1 + N_2 + 2 \cdot N_3}, & \text{otherwise} \end{cases} \tag{10}$$

The number *subConceptOf* is defined by N1 and N2 that make link from C_i and C_j to their most particular concept C_k that subsumes both of them. Also, N_3 stands to the number of *subConceptOf* links from C_k to the root of the ontology (root concept). Compared to TSC_{CM}, $TSC_{Wu\&Palmer}$is also based on the distance between concepts in ontology. Lin's taxonomy similarity presented by [34] is chosen as the third measure for computing TSC. Lin's taxonomy similarity is an information theoretic approach based on probabilistic model. In the following, the taxonomy similarity between concepts by Lin's taxonomy similarity (TSC_{Lin}) is presented as :

$$TSC_{Lin}(C_i, C_j) = \begin{cases} 1, & \text{if } C_i = C_j \\ \dfrac{2 \cdot \log P_r(C_k)}{\log P_r(C_i) + \log P_r(C_j)}, & \text{otherwise} \end{cases} \tag{11}$$

$Pr(C_n)$ stands to the probability which a randomly chosen instance belongs to concept C_n, and incorporating C_i and C_j is C_k representing the most specific concept.

The Movie concept and Feature concept are the two concepts utilized in this study, and the values of their instances have no effect on each other's probabilities. As an example, only the Movie instances are considered when the probability of a concept belongs to Movie concept. $Pr(C_n)$ is therefore represent the following.

$$Pr(C_n) = \begin{cases} \dfrac{|ISET(C_n)|}{|ISET(Movie)|}, & \text{if } Movie \in UC(C_n, H^c) \\[2ex] \dfrac{|ISET(C_n)|}{|ISET(Feature)|}, & \text{if } Feature \in UC(C_n, H^c) \end{cases} \tag{12}$$

A set of instances is determined by $ISET(C_n)$ which are instances of the concepts that are linked to the C_n concept by *subConceptOf* links. $ISET(C_n)$ can be defined as following:

$$ISET(C) = \{I \in \tilde{I} \mid C \in UC(CSET(I), H^C)\}$$
$$CSET(I) = \{C \in \tilde{C} \mid C(I) \tag{13}$$

$CSET(I)$ indicates the set of concepts that instance I is linked by *instanceOf* links. The other measure by [35] varied strategies of similarity calculation are analysed and similarity measure defined in the following equation which is called taxonomy similarity between concepts using Mclean's taxonomy similarity (TSC_{Mclean}), gives the best performance.

$$TSC_{Mclean}(C_i, C_j) = \begin{cases} 1, & \text{if } c_i = c_j \\ e^{-\alpha l} \cdot \dfrac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}}, & \text{otherwise} \end{cases} \tag{14}$$

The work carried out in [35] reveals that Mclean's taxonomy similarity measurement produced the best performance with optimal values of parameters α and β having 0.2 and 0.6 respectively, when evaluation was done on separate similarity calculation strategies. l and h are the shortest path length between C_i and C_j, and the most specific concept in ontology respectively. As stated above, TSC_{CM}, $TSC_{Wu\&Palmer}$, and TSC_{Mclean} are based on distance between concepts while TSC_{Lin} on information theoretic approach.

$$TS(I_i, I_j) = \begin{cases} 1, & \text{if } I_i = I_j \\ SSIM(CSET(I_i), CSET(I_j)), & \text{otherwise} \end{cases} \tag{15}$$

In the Equation 15 the CSET was determined. SSIM (S1, S2) indicates the similarity between two sets S1 and S2. Similarity between two sets can be calculated applying the similarities between their elements, in this case TSC of concepts, and a method that identifies a way of employing these similarities.

