A Collaborative Filtering Algorithm Based on Double Clustering and User Trust

Tonglong Tang[1, a *], Xiaoyu Li[1, b]

[1]School of Information Engineering, Zhengzhou Unversity, China, 450001

[a]532697721@qq.com, [b]iexyli@zzu.edu.cn

Keywords: Collaborative filtering algorithm; Double clustering; User trust; Score prediction; Recommender system.

Abstract. A collaborative filtering algorithm based on double clustering and user trust to solve data sparse and cold start problem is present. This algorithm uses user-clustering matrix to measure the user's degree of similarity, which could reduce the dimension of the user-item matrix. On the other hand it uses user level trust to perform predictions in rating predicting step. The experiments results show that this method could relieve the sparsity problem and improve the accuracy of the prediction results.

Introduction

With the rapid progress of computer technology, electronic commerce and social network are more and more popular in society. Massive product information and knowledge appears in Internet and brings great convenience to people's work and life. But too much information causes information overload problem. That is to say, people can't find useful information from huge access information. So recommender system emerges as the times require.

Since Gold Berg provided the first recommender system Tapestry. More and more researchers have devoted to the study field. Now collaborative filtering technology is widely applied in electronic commerce. The traditional collaborative filtering schemes are based on user-item matrix. But some statistics show that the items commented by user are often no more than 2 percent of all the product items in an electronic commerce website [2]. This is the sparsity problem which makes the recommendation unreliable. On the other hand collaborative filtering algorithm needs the records of the users' historical behavior. But there are no records for a new user so that no recommendations can be given which is the cold start problem. To solve these problems some scholars integrated clustering methods with traditional collaborative filtering algorithms [2-4]. Other scholars began to apply user trust to recommend algorithm [5-9]. Although these methods can relieve the sparsity and improve the accuracy of predictions to some extent, they still needs user-item matrix to measure the similarity. So the sparsity problem can't be solved well.

In this paper a collaborative filtering algorithm based on double clustering and trust is present. First we perform clustering to the items. Then we definite the interest measure and integrate it with user trust to get the mixed similarity between users. On the other hand in the clustering step considering the authority of expert users we take them as the clustering center. Next in prediction step we use user level trust to predict the score because different user's score should be given different trust degree. Our algorithm can reduce the user-item matrix and the search band of the nearest neighbors. So it can gain higher efficiency and reduce the affection of sparsity. At the same time it can overcome the difficulty of the cold start problem to a considerable degree.

Background

User Trust. We take PKI (Public Key Infrastructure) mode which is issued by Maurer *et al* [10] to represent the trust between users, or in other words, user trust.

$$Trust(u,v)=\frac{\sum_{i=2}^{k} e_{ul_1} e_{l_1 l_2} \cdots e_{l_{k-1}v}}{\sum_{w \in N_u} e_{uw}} \tag{1}$$

in which K represent the maximum length of the path from user μ to user ν. e_{ul_1}, $e_{l_1 l_2}$,…, $e_{l_{k-1}v}$ are the edge of the path with a length of i from user nodeμ to user node ν.As known there are Six Degrees of Separation theory [11] which determine that the path won't be longer than 6 in fact. So we can take *k*=3.

User Item Level Trust. In traditional collaborative filtering algorithm people use the similarity between users to perform scores prediction. In this paper we suggest to use item trust to predict. It can be calculated as follows.

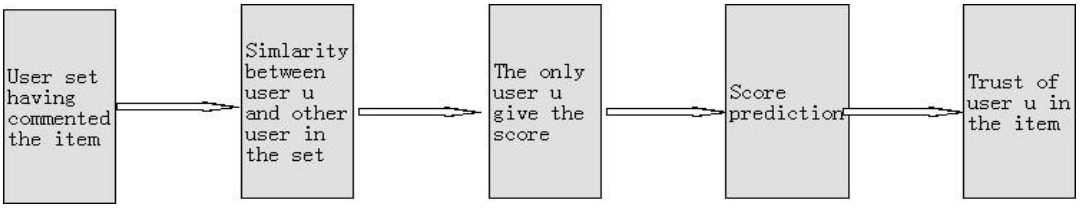

Figure 1. user level trust

The accuracy of for user μ to predict user ν in relative to item i is

$$T^i(v,u) = 1 - \frac{|p_{vi} - r_{vi}|}{s} \tag{2}$$

in which *s* represents the largest value of the score which user can give and r_{vi} represents the real score which user ν give to item i. So consider all the users which have grant to item i, the trust for user μ to item i is

