AN OPTIMIZED ITEM-BASED COLLABORATIVE FILTERING RECOMMENDATION ALGORITHM

Jinbo Zhang, Zhiqing Lin, Bo Xiao, Chuang Zhang

Pattern Recognition & Intelligent System Lab,
Beijing University of Posts and Telecommunications, Beijing
boboalwaystry@163.com,linzq@bupt.edu.cn,xiaobo@bupt.edu.cn,zhangchuang@bupt.edu.cn

Abstract

Collaborative filtering is a very important technology in E-commerce. Unfortunately, with the increase of users and commodities, the user rating data is extremely sparse, which leads to the low efficient collaborative filtering recommendation system. To address these issues, an optimized collaborative filtering recommendation algorithm based on item is proposed. While calculating the similarity of two items, we obtain the ratio of users who rated both items to those who rated each of them. The ratio is taken into account in this method. The experimental results show that the proposed algorithm can improve the quality of collaborative filtering.

Keywords: Personalized Recommendation; Item-based collaborative filtering; item similarity; MAE

1 Introduction

In recent years, with the rapid development of Internet, the online resources have increased sharply, which makes it more difficult for users to find the information they need than before. Therefore, it is necessary to find a recommendation system that can help us extract the most valuable information from all the information, that is why the Personalized Recommendation is used. Many of us may be familiar with Personalized Recommendation, for example, we can see "Related Articles" in the bottom or on the side while browsing a page.

Currently, many E-commerce sites have used the recommended system, such as Amazon, CDNOW, etc. Data mining technology is used to help customers access their favorite information or products in E-commerce sites. Recommendation system makes significant contributions to expand sales revenue.

Collaborative filtering is one of the most successful technologies in Personalized Recommendation. Collaborative filtering works through the establishment of a database of preferences for items by users, the basic idea of CF(collaborative filtering)-based algorithms is to provide item recommendations or predictions based on the opinions of other like-minded users[3]. According to the historical ratings that target user rated and the ratings other users rated, we can predict the ratings that target user haven't rated, resulting in the recommendation.

In order to recommend accurately, researchers proposed several recommendation algorithms, such as user-based collaborative filtering algorithm, item-based collaborative filtering algorithm, Bayesian networks, clustering, etc.

User-based algorithm is based on such an assumption: A good way to find interesting content is to find other people who have similar interests, and then recommend titles that those similar users like[2]. Due to the extremely sparse of data, the items that two users both rated may be none, which results in that the similarity between users is likely to be 0, so item-based collaborative filtering algorithm is proposed. Item-based collaborative filtering algorithm is based on such an assumption: If the majority of users rate some items similarly, the target user will rate the items similarly[4].

Unfortunately, with the enlarger of E-commerce systems' size, the number of users and commodities is increasing, the user rating data is extreme sparse. In the case of extremely sparse data, traditional similarity measurement methods all have their own limitations, which lead to the low efficient collaborative filtering recommendation system.

To improve the quality of the recommendations, this paper presents an optimized item-based collaborative commendation algorithm. The quality of the recommendation is related to the accuracy of the prediction rating score. We search for the similar items of target item, through the score of the similar items that target user rates, we can get the prediction rating. While calculating the similarity of items, we take the ratio of users that rated both two items to those rated each item into account. Experimental results show that the proposed algorithm can improve the quality of collaborative

978-1-4244-4900-2/09/$25.00 ©2009 IEEE

filtering better than the traditional item-based collaborative filtering algorithm does.

2 Related work about item-based collaborative filtering algorithm

The goal of a collaborative filtering algorithm is to suggest new items or to predict the utility of a certain item for a particular user based on the user's previous likings and the opinions of other like-minded users[3].

CF algorithms use a ratings matrix, A, to represent the entire $m \times n$ user-item data, line m represents the m-th user, column n represents the n-th item, each entry $R_{i,j}$ in the line i and column j represents the rating of the target item j by user i, which can expressed his/her opinions about the item within a certain numerical scale. In general, $0 \le R_{i,j} \le 5$, the higher rating score is, the more preference user like the item. Table 1 shows the ratings matrix.

Table 1. User rating data matrix

	$Item_1$	…	$Item_j$	…	$Item_n$
$User_1$	$R_{1,1}$	…	$R_{1,j}$	…	$R_{1,n}$
…	…	…	…	…	…
$User_i$	$R_{i,1}$	…	$R_{i,j}$	…	$R_{i,n}$
…	…	…	…	…	…
$User_m$	$R_{m,1}$	…	$R_{m,j}$	…	$R_{m,n}$

2.1 Item Similarity computation

There are many ways to compute the similarity between items, including the following three methods: cosine-based similarity, correlation- based similarity and adjusted-cosine similarity.