4.2. Relation Similarity

Relation similarity (RS) is another similarity measure that uses ontology-based metadata [36]. In ontology-based metadata, RS between two instances is based on their relations to other instances. Assume that Director Z is as a director of Movie α and Movie β and Director Y is as a director of Movie ρ. That is clear that the RS between Movie α and Movie β is higher than the RS between Movie α and Movie ρ. It is because of belonging same director for Movie α and Movie β. For RS measure, the modified version of Maedche and Zacharias's RS measure from the [37] is used. RS between instances I_i and I_j can be computed as follows:

$$RS(I_i, I_j) = \begin{cases} 1, & \text{if } I_i = I_j \\ \dfrac{\sum\limits_{\rho \in P_{Co-I}} OR(I_i, I_j, \rho, IN) + \sum\limits_{\rho \in P_{Co-O}} OR(I_i, I_j, \rho, OUT)}{|P_{CO-I}| + |P_{CO-O}|}, & \text{otherwise} \end{cases} \tag{16}$$

P_{co-I} and P_{co-O} stands are for incoming relations and outgoing relations respectively. The former is the set of relations allowing $UC(C\ (I_i), H^c)$ and $UC(C(I_j), H^c)$ as ranges while the latter is the set of relations granting $UC(C(I_i), H^c)$ and $UC(C(I_j), H^c)$ as domains. The average of the calculated similarities for each incoming and outgoing relations of instances give rise to the relation similarity between instances. $OR(I_i, I_j, P, DIR)$ denotes the similarity for relation P and direction DIR between instances I_i and I_j where $DIR \in IN, OUT$ and can be calculated putting into consideration the associated instances of Ii and Ij with respect to P and DIR. For example, in the similarity of relation *hasDirector* and direction *OUT* between two movie instances in Movie Ontology, the directors of the two movies are considered. In similar fashion, the similarity of relation *hasDirector* and direction *IN* between two directors, the movies are considered. Associated instances (As) of instance In with respect to the relation P and direction DIR is the following:

$$A_S(P, I_n, DIR) = \begin{cases} \{I_k : I_k \in \tilde{I} \wedge (P(I_k, I_n))\}, & \text{if } DIR = IN \\ \{I_k : I_k \in \tilde{I} \wedge (P(I_n, I_k))\}, & \text{if } DIR = OUT \end{cases} \tag{17}$$

As (P, In, DIR) is defined as the related instances (As) of instance In with regard to the relation P and direction $OR\ (Ii, Ij, P, DIR)$ calculation and DIR is reduced to similarity between two sets with associated instances.

$$OR(I_i, I_j, DIR) = \begin{cases} 0, & \text{if } (A_s(P, I_i, DIR) = 0 \vee A_s(P, I_i, DIR) = 0) \\ SSIM(A_s(P, I_i, DIR), A_s(P, I_j, DIR)) & \text{otherwise} \end{cases} \tag{18}$$

Recalling what was said in previous sections that similarities between elements triggers the similarity between two sets (SSIM) using a method. RS is used when calculating SSs between two instances and SSs is employed in calculating RS s between instances, this leads to infinite cycles and the to avert this, a maximum recursion depth has to be defined.

Relation similarity is advantageous because similarities between associated instances are given prominence. In a movie instance, the associated instances are feature-values of these movies. In a movie that has only one feature, the actor starred in the movie, and decided to find similarity between *MovieX* and *MovieY* having feature-value Actor α and Actor β respectively. With the user rating movies casting only Actor α, predicting the rating of *Movie Y* becomes impossible has stated. The relation similarity between *MovieX* and *MovieY* depends on the semantic similarity between Actor α and Actor β, and also the semantic similarity between other instances with relations to Actor α and Actor β. As such, similarity value of the movies can be found and rating prediction done.

4.3. Attribute Similarity

For calculating semantic similarities of ontology-based meta data Attribute Similarity (AS) is used as a third similarity measure [38]. Compare to the relation similarity, also attribute values is selected for as between two objects. Hence, AS between two instances I_i and I_j is defined as:

$$AS(I_i, I_j) = \begin{cases} 1, & \text{if } I_i = I_j \\ \dfrac{\sum\limits_{a \in P_A} OA(I_i, I_j, a)}{|P_A|}, & \text{otherwise} \end{cases} \tag{19}$$

PA denotes the set of attributes that includes attributes of both $UC(C(I_i), H^c)$ and $UC(C(I_j), H^c)$. The similarity between objects I_i and I_j is determined by $OA(I_i, I_j, a)$ for attribute a. Thus, attribute similarity between two instances is calculated by computing similarities for each attribute in the set PA and taking average of these similarities. Similar to the computation of $OR(I_i, I_j, a)$, $OA(I_i, I_j, a)$ is calculated by considering associated literals of I_i and I_j with respect to the attribute a. Associated literal (Al) of in regard to the attribute A is as follow:

$$A_l(A, I_n) = \begin{cases} L_x, & if\ \mathrm{L}_x \in \tilde{L} \wedge \mathrm{A}(\mathrm{I}_n, L_x) \\ 0, & otherwise \end{cases} \tag{20}$$