$$Trust^i(u) = \frac{\sum_{v \in I_i} T^i(v,u)}{|I_i|} \tag{3}$$

In which I_i is the set of all the users which have given scores to item i.

Expert User. Social psychology research [12] shows that a person's identity in a field will affect the trust of his or her recommendation evidently when he or she recommends items. When people want to know a field, they often ask experts in this field. So it's necessary to bring expert users in recommend system. In this paper we use the user trust model in [13, 14] to determine whether a user is an expert user by user activity, score expertise and degree of trust by user.

$$Expert(u) = \alpha Active(u) + \beta AvgDev(u) + \gamma \operatorname{Re} putation(u) \tag{4}$$

In Eq.4 we have $\alpha + \beta + \gamma = 1$ in which α, β and γ are the weights of user activity, score expertise and degree of trust by user.

Collaborative Filtering Algorithm Based on Double Clustering and User Trust

Item Clustering. There are two methods to perform item clustering. One uses the users' score to the item to do clustering. The other is uses the attributes of the item to do it. With a view of the sparsity of item score matrix, the first method is unsuitable. On the other hand the data set MovieLens which we use gives attributes for every item. So we take the item attributes to perform clustering.

First we build the attribute vector for item i. Then we perform clustering as follows.

Input: The set of item I; the attributes of the items.

Output: *k* item clusters.

Step 1: Choose *k* items from item set $I = \{i_1, i_2, \cdots, i_n\}$ as the centers of clusters. The set of the centers is denoted as $Ccenter = \{cc_1, cc_2, \cdots, cc_k\}$.

Step 2: Initialize *k* empty clusters $C_1, C_2, \cdots, C_k$. Denote the set of the *k* clusters as $C = \{C_1, C_2, \cdots, C_k\}$.

Step 3: Repeat

Step 4: For $i \in I$

Step 5: For $cc_j \in Ccenter$

Step 6: Calculate the distance between the attribute vector of item *i* and the cluster center vector cc_j by cosine similarty.

Step 7: End For

Step 8: Choose the nearest cluster center cc_m and add item i to the cluster C_m to which cc_m belongs.

Step 9: End For

Step 10: For $C_i \subseteq C$

Step 11: Update the center cc_i of cluster *i*.

Step 12: Calculate the sum of square of the errors between all items in the cluster and the cluster center as follows.

$$\mathrm{E} = \sum_{i=1}^{k} \sum_{j \in C_i} sim(j, cc_i)^2$$

Step 13: Until all the centers of the clusters no longer vary, or in other words, the sum of square of the errors converges.

Similarity Mixture. The similar items converge into a cluster after item clustering. We can determine a user's degree of interest to a cluster by the proportion for which the sum of his score to all items in the cluster accounts of his score to all the items. So we denote the degree of interest of one user μ to item one cluster C_i as

$$hobby_{u,i} = \frac{\sum_{l \in C_i} r_{ui}}{\sum_{j \in C} r_{uj}} \tag{5}$$

Next we can get the cluster similarity between μ and user ν by similarity correlation formula

$$sim_c(u, v) = \frac{\sum_{i \in C_{uv}} \left(hobby_{u,i} - \overline{hobby}_u\right) \cdot \left(hobby_{v,i} - \overline{hobby}_v\right)}{\sqrt{\sum_{l \in C_{uv}} \left(hobby_{u,l} - \overline{hobby}_u\right)^2} \sqrt{\sum_{l \in C_{uv}} \left(hobby_{v,l} - \overline{hobby}_v\right)^2}} \tag{6}$$

In this paper we take the mixture of degree of interest and user trust to measure the similarity between users. It can be obtained by the following equation.

$$sim(u, v) = \alpha \cdot sim_c(u, v) + \beta \cdot Trust(u, v) \tag{7}$$

In which $\alpha + \beta = 1$. $sim_c(u, v)$ represents the item cluster similarity between user μ and user ν. $Trust(u, v)$ represents the degree of trust for user μ to user ν.

