An Improved Collaborative Filtering Algorithm Based on User Interest

Li Zhang

School of Information Technology & Management Engineering,University of International Business and Economics, Beijing,China

Email:zhangli_amy@hotmail.com

Tao Qin PiQiang Teng

School of Information Technology & Management Engineering,University of International Business and Economics, Beijing,China

Email: {tasummer@sina.com, taspring@sina.com}

***Abstract*—With the development of personalized services, collaborative filtering techniques have been successfully applied to the network recommendation system. But sparse data seriously affect the performance of collaborative filtering algorithms. To alleviate the impact of data sparseness, using user interest information, an improved user-based clustering Collaborative Filtering (CF) algorithm is proposed in this paper, which improves the algorithm by two ways: user similarity calculating method and user-item rating matrix extended. The experimental results show that the algorithm could describe the user similarity more accurately and alleviate the impact of data sparseness in collaborative filtering algorithm. Also the results show that it can improve the accuracy of the collaborative recommendation algorithm.**