The difference between A_l and A_s is that A_l can include at most one literal unlike A_s. Thus, in order to calculate OA, calculating similarity between attribute values is more preferred rather than calculating similarity between two sets.

$$OA(I_i, I_j, a) = \begin{cases} 0, & if\ (\mathrm{A}_l(\mathrm{A}, I_i) = 0 \vee \mathrm{A}_l(\mathrm{A}, I_j) = 0) \\ LSIM\ (L_i, L_j, a) & otherwise \end{cases} \tag{21}$$

$$L_i = A_l(a, I_i)\ and\ L_j = A_l(a, I_j) \tag{22}$$

5.   RECOMMENDER SYSTEM BASED ON SEMANTIC SIMILARITY

Collaborative filtering applied similarity method for finding K-nearest neighbour users to target user. After that, they utilize the past ratings of neighbour users in order to predict or recommend new content to target user who will like. In this current paper, we use semantic similarity among users to find k-nearest neighbour users. It's worth mentioning that, users profile must be constructed based on ontology. All activities of user can be collected and saved in web proxy. System can classify the records of the user's activities using Machine Learning Algorithm and ontology of the items.

Some attribute of items that a user tries to browse and search can be used to develop the initial user profile ontology. Finally, a user's feedbacks on the results of recommendation can be used as an important act to adjust the user's profile.

In order to develop the profile ontology, items ontology is primarily needed as elaborated in the previous steps. After that, user's interests and preferences are made with regard to the content of the items previously browsed and searched by the user. The ontology generator uses the user's previous activities regarding the various items to develop the initial user profile ontology. Therefore, the user's profile is developed based on the ontology of some reference ontology nodes and each node has an attribute called interest value. This profile is updated with regard to the user's new activities such as shopping, visiting the pages, explicit rating, browsing and searching. The Figure 1 shows the user profiling module used in this study.
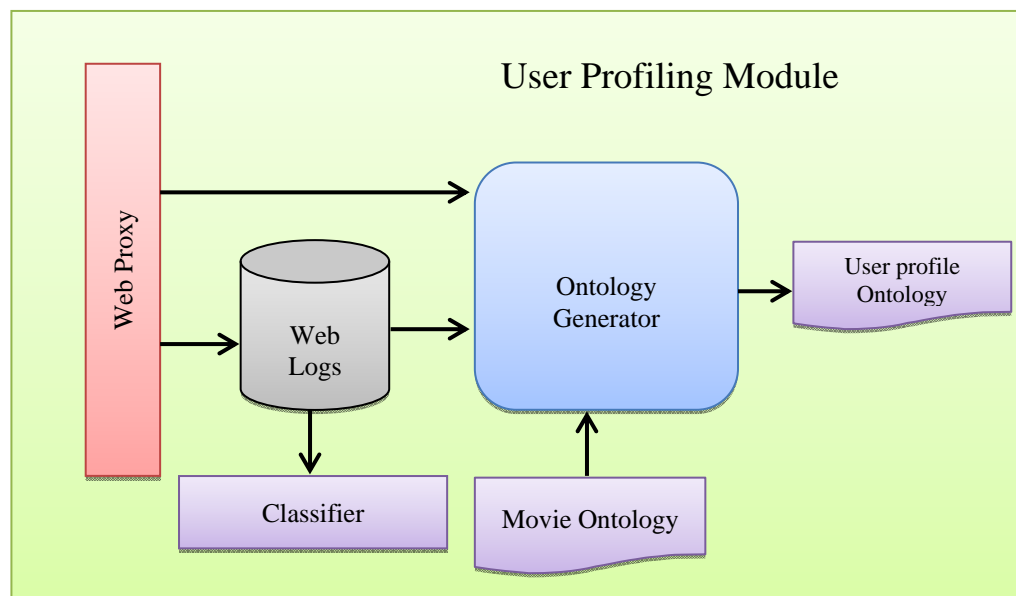

