Recommendation Model Based on Collaborative Filtering Recommendation Algorithm

Jun Huang
Communications and Information Engineering College, Yunnan Open University, Kunming 650223, Yunnan, China

ABSTRACT: There are problems concern the current recommendation model such as the information recommended is not inaccurate enough. This paper presents a collaborative filtering algorithm based on K-means algorithm. Firstly, we analyzed the similarity calculation method of collaborative filtering recommendation algorithm, then we proposed a valuation formula based on user rating scale and information popularity to assign value for ungraded items at sparse ratings matrices to improve the scoring matrix density, increase the accuracy of similarity calculation, and build the recommendation model. Simulation results show that the proposed collaborative filtering recommendation algorithm based on K-means has higher prediction accuracy and classification accuracy than traditional collaborative filtering algorithm.
KEYWORD: Collaborative Filtering; Recommendation; User Rating Scale; Welcome Degree Valuation; Sparse Matrix Evaluation

1 INTRODUCTION

The idea of collaborative filtering was first proposed in 1992 by Glodberg et al, which greatly promoted the research and development of recommendation models (Zhang F, 2005). In recent years, with the rise of big data, recommendation system has been more widely used, which has injected new vitality into the study of recommendation model. Amazon, eBay, and Taobao have all adopted the intelligent recomme ndation model to provide personalized recomme ndation service for users (Goldberg D, 199 2).

The basic idea of collaborative filtering is to use "wisdom of the crowd" to filter information. And the basic assumption is that the information needs of users with the same or similar interest preferences is similar. In the early collaborative filtering technology, the system only makes recommendations after users know each other's hobby (Salwar B M, 2000); then as the study going further, automated collaborative filtering system has been developed, of which GroupLens system is a typical representative developed by GroupLens (Soboroff I, 1999) of the Minnesota State University in the United States. The biggest problem faced by collaborative filtering algorithm is the data sparseness problem. In order to solve the data sparseness problem, Sarwar and others using the singular value decomposition (SVD) method to reduce dimentions of user - item rating matrix to obtain relatively dense data. However, it will decrease the recommendation accuracy. Zhang et al proposed to use neural network filling method to solve the data sparseness problem. But the biggest drawback of this matrix filling technique is the scalability problems which will have great amount of calculation when the amount of data increase which will lower the recommendation speed. In addition, the dimension reduction techniques for collaborative filtering have other methods as principal component analysis (Principal Component Analysis), LSI (Latent Semantic Indexing). Karypis et al [10] proposed a collaborative filtering algorithm (Item-based CF) based on project. Due to the similarity ratios based on project are stable than those based on users, to a certain extent, it ease the data sparseness problem.

2 COLLABORATIVE FILTERING RECOMMENDATION ALGORITHM BASED ON K-MEANS ALGORITHM

2.1 *User clustering algorithm based on K-means*

This article design an information recommendation system algorithm with at least tens of thousands of users, or maybe a million, which will occur a huge amount of data processing, in order not to increase the scalability issues, low complexity algorithms are used to deal with user clustering processing. Before

user clustering, it is necessary to construct a user rating matrix according to users' historical rating data. Assume the total number of system users is M, the total number of commodities is N, the established user - information ratings matrix is a $M \times N$ matrix.

Goods

Users	S1	S2	S3	…	Sj	…	Sbl
U1	3						
U2	4						2
…			4				
…						4	
Ui	5			3			
…					1		
…				5			4
		5				3	
Usl			2				

Figure1 M×N dimensional user - commodity scoring matrix diagram

We let the row vector of the matrix represents a user, for example, we use a row vector $\{5,0,0,3...4...0\}$ of row i to represent user U_i. Then, we use the K-means algorithm to do clustering process for all number M users and cluster them into k categories.

Using K-means algorithm to do the clustering processing are as follows:

(1) We select k user vectors as the initial cluster centers;

(2) We calculate other users' vector's Euclidean distance to the cluster center and the Euclidean distance is shown as following:

$$Euclidena(U_m, U_n) = (\sum_{i=1}^{N} \left| R_{m,j} - R_{n,j} \right|^{\frac{1}{2}} \quad (1)$$

(3) We classify each user into a cluster center that has the shortest distance to it based on its own distance to k cluster centers. The k categories is shown as $C, D = \{C_1, C_2, C_3, ..., C_{k-1}, C_k\}$

(4) Based on the division of clusters, we recalculate the center of each cluster, we use the average of all user vector as the center of the cluster.

(5) We repeat steps (2), (3), (4), until there is no longer any changes of the center, at which point the user set was divided into k separate clusters for sure.