1) Cosine-based Similarity(cosine):Rating for items are thought of as vectors. If user don't rate some items, we set these rating scores 0[1]. The similarity between two items can be measured by computing the cosine of the angle between the two vectors[3]. The ratings of the i-th item and j-th item in the m dimensional user-space can be expressed as $\vec{i}$, $\vec{j}$,the similarity between items i and j can be computed as the following formula.

$$sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\left\|\vec{i}\right\| \times \left\|\vec{j}\right\|} \quad (1)$$

Where " " denotes the dot-product of the two vectors.

2) Correlation-based Similarity(correlation):First, we find the co-rated cases[3](which means cases where the users who rated both items i and j for example), and set the users assembly $U_{i,j}$ who both rated i and j, the correlation similarity between items i and j can be computed as the following formula.

$$sim(i, j) = \frac{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_i})(R_{u,j} - \overline{R_j})}{\sqrt{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_i})^2} \sqrt{\sum_{u \in U_{ij}} (R_{u,j} - \overline{R_j})^2}} \quad (2)$$

In this formula, $R_{u,i}$ denotes the rating of user u on item i, $R_{u,j}$ denotes the rating of user u on item j, $\overline{R_i}$ is the average rating of the i-th item, $\overline{R_j}$ is the average rating of the j-th item.

3) Adjusted Cosine Similarity(adjusted cosine): This similarity among items takes the difference in rating scale between different users into account. It can be computed as the following formula.

$$sim(i, j) = \frac{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_u})(R_{u,j} - \overline{R_u})}{\sqrt{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_u})^2} \sqrt{\sum_{u \in U_{ij}} (R_{u,j} - \overline{R_u})^2}} \quad (3)$$

Where $\overline{R_u}$ is the average of the u-th user's ratings.

2.2 Prediction Computation

This is the most important step in the collaborative filtering system. In order to recommend items to users, we must predict the rating score the target user will rate the item. If the prediction is very high, that means the target user may be interest in the item.

$P_{u,i}$[3] that the user u predict item i is given by

$$P_{u,i} = \frac{\sum_{all\_similar\_item,N} (S_{i,N} * R_{u,N})}{\sum_{all\_similar\_item,N} (\left|S_{i,N}\right|)} \quad (4)$$

Where $S_{i,N}$ denotes the similarity between the target item i and the neighbor item j. In order to find all similar items, we could pick the top k items which are the most similar to the target item.

2.3 Item Similarity Analysis

With the expansion of E-commerce system's size, the number of users and items is rapidly increasing, the rating data is extremely sparse. In most large E-commerce system, active users may rate items less than 1% of the whole items[3], the co-rated items is even less.

We take an example using the above-mentioned item similarity formula.

Table 2. A rating matrix example

	I_1	I_2	I_3
U_1	2	/	3
U_2	5	/	5
U_3	3	3	3
U_4	/	5	/
U_5	/	2	3

We use formula (2) as the similarity computation, and can get:

$$sim(I_1, I_2) = 1$$

$$sim(I_1, I_3) = 0.89$$

We can find that the similarity between I_1 and I_2 is bigger than the similarity between I_1 and I_3 just because the number of co-rate items is smaller. Apparently, this is not realistic nor accurate. More unaccepted is that we think two items' similarity is 1 if only one user rate both the two items and rate them the same score using this method.

So in the case of extreme sparse rating data, item-based collaborative filtering is not accurate.

3 An optimized item-based collaborative filtering recommendation algorithm

To address the above-mentioned issues, we propose an optimized item-based collaborative filtering recommendation algorithm. We take the percentage of the co-rated items into account. While compute the similarity between items *i* and *j*, we first count the number of users that rate both items *i* and *j*, and record it as *N*, then count the user's number of users that rate item *i* or *j*, and record it as *M*. Each similarity is weighted by an function $f(\frac{N}{M})$ that relates to the ratio of the co-rated items' number to the corresponding items' total number. Therefore, the optimized similarity is as follows:

$$sim'(i, j) = f(\frac{N}{M})\ sim(i, j) \tag{5}$$

Where the $f(\frac{N}{M}) = 1 - \alpha\ (1 - \frac{N}{M})$. The parameter $\alpha \in [0,1]$, is determined as the weight of the percentage of the co-rated items. If we select α=0, we can get $sim'(i, j) = sim(i, j)$, that is to say, the new similarity is the same as the traditional similarity. Therefore, the optimized corresponding similarities of the three item similarities above are as follow:

$$sim'(i, j) = (1 - \alpha + \alpha\ \frac{N}{M})\ \frac{\vec{i}\ \vec{j}}{\|\vec{i}\| \times \|\vec{j}\|} \tag{6}$$

$$sim'(i, j) = \left[1 - \alpha\ (1 - \frac{N}{M})\right] \frac{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_i})(R_{u,j} - \overline{R_j})}{\sqrt{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_i})^2} \sqrt{\sum_{u \in U_{ij}} (R_{u,j} - \overline{R_j})^2}} \tag{7}$$

$$sim'(i, j) = \left[1 - \alpha\ (1 - \frac{N}{M})\right] \frac{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_u})(R_{u,j} - \overline{R_u})}{\sqrt{\sum_{u \in U_{ij}} (R_{u,i} - \overline{R_u})^2} \sqrt{\sum_{u \in U_{ij}} (R_{u,j} - \overline{R_u})^2}} \tag{8}$$

Again, we take the figure 2 as example, select α=1, and can get the similarities.

$$sim'(I_1, I_2) = 0.2$$

$$sim'(I_1, I_3) = 0.6675$$

We can see that the similarity between I_1 and I_3 is bigger than the similarity between I_1 and I_2.

4 Experimental evaluation

4.1 Data set

In order to verify the effectiveness of the algorithm, we use data provided by MovieLens recommender system(http://movielens.umn.edu/), which is a web-based research recommender system developed by GroupLens research group in University of Minnesota. Users visit MovieLens to rate and receive recommendation for movies. The site now has over 45000 users who have expressed opinions on 6600 different movies[5].

This paper selects the *m1* data set provided by MoiveLens, it contains 100,000 ratings ranging from 1 to 5, which are rated by 943 users to 1682 movies, each user rated at least 20 movies. We divided the data set into a training set and a test set, 80,000 ratings was used as training set and 20,000 ratings was used as test set.

To measure the sparse degree of data set, we take another factor into consideration, sparse level of data set is defined as the percentage of no rating entries in the total entries. The sparse level of the movie data set is:

$$1 - \frac{10000}{943 \times 1682} = 0.937$$

4.2 Evaluation Metrics

The measure for evaluating the quality of recommender systems mainly contains two categories: statistical accuracy metrics, decision-support measure metrics[11]. In this paper, we use the statistical accuracy metrics. One of the statistical accuracy metrics frequently used is Mean Absolute Error(MAE)[5], MAE is easier to be understood, and can measure the quality of the recommendation system directly, it is a widely used metric.

The paper uses MAE as the measure to evaluate the quality of the recommender systems. MAE computes the predictions against the actual user ratings. We record the collection of predictions as $\{p_1, p_2, ..., p_N\}$, the corresponding actual user ratings as $\{q_1, q_2, ..., q_N\}$. Formally,

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \tag{9}$$

4.3 Experimental Results

There are three item similarity measures in the proposed algorithm. We will test the three item similarity measures with different α on different

neighborhood sizes using the same data set. While α=0.1, the proposed optimized item-based collaborative filtering recommendation algorithm is the same as the traditional. We use MAE as the measure to evaluate the quality of the recommender systems, the neighborhood size ranges from 4 to 20 with an increment of 4, our results are shown in Figures 1-3.