Figure 1. User Profiling Module

In this study for making recommendation list by collaborative filtering, firs K-nearest neighbour of active user (target user) must be gained. For obtaining this result, semantic similarity methods are applied. for obtaining K-NN users to active user, semantic similarity between ontology is used [32]. In this method of similarity, both lexical similarity and conceptual similarity are considered for measuring similarity between two ontologies. Conceptual Comparison Level includes Comparing between two Taxonomies and Comparing Relations between corresponding concepts of two taxonomies. After producing K-nearest neighbour users, all items of this list that neighbour users have purchased but target user has not purchased, recommended to him.

In content-based filtering systems, if items are highly similar to the users' profiles, they can be recommended to user by considering item's content. In this study, content based filtering uses of semantic similarity among items in the item ontology domain in order to anticipate unknown rating for target user based on his/her profile. In this stage, a list including top-N recommendation items are prepared for recommendation to target user based on the user's history record.

6. EVALUATION

In order to evaluate how accurate the proposed methods work in recommender systems, it is better to use the transactions (selling and buying) in a store with various products. In this study, the bills of a construction materials supplier were used. The data include 2266 buyers, 2581 products, and 21662 sales invoices.

To evaluate the recommender system, firstly, the items purchased by each user should be divided into two sets. The first set was called training set and the second one was called "the test set" and sets were selected randomly. The proposed algorithms were first implemented on the training set in order to filter N items to be recommended to users. The N items recommended to the target user are called Top-N. Then, the items in Top-N were compared with the items in the test set. The common items in the test set and Top-N were called Hit Set. After obtaining the test set, training set, and Hit Set, the final step is to determine the accuracy percentage of the algorithm using evaluation criteria. Here, two evaluation criteria called Precision and Recall are used.

$$Precision = \frac{\text{size of hit set}}{\text{size of top-N set}} \tag{23}$$

$$Recall = \frac{\text{size of hit set}}{\text{size of test set}} \tag{24}$$

For a better performance, F1 that is combination of the two above criteria was used:

$$F_1 = \frac{2 * Recall * Percision}{\left(Recall + Percision\right)} \tag{25}$$

F1 was computed for each user and the average F1 obtained from all users was considered as the criterion for determining the algorithm accuracy. In order to compare the proposed methods with the previous methods, they are compared with the recommender system that has been designed based on association rules. The following diagrams show the results of these algorithms. In the following evaluations, the various values of TOP-N were considered from 10 to 130.