3 SIMILARITY CALCULATION ALGORITHM DESIGN

Similarity calculation is the most important step of collaborative filtering recommendation algorithm. The accuracy of similarity calculation directly determines the accuracy of the prediction and recommendation of the recommendation algorithm. So choose a good similarity calculation methods is essential for collaborative filtering algorithms.

There are three most common similarity calculation method, namely cosine similarity, modified cosine similarity and correlation coefficient. We compare results of collaborative filtering recommendation algorithm using different similarity calculation method and the experimental results show the collaborative filtering recommendation algorithm that use the modified cosine similarity has the highest recommendation accuracy.

Modified cosine similarity of commodity i and j, is calculated as shown in Formula 5,

$$similarity(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \overline{R_u})(R_{u,j} - \overline{R_u})}{\sqrt{\sum_{u \in U} (R_{u,i} - \overline{R_u})^2} \sqrt{\sum_{u \in U} (R_{u,j} - \overline{R_u})^2}} \quad (2)$$

Here, $R_{u,i}$ and $R_{u,j}$ represent the score user u toward commodity i and j, U represent the whole users set, $\overline{R_u}$ represent user u's average score.

The caculation of the similarity between commodities are as following:

(1) We calculate the average score $\overline{R_u}$ of each user in the origin score matrix.

$$\overline{R_u} = \frac{1}{|U|} \sum_{c \in U} (R_{c,j} - \overline{R_c}) \quad (3)$$

Set U are commodity set that have been scored by user u and $|U|$ represent the commodity number of set U.

(2)We use Cosine similarity formula to calculate similarity between two items, calculation is shown as following

$$similarity(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \overline{R_u})(R_{u,j} - \overline{R_u})}{\sqrt{\sum_{u \in U} (R_{u,i} - \overline{R_u})^2} \sqrt{\sum_{u \in U} (R_{u,j} - \overline{R_u})^2}} \quad (4)$$

$\overline{R_u}$ is the average score by user u, and $R_{u,i}$ is the average rating score by a new user.

(3) After the calculating of the similarity of any two items, we establish a similarity matrix. Assuming there are N items, the similarity matrix will be a $N * N$ symmetric matrix.

4 EMPIRICAL CALCULATION FOR THE RECOMMENDATION MODEL

About user ratings data sets, this article use Movielens data sets developed by University of Minnesota recommendation system research team as empirical data to do simulation. Data set is a collection of user rating data about movies on website Movielens, which is widely used as a data

set for recommendation algorithms study. The range of user ratings toward movies is 1-5 points, scores represent user's preferences. 1 point is the minimum, while 5 point is the maximum. There are 943 users and 1682 merchandises in the data set, generating a total of 100,000 user ratings. Movielens data set consists the training and test sets, of which 80% of the ratings data that 80000 ratings used as the training set to do similarity calculation and rating prediction; while the remaining 20% of the score data, namely 20,000 rating, is used as a test set to compare with the prediction score and measure the accuracy of the algorithm. Further, since the algorithm requires a large amount of matrix operations, the empirical calculations of the algorithm are completed in Matlab.

Simulation design program are as follows:

(1) First, we use matlab to program the collaborative filtering algorithm. At this moment users do not need to cluster. We only predict rates to those ungraded commodity. When we calculate the predicted rating, number of neighbor commodity start to increase at 10 and increase 10 neighbors each time until the number of neighbors hits 100. After we get the predicted rating set, we will compare the test set with the predicted set to calculate the average absolute error (MAE);

(2) After we get 10 MAE points, we can draw a MAE curve related to the change of the number of commodities neighbor curve;

(3) We use the evaluation formula to fill out the rating matrix for the original users, where the filling density increase from 5% to 100% linear with a 5% increments. We calculate different absolute error MAE of predicted rating under different density. Under each density, we should study the impact of the number of neighbors toward MAE, the number of neighbors start to increase from 10, within increments of 10 neighbors, until the number of neighbor arrives 100;

According to the results, we obtain MAE curve over the number of neighbors under different densities,. Figure 2 shows the curve when matrix density increased by 7%, 15%, 20% and when the matrix density did not increase, how collaborative filtering algorithm's mean absolute error (MAE) change with the number of neighbors.change with the number of neighbors.

As can be seen from the figure, you can see the following rules: Recommended accuracy increases with the number of neighbors increases; when the number of neighbors reached 40, the mean absolute error reaches a minimum value; when the number of neighbor continues increases, the average absolute error will not have significant changes any more.

Therefore, it is easy to conclude that when the number of commodity neighbors reach 30, the average absolute error of collaborative filtering reaches minimum level that is the recommendation accuracy reaches maximum.