User Clustering. The similarity in Eq.7 is also the basis to measure the distance between users in user clustering. We take the classical K-mean clustering algorithm to perform user clustering. Since the classical K-mean clustering algorithm is sensitive to outer points, how to choose the cluster center

is very important. In this paper the expert user are taken as cluster centers which aren't outer points from both score and social contact. So it may help us to get relatively good results. The procedure of user clustering is similar to that of item cluster except using mixed similarity to compute the distance. So we needn't introduce it again.

The Nearest-Neighbor Search. In our collaborative filtering algorithm based on double clustering and user trust, we search the nearest neighbors of a user from the cluster in which he or she is. We use Eq.7 to find the mixed similarity between the target user μ and the other user μ in the same cluster. Then we choose N users with the maximum mixed similarity to form the nearest-neighbor set Nei_u of the target user μ .

Score Prediction. Now we can predict the scores which user μ gives to an item after we get Nei_u of it. We use the user level trust in Eq.7 to predict. So we get

$$P_{u,i} = \bar{r}_u + \cdot \frac{\sum_{v \in Nei_u} Trust^i(v) \cdot (r_{v,i} - \bar{r}_v)}{\sum_{v \in Nei_u} |Trust^i(v)|} \tag{8}$$

Experiment Results and Analysis

The Dataset and Evaluation Metrics. We take Movie Lens dataset with 1M size which is provided by Group Lens laboratory. But there are no datasets with both friendship relation and degree of trust between users. So we produce friendship relation and degree of trust at random by the density of friend of the users in Opinions dataset.

The mean absolute error (MAE) is applied in evaluating the quality of a recommender system. It's the metric of the difference between prediction scores and actual scores. It's obvious that the smaller MAE is, the better performance does the recommender system gain.

Results Analysis. First we discuss how many expert users we should choose in the recommender system.