Experimental results demonstrate that accuracy of collaborative filtering based on semantic similarity (CF+SeSi) is higher than collaborative filtering based on Pearson correlation similarity (CF+PC) approach. Further, experimental results shows that accuracy of content based filtering based on semantic similarity (CBF+SeSi) is higher than content based filtering based on cosine similarity (CBF+CS) approach (see Figures 2 and 3).
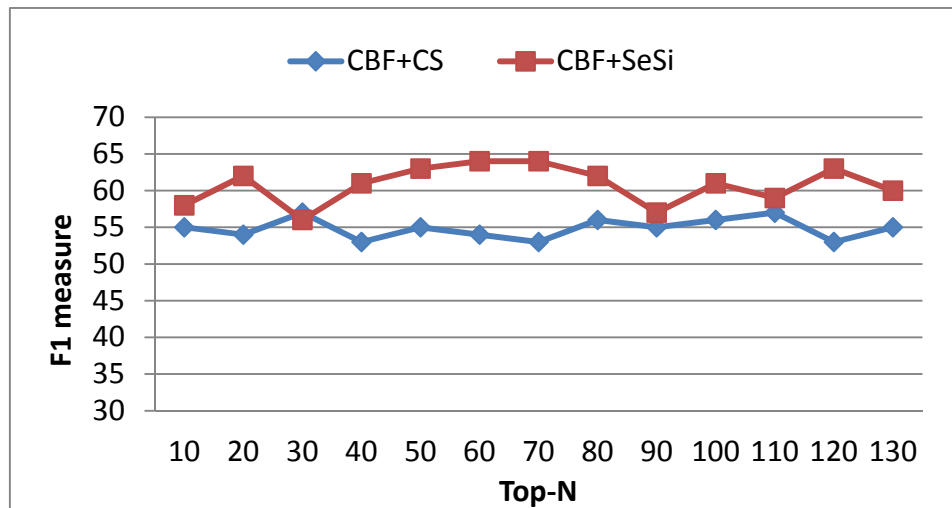
Figure 2. Comparison F1 metric between CBF based on cosine similarity and CBF based on semantic similarity
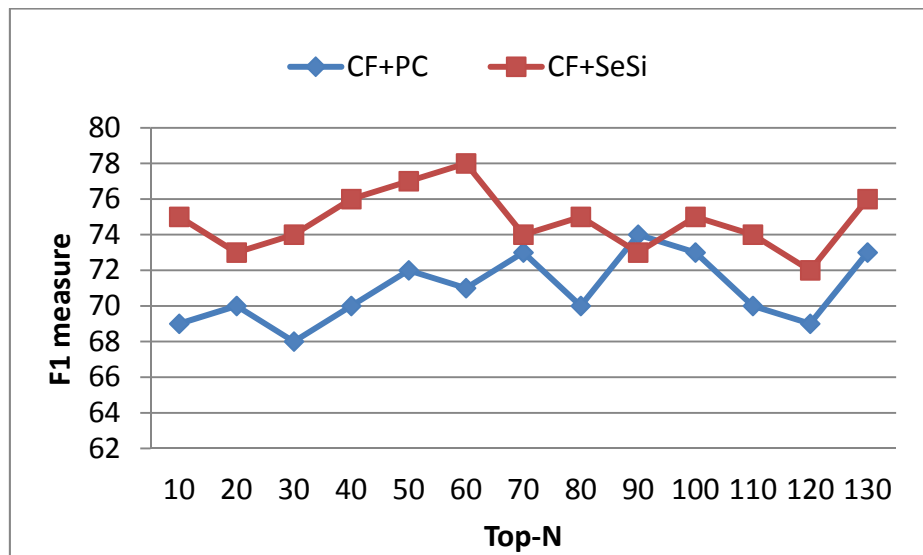

Figure 3. Comparison F1 metric between CF based on Pearson correlation and CBF based on semantic similarity

7.   CONCLUSION

In this paper, we proposed two new recommendation methods by incorporating the semantic similarity in both CF and CBF recommendation approaches. In CF approach, to find a set of k nearest neighbours to the target user, users' profile based on ontology was formed and then semantic similarity among users' profile was used. In CBF approach, for finding similar items to items purchased in the past by target user, semantic similarity was used. Consequently, using most broadly popular measurement metrics, F1, two methods were compared to the CF based on Pearson correlation and CBF based on cosine similarity, respectively.

In order to evaluate how accurate the proposed methods work in recommender systems, we used the transactions (selling and buying) in a store with various products. In this study, the bills of a construction materials supplier were used. In the dataset, there were 2266 buyers, 2581 products and 21662 sales invoices and evaluations were made for the various values of TOP-N from 10 to 130.Experimental results on private building company dataset demonstrated that the high accuracy is obtained in both CBF and CF by incorporating semantic similarity.

ACKNOWLEDGEMENTS

REFERENCES

[1] Bobadilla J, et al. *Recommender systems survey.* Knowledge-Based Systems. 2013.

[2] Rich E. *User modeling via stereotypes.* Cognitive science. 1979; 3(4): 329-354.

[3] Powell MJD. *Approximation theory and methods.* 1981: Cambridge university press.

[4] Salton G. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of.* 1989: Addison-Wesley.

[5] Armstrong JS. *Principles of forecasting: a handbook for researchers and practitioners.* Springer. 2001; 30.

[6] Murthi B and S Sarkar. *The role of the management sciences in research on personalization.* Management Science. 2003; 49(10): 1344-1362.

[7] Lilien GL, P Kotler and KS Moorthy. *Marketing models.* 1992: Prentice-Hall Englewood Cliffs.

[8] Anand SS and B Mobasher. *Intelligent techniques for web personalization.* in Proceedings of the 2003 international conference on Intelligent Techniques for Web Personalization. 2003. Springer-Verlag.

[9] McSherry F and I Mironov. *Differentially private recommender systems: building privacy into the net.* in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. 2009. ACM.