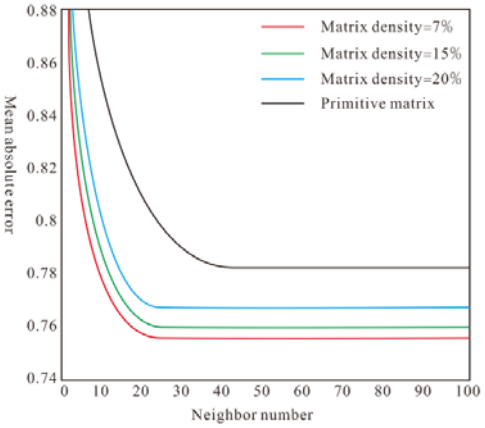

Figure 2 The curve of the mean absolute error with the number of neighbors

Table1 MAE under different K value and matrix density

Clustering number K	Matrix virtual density	MAE after clustering
5	99.9%	0.80105
10	99.2%	0.78544
15	96%	0.77267
20	91.09%	0.75866
25	85.55%	0.74687
30	80.058%	0.74104
35	74.892%	0.74353

We set value k within the K-means algorithm first, and the values of k are shown in Table 1. As the first column shows, the fixed number of neighbors for score prediction are 30 commodities. We then calculate the mean absolute error MAE for different prediction score of value k. The following figure 2 shows the curve that present how mean absolute error MAE change with median number of clustering algorithm in the recommended system.

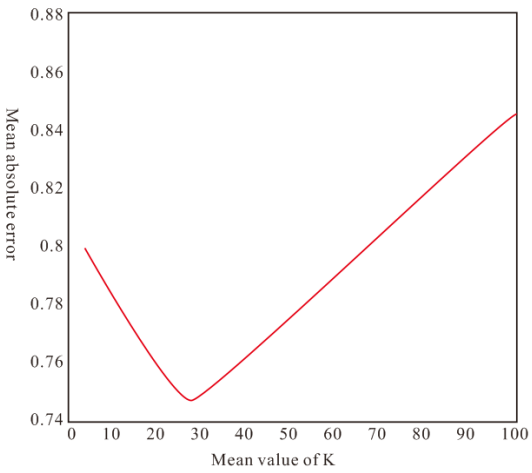

Figure 3 The variation law of the mean absolute error of the recommended system with the K value

As it can be seen from the figure, at the beginning, when k is 1, the users are not clustering. There is only increasing number of the valuation formula. And we let the matrix density reaches the maximum level of 100%. The mean absolute error of calculated forecasting rates is up to 0.79353. With the increase of user clustering number *k*, the average absolute error has been decreased, when k =30, the mean absolute error of this algorithm reaches a minimum value of 0.74104. The average absolute error starts to increase when k value keep increase.

Compared to non-clustering recommendation system, the one did users clustering has improved recommendation accuracy significantly. And when k takes the value between25-30, recommendation accuracy reached its lowest point, The MAE minimum value is 0.74104. Experimental results show that accuracy of the recommendation algorithm is related to k in the clustering algorithm, when k =30, the predictive score a is most accurate and the recommendation accuracy is the highest.

5 CONCLUSIONS

We did empirical analysis to the proposed collaborative filtering recommendation algorithm based on K-means, which includes several major factors that affect the accuracy of the algorithm prediction, such as user product rating matrix density, clustering algorithms and influence of clustering algorithms to the accuracy of prediction. Experimental results show that the users clustering can improve the accuracy of valuation formula, thereby improve the accuracy of product prediction. Experimental results also show that the proposed collaborative filtering recommendation algorithm based on K-means having a higher prediction accuracy and classification accuracy than traditional collaborative filtering algorithms.

REREFERNCES

B.Sarwar, G.Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In Proc. Of the WWW Conference, 2001.

Goldberg D, Niehols D, Oki B M, et al. Using Collaborative Filtering to Weave an Information Tapestry[C]. Communications of the ACM, 1992, 35(12):61-70.

Goldberg K, Roeder T, Gupta D, et al. Eigentaste: A Constant Time Collaborative Filtering Algorithm [J]. Information Retrieval, 2001, 4(2): 133-151.

Resnick P, Iakovou N, Sushak M, et al. Group Lens: An open architecture for collaborative filtering of net news. Proc 1994 Computer Supported Cooperative Work Conf, Chapel Hill, 1994: 175 -186.

Salwar B M, Karypis G, Konstan J A, Riedl J. Application of Dimensionality Reduction in Recommender System-A Case Study.In ACM 2000 KDD Workshop on Web Mining for e-commerce-Challenges and Opportunities, Boston, MA,2000.

Soboroff I, Nicholas C. Combining content and collaboration in text filtering. In: Proceedings of the International Joint Conferences on Artificial Intelligence Workshop: Machine Learning for Information filtering, Stockholm,1999,86-91

Zhang F, Chang H. A collaborative filtering algorithm embedded BP network to ameliorate sparsity issue[C]. Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on IEEE, 2005,3:1839-1844.