2009 Second International Symposium on Electronic Commerce and Security

Joining Case-based Reasoning and Item-based Collaborative Filtering in Recommender Systems

SongJie GONG
Zhejiang Business Technology Institute,
Ningbo 315012, P. R. China
e-mail: songjie_gong@sina.com

***Abstract*—Recommender systems can find user interested information based on the information filtering algorithms. Collaborative filtering technique has been proved to be one of the most successful techniques in recommender systems. And there are two approaches: one is user-based collaborative filtering and the other is item-based collaborative filtering. Data sparsity is the main problem in recommender system, which leads to the bad accuracy. To solve the sparsity problem, this paper presents a personalized recommendation algorithm joining case-based reasoning and item-based collaborative filtering. At first, it employs case-based reasoning technology to fill the vacant ratings of the user-item matrix. And then, it produces prediction of the target user to the target item using item-based collaborative filtering. The recommendation algorithm combining the case-based reasoning and item-based collaborative filtering can alleviate the sparsity issue and can produce more accuracy recommendation than the traditional recommender systems.**

Keywords-recommender systems; item-based collaborative filtering; case-based reasoning

I. Introduction

Recommender systems represent services that aim at predicting a user's interest on information items available in the application domain, using users' ratings on items. Peoples' experiences often do not enough to deal with the vast amount of available information. Thus, methods to help find resources of interest have attracted much attention from both researchers and vendors. Collaborative filtering (CF) technology has proved to be one of the most effective for its simplicity in both theory and implementation [1,2].

The famous electronic commerce website Amazon and CD-Now have employed CF technique to recommend products to customers and it has improved quality and efficiency of their services [3.4]. Collaborative filtering recommender systems are employed in an interactive and iterative manner by their users. The main idea is to compare the user-model of an active user, defined in terms of user preferences and characteristics, with the user models of previous users in order to find k similar users. The historical user models are then used to determine the likely preferences of the active user, and the predicted relevant information content, deemed as personalized information, is provided to the active user.

The sparsity of ratings problem is particularly important in domains with large updated number of items as well as a large number of users [5]. Different treatments are required and different prediction techniques must be employed depending on the sparsity conditions, making the selection of an appropriate approach a cumbersome task. Current CF approaches are limited in the sense that they address specific aspects of the above problem.

To solve the sparsity problem, in this paper, we present a personalized recommendation algorithm joining case-based reasoning and item-based collaborative filtering. At first, it employs case-based reasoning technology to fill the vacant ratings of the user-item matrix where necessary. Then, it produces prediction of the target user to the target item using item-based collaborative filtering. The recommender algorithm combining the case-based reasoning and item-based collaborative filtering can alleviate the sparsity issue and can produce more accuracy recommendation than the traditional CF systems.

II. Using case-based reasoning to fill the vacant ratings

A. *Case-based reasoning(CBR)*

The artificial intelligence based reasoning paradigm of case-based reasoning (CBR) provides analogy based recommendations based on historical models or past experiences [3]. It is applied to solve similar new problems. Typically, CBR recommends the entire solution of previous cases as the solution to the new case, despite any inherent dissimilarity between the new and past cases. It makes direct use of past experiences or cases to solve a new problem by recognizing its similarity with a specific known problem and by applying to find a solution for the current situation.

The main advantages of the CBR over other techniques are as follows [4]: First, in the CBR system, most knowledge is acquired in the case database and so it reduces the knowledge acquisition effort. That is, it makes use of existing case database, so it requires less general knowledge which is very difficult to get. Second, it requires less maintenance effort. Since rule bases or models should consider many dependencies between rules and effects of changes of the rule base are hard to predict, it is difficult to maintain. However, case bases are easier to maintain, because cases are independent of each other, domain experts and novices understand cases quite easily and maintenance of the CBR system can be done.

B. Using CBR to fill vacant ratings

We use the case-based reasoning technology to fill the vacant ratings in the original user-item matrix, as figure 1 shows.

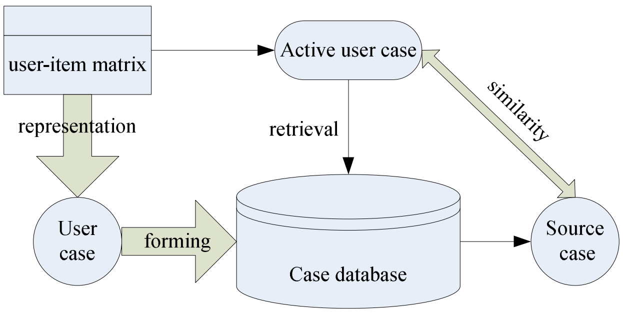

Figure 1. Using CBR to fill vacant ratings in the original user-item matrix

First, the user-item matrix represents as the user cases, and the user cased form the case database. Then, active user case retrieval searches the case database to select existing cases sharing significant features with the new case. Through the retrieval step, similar cases that are potentially useful to the current problem are retrieved from the case database.