[10] Goldberg D, et al. *Using collaborative filtering to weave an information tapestry.* Communications of the ACM. 1992; 35(12): 61-70.

[11] Resnick P and HR Varian. *Recommender systems.* Communications of the ACM. 1997; 40(3): 56-58.

[12] Schafer JB, J Konstan and J Riedi. *Recommender systems in e-commerce.* in Proceedings of the 1st ACM conference on Electronic commerce. 1999. ACM.

[13] Burke R. Hybrid web recommender systems. in *The adaptive web.* Springer. 2007: 377-408.

[14] Ziegler CN, et al. *Improving recommendation lists through topic diversification.* in Proceedings of the 14th international conference on World Wide Web. 2005. ACM.

[15] Schafer JB, et al. Collaborative filtering recommender systems. in *The adaptive web.* Springer. 2007: 291-324.

[16] Roh TH, KJ Oh and I Han. *The collaborative filtering recommendation based on SOM cluster-indexing CBR.* Expert Systems with Applications. 2003; 25(3): 413-423.

[17] Liu DR, CH Lai, and WJ Lee. *A hybrid of sequential rules and collaborative filtering for product recommendation.* Information Sciences. 2009; 179(20): 3505-3519.

[18] Barragáns-Martínez AB, et al. A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition. *Information Sciences.* 2010; 180(22): 4290-4311.

[19] Burke R. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction.* 2002; 12(4): 331-370.

[20] Ye J. Cosine similarity measures for intuitionistic fuzzy sets and their applications. *Mathematical and Computer Modelling.* 2011; 53(1): 91-97.

[21] Zhu S, et al. Scaling up top-< i> K</i> cosine similarity search. *Data & Knowledge Engineering.* 2011; 70(1): 60-83.

[22] Billsus D and MJ Pazzani. *User modeling for adaptive news access. User modeling and user-adapted interaction.* 2000; 10(2-3): 147-180.

[23] Lang K. *Newsweeder: Learning to filter netnews.* in In Proceedings of the Twelfth International Conference on Machine Learning. Citeseer. 1995.

[24] Breese JS, D Heckerman, and C Kadie. *Empirical analysis of predictive algorithms for collaborative filtering.* in Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc. 1998.

[25] Benesty J, et al. Pearson correlation coefficient. in *Noise reduction in speech processing.* Springer. 2009: 1-4.

[26] Di Lena P and L Margara. Optimal global alignment of signals by maximization of Pearson correlation. *Information Processing Letters.* 2010; 110(16): 679-686.

[27] Schemper M and A Kaider. A new approach to estimate correlation coefficients in the presence of censoring and proportional hazards. *Computational Statistics & Data Analysis.* 1997; 23(4): 467-476.

[28] Herlocker J, JA Konstan and J Riedl. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval.* 2002; 5(4): 287-310.

[29] Herlocker JL, et al. *An algorithmic framework for performing collaborative filtering.* in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM. 1999.

[30] Pedersen T, et al. *Measures of semantic similarity and relatedness in the biomedical domain. Journal of biomedical informatics.* 2007; 40(3): 288-299.

[31] Yin Y and K Yasuda. Similarity coefficient methods applied to the cell formation problem: a taxonomy and review. *International Journal of Production Economics.* 2006; 101(2): 329-352.

[32] Maedche A and S Staab. Measuring similarity between ontologies. in *Knowledge engineering and knowledge management: Ontologies and the semantic web.* Springer. 2002: 251-263.

[33] Wu Z and M Palmer. *Verbs semantics and lexical selection*. in Proceedings of the 32nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics. 1994.

[34] Lin D. *An information-theoretic definition of similarity*. in Proceedings of the 15th international conference on Machine Learning. San Francisco. 1998.

[35] Li Y, ZA Bandar and D McLean. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*. 2003; 15(4): 871-882.

[36] Plaza E, et al. A logical approach to case-based reasoning using fuzzy similarity relations. *Information Sciences.* 1998; 106(1): 105-122.

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

[38] Liu Q, et al. A density-based spatial clustering algorithm considering both spatial proximity and attribute similarity. *Computers & Geosciences*. 2012; 46: 296-309.