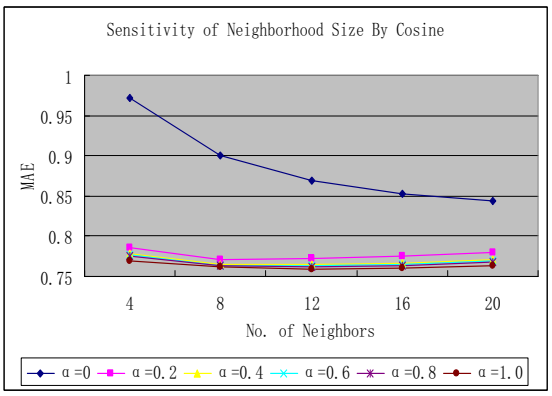

Figure 1. Comparison of Cosine similarity of different α

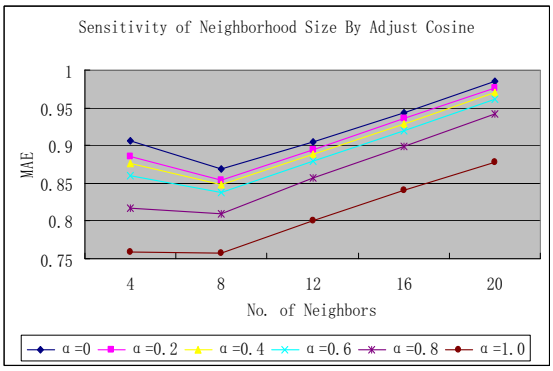

Figure 2. Comparison of Adjust Cosine similarity of different α

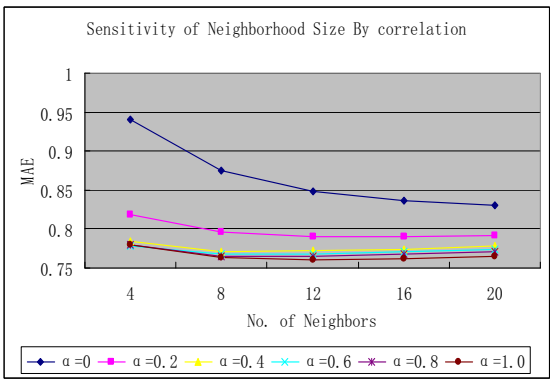

Figure 3. Comparison of Correlation similarity of different α

From the figures, we can see MAE gets less with the bigger of α no matter which similarity measure we select. Therefore, if we choose α=1, we can get the least MAE, that is to say, if we use $sim'(i,j) = \frac{N}{M} sim(i,j)$ as the similarity algorithm, we can get best effect in the collaborative filtering recommendation system.

5 Conclusions

This paper first analyzes the problem of Cosine-based Similarity, Correlation-based Similarity, and Adjusted Cosine Similarity. Then the paper proposes an optimized algorithm, which can solve the problem and find the neighbor items of target item more accurately. The experimental results show that the optimized item-based collaborative filtering algorithm can solve the drawback of the traditional item-based collaborative filtering algorithm in the case of extreme rating data sparse, and improve the quality of the recommendation system apparently.

Acknowledgements

This work was supported by the national High-tech Research and Development Plan of China under grant No. 2007AA01Z417 and the 111 Project of China under grant No. B08004.

References

[1] Deng Ai-lin, Zhu Yang-yong, Shi Bo-le. A collaborative filtering recommendation algorithm based on item rating prediction. Journal of Software, 2003

[2] John S. Breese, David Heckerman, Carl Kadie. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Proc of the 14th Conf on Uncertainty in Artificial Intelligence. Madison. WI: Morgan Kaufmann Publisher, 1998

[3] Sarwar B, Karypis G, Konstan J. Item-based collaborative filtering recommendation algorithms. Proceeding of the 10th International World Wide Web Conference. 2001

[4] Wu Yan, Shen Jie, Gu Tianzhu, CHEN Xiaohong, LI Hui, ZHANG Shu. Algorithm for Sparse Problem in Collaborative Filtering. In: Application Research of Computers. Vol.24 No.6. June, 2007.

[5] Gong Song-jie, Ye Hong-wu. An Item Based Collaborative Filtering Using BP Neural Networks Prediction. 2009 International Conference on Industrial and Information Systems

[6] Kwok-Wai Cheung,LilyF. Tian, Learning User Similarity and Rating Style for Collaborative Recommendation, Information Retrieval, 2004

[7] Manos Papagelis, Dimitris Plexousakis, Qualitative analysis of user-based and item-

based prediction algorithms for recommendation agents. Engineering Application of Artificial Intelligence 18(2005).
[8] Sarwar B, Karypis G, Konstan J, Riedl J. Analysis of recommendation algorithms for E-commerce. ACM Conference on Electronic Commerce,2000
[9] Xing Chun-xiao, Gao Feng-rong, Zhan Si-nan, Zhou Li-zhu. A Collaborative Filtering Recommendation Algorithm Incorporated with User Interest Change. Journal of Computer Research and Development,2007
[10] G Karypis. Evaluation of item-based top-N recommendation algorithms[c]. Proc of CIKM 2001.New York: ACM Press, 2001
[11] M.G. Vozalis, K.G.Margaritis. Using SVD and demographic data for the enhancement of generalized Collaborative Filtering. Infromation Sciences, 2007.