The computing of the degree of similarity between the active user case and the source user case can usually be calculated using various similarity functions. In this paper, we use the Euclidean distance as follows.

$$sim(X,Y)=\sqrt{\sum_{i=1}^{n} w_i*(x_i-y_i)*(x_i-y_i)} \qquad (1)$$

Where wi is the weight of the ith feature, xi is the value of the ith feature for the input case, yi is the value of the ith feature for the retrieved case, and n is the total features.

III. USING ITEM BASED CF TO PRODUCE RECOMMENDATIONS

A. Measuring the item rating similarity

There are several similarity algorithms that have been used [2,3,4]: Pearson correlation, cosine vector similarity, adjusted cosine vector similarity, mean-squared difference and Spearman correlation.

Pearson's correlation, as following formula, measures the linear correlation between two vectors of ratings as the target item t and the remaining item r.

$$sim(t,r)=\frac{\sum_{i=1}^{m}(R_{it}-A_t)(R_{ir}-A_r)}{\sqrt{\sum_{i=1}^{m}(R_{it}-A_t)^2\sum_{i=1}^{m}(R_{ir}-A_r)^2}} \qquad (2)$$

Where R_{it} is the rating of the target item t by user i, R_{ir} is the rating of the remaining item r by user i, A_t is the average rating of the target item t for all the co-rated users, A_r is the average rating of the remaining item r for all the co-rated users, and m is the number of all rating users to the item t and item r.

The cosine measure, as following formula, looks at the angle between two vectors of ratings as the target item t and the remaining item r.

$$sim(t,r)=\frac{\sum_{i=1}^{m}R_{it}R_{ir}}{\sqrt{\sum_{i=1}^{m}R_{it}^{2}\sum_{i=1}^{m}R_{ir}^{2}}} \qquad (3)$$

Where R_{it} is the rating of the target item t by user i, R_{ir} is the rating of the remaining item r by user i, and m is the number of all rating users to the item t and item r.

The adjusted cosine, as following formula, is used for similarity among items where the difference in each user's use of the rating scale is taken into account.

$$sim(t,r)=\frac{\sum_{i=1}^{m}(R_{it}-A_i)(R_{ir}-A_i)}{\sqrt{\sum_{i=1}^{m}(R_{it}-A_i)^2\sum_{i=1}^{m}(R_{ir}-A_i)^2}} \qquad (4)$$

Where R_{it} is the rating of the target item t by user i, R_{ir} is the rating of the remaining item r by user i, A_i is the average rating of user i for all the co-rated items, and m is the number of all rating users to the item t and item r.

B. Prediction using item-based CF

Since we have got the membership of item, we can calculate the weighted average of neighbors' ratings, weighted by their similarity to the target item.

The rating of the target user u to the target item t is as following:

$$P_{ut}=\frac{\sum_{i=1}^{c}R_{ui}\times sim(t,i)}{\sum_{i=1}^{c}sim(t,i)} \qquad (5)$$

Where Rui is the rating of the target user u to the neighbour item i, sim(t, i) is the similarity of the target item t and the neighbour item i, and c is the number of the neighbours.

IV. EXPERIMENTAL EVALUATION AND RESULTS

A. Data set

We use MovieLens collaborative filtering data set to evaluate the performance of proposed algorithm [6]. MovieLens data sets were collected by the GroupLens

Research Project at the University of Minnesota and MovieLens is a web-based research recommender system that debuted in Fall 1997. Each week hundreds of users visit MovieLens to rate and receive recommendations for movies. The site now has over 45000 users who have expressed opinions on 6600 different movies. The ratings are on a numeric five-point scale with 1 and 2 representing negative ratings, 4 and 5 representing positive ratings, and 3 indicating ambivalence.

B. Performance measurement

The metrics for evaluating the accuracy of a prediction algorithm can be divided into two main categories [7]: statistical accuracy metrics and decision-support metrics. Statistical accuracy metrics evaluate the accuracy of a predictor by comparing predicted values with user provided values. Decision-support accuracy measures how well predictions help user select high-quality items. In this paper, we use decision-support accuracy measures.