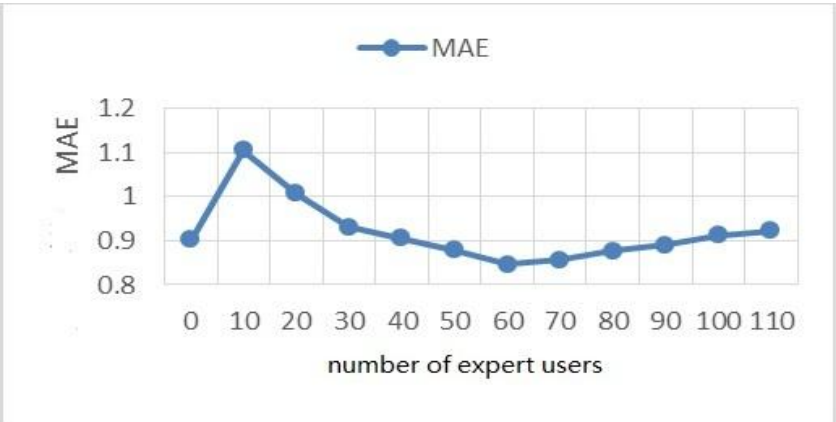

Figure 2. MAE---- number of expert users

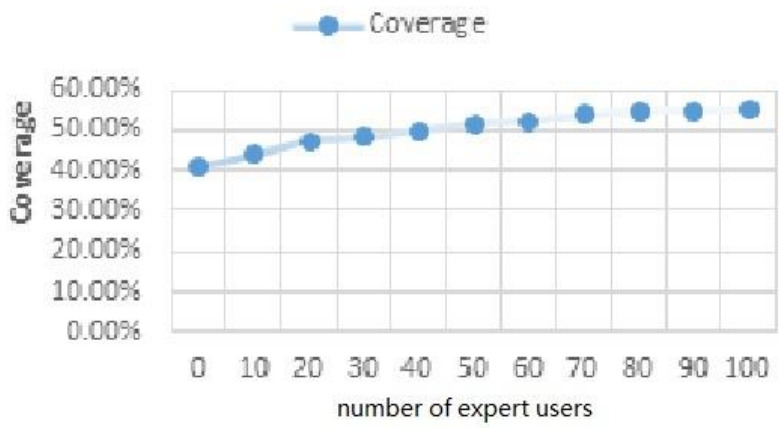

Figure 3. Coverage---- number of expert users

From Fig.2 we can find that MAE decreases with when the number of expert users increases. But MAE increases after the number of expert users reaches 60. From Fig.3 we can find that the coverage increases rapidly when the number of expert users increases. But the coverage no longer obviously increases when the number of expert users reaches 70. The reason is that the popularity of an item obeys the long tail effect. The proportion of items which the expert users comments in all items is still low. When the number of expert user reaches a certain quantity, their interest and trust have reaches high coverage. So it's of no use to increase the number of expert users any more. Consider all the result from Fig.2 and Fig.3, it's the best choice to let the number of expert users is 60.

Second we need find the best parameter α in the mixed similarity of the double clustering algorithm by.

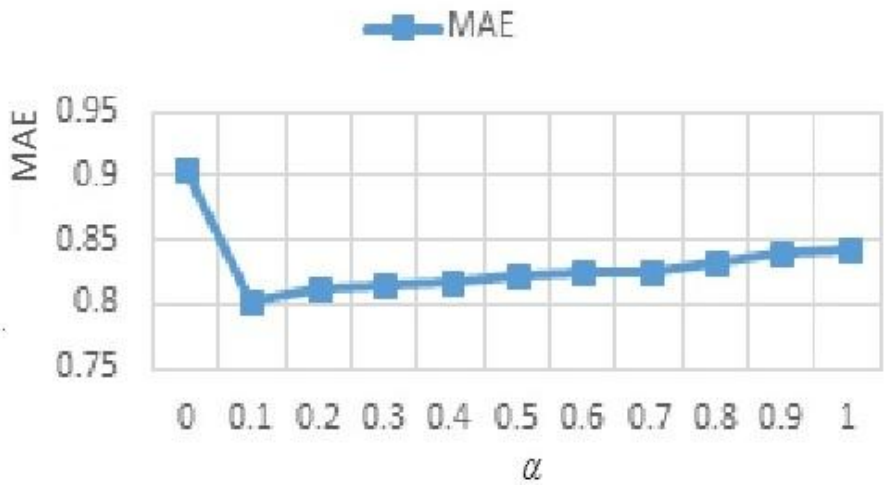

Figure.4 MAE---- α

FromFig.4 we can find that MAE has the smallest value when $\alpha = 0.1$. So we take this value in our algorithm.

Finally we perform experiments using traditional collaborative filtering algorithm based on user (CFU), collaborative filtering algorithm based on user and clustering (CFUC) and our collaborative filtering algorithm based on double clustering and user trust (CFDCUT) respectively. So we get