Decision support accuracy metrics evaluate how effective a prediction engine is at helping a user select high-quality items from the set of all items. The receiver operating characteristic (ROC) sensitivity is an example of the decision support accuracy metric. The metric indicates how effectively the system can steer users towards highly-rated items and away from low-rated ones. We use ROC-4 measure as the evaluation metric. Assume that p1, p2, p3, ..., pn is the prediction of users' ratings, and the corresponding real ratings data set of users is q1, q2, q3, ..., qn. See the ROC-4 definition as following:

$$ROC\text{-}4 = \frac{\sum_{i=1}^{n} u_i}{\sum_{i=1}^{n} v_i} \tag{6}$$

$$u_i = \begin{cases} 1, & p_i \geq 4 \ \ and \ \ q_i \geq 4 \\ 0, & otherwise \end{cases} \tag{7}$$

$$v_i = \begin{cases} 1, & p_i \geq 4 \\ 0, & otherwise \end{cases} \tag{8}$$

The larger the ROC-4, the more accurate the predictions would be, allowing for better recommendations to be formulated.

C. Comparing the proposed CF with the user based CF

In this paper, we compare the proposed CF that combining the case-based reasoning and the item-based CF with the only utilizing the user based CF. As showing in the Figure 2, it includes the decision support accuracy metrics of ROC-4 for the two comparing methods in relation to the different numbers of recommender items. The obvious conclusion is that the combining method is better than only using the user based CF.

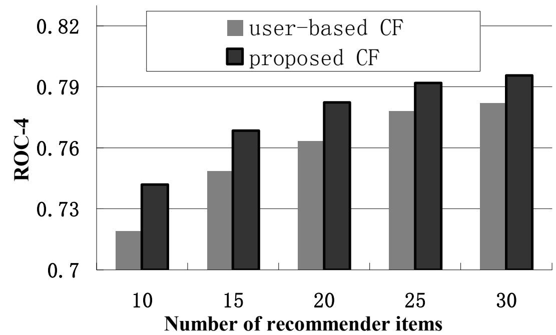

Figure 2. Comparing the two algorithms

V. CONCLUSIONS

In this paper, we present a personalized recommendation algorithm joining case-based reasoning and item-based collaborative filtering. At first, it employs case-based reasoning technology to fill the vacant ratings of the user-item matrix where necessary. Then, it produces prediction of the target user to the target item using item-based collaborative filtering. The recommender algorithm combining the case-based reasoning and item-based collaborative filtering can alleviate the sparsity issue and can produce more accuracy recommendation than the traditional CF systems.

ACKNOWLEDGMENT

A Project Supported by Scientific Research Fund of Zhejiang Provincial Education Department (Grant No. Y200806038).

REFERENCES

[1] Yu Li, Liu Lu, Li Xuefeng, A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce, Expert Systems with Applications 28 (2005) 67–77.

[2] JIANG Lihong, XU Boyi, XI Junhong, CBR filtering algorithm and intelligent information recommending system, J Tsinghua Univ (Sci & Tech) , 2006, 46(S1):1074-1077.

[3] Zeina Chedrawy, Syed Sibte Raza Abidi, An Intelligent Knowledge Sharing Strategy Featuring Item-Based Collaborative Filtering and Case Based Reasoning, 5th International Conference on Intelligent Systems Design and Applications (ISDA'05), 2005,pp.67-72.

[4] Tae Hyup Roh, Kyong Joo Oh, Ingoo Han, The collaborative filtering recommendation based on SOM cluster-indexing CBR, Expert System with Application 25(2003) 413-423.

[5] Chong-Ben Huang, Song-Jie Gong, Employing rough set theory to alleviate the sparsity issue in recommender system, In: Proceeding of the Seventh International Conference on Machine Learning and Cybernetics (ICMLC2008), IEEE Press, 2008, pp.1610-1614.

[6] Songjie Gong, Chongben Huang, Employing Fuzzy Clustering to Alleviate the Sparsity Issue in Collaborative Filtering Recommendation Algorithms, In: Proceeding of 2008 International Pre-Olympic Congress on Computer Science, World Academic Press, 2008, pp.449-454.

[7] Gao Fengrong, Xing Chunxiao, Du Xiaoyong, Wang Shan, Personalized Service System Based on Hybrid Filtering for Digital Library, Tsinghua Science and Technology, Volume 12, Number 1, February 2007,1-8.