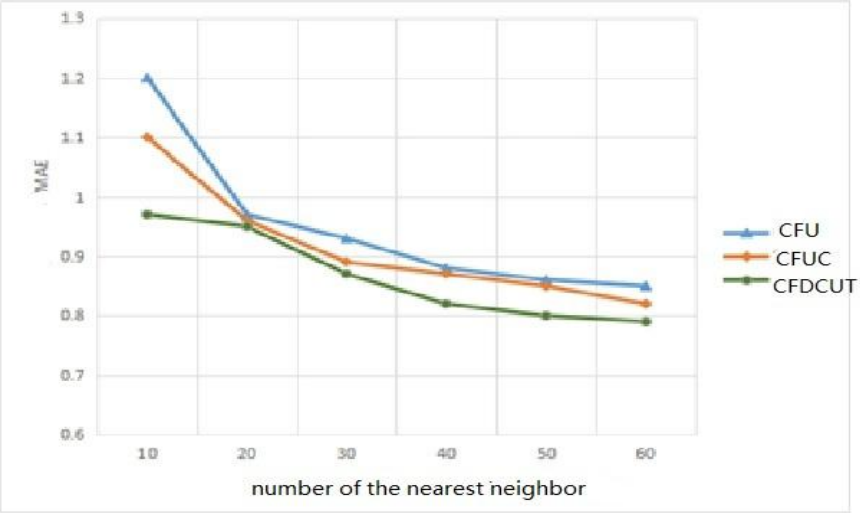

Figure.5 MAE----number of the nearest neighbor

From Fig.5 we can find that MAE in all the three algorithms decrease when the number of the nearest neighbors increase, or in other words, the accuracy in every algorithm increases. But MAE in CFDCUT is always the smallest. So CFDCUT can improve the accuracy of score prediction.

Conclusion

In this paper we present a collaborative filtering algorithm based on double clustering and user trust. The experiments results show that this method could relieve the sparsity problem and improve the accuracy of the prediction results.

Acknowledgements

This work is supported by Natural Science Foundation of China (Grants 61472412), Natural Science Foundation of the Education Department of Henan Province of China (Grants 14A520012) and Natural Science Basic Research Plan in Shannxi Province of China (No. 2014JM2-6103).

References

[1] Velásquez, Juan D., and Vasile Palade. "Building a knowledge base for implementing a web-based computerized recommendation system." International Journal on Artificial Intelligence Tools 16.05 (2007): 793-828.

[2] Banerjee, Arindam. "A generalized maximum entropy approach to bregman co-clustering and matrix approximation." Journal of Machine Learning Research 8.12(2007):509--514.

[3] George, Thomas, and S. Merugu. "A scalable collaborative filtering framework based on co-clustering." IEEE International Conference on Data Mining IEEE, 2010:625-628.

[4] DENG Ai-lin, ZUO Zi-ye1, and ZHU Yang-yong，"Collaborative Filtering Recommendation Algorithm Based on Item Clustering." Journal of Chinese Computer Systems 25.9 (2004): 1665-1670.

[5] Sinha, Rashmi R., and Kirsten Swearingen. "Comparing Recommendations Made by Online Systems and Friends." DELOS workshop: personalisation and recommender systems in digital libraries. Vol. 1. 2001.

[6] Matsuo, Yutaka, and Hikaru Yamamoto. "Community gravity: measuring bidirectional effects by trust and rating on online social networks." Proceedings of the 18th international conference on World wide web. ACM, 2009.

[7] Victor, Patricia, et al. "Key figure impact in trust-enhanced recommender systems." AI Communications 21.2-3 (2008): 127-143.

[8] Ma, Nan, et al. "Trust relationship prediction using online product review data." Proceedings of the 1st ACM international workshop on Complex networks meet information & knowledge management. ACM, 2009.

[9] Jamali, Mohsen, and Martin Ester. "A matrix factorization technique with trust propagation for recommendation in social networks." Proceedings of the fourth ACM conference on Recommender systems. ACM, 2010.

[10] Maurer, Ueli. "Modelling a public-key infrastructure." Computer Security—ESORICS 96. Springer Berlin Heidelberg, 1996.

[11] Leskovec, Jure, and Eric Horvitz. "Planetary-scale views on a large instant-messaging network." Proceedings of the 17th international conference on World Wide Web. ACM, 2008.

[12] Liu, Guanfeng, Yan Wang, and Mehmet Orgun. "Trust inference in complex trust-oriented social networks." Computational Science and Engineering, 2009. CSE'09. International Conference on. Vol. 4. IEEE, 2009.

[13] Li Lu, Research of Credible User Recommendation Model In Network Community Based On Interest Relationship. MS thesis. Jiangxi University of Finance and Economics, 2014.

[14] Palla, Gergely, et al. "Uncovering the overlapping community structure of complex networks in nature and society." Nature 435.7043 (2005): 814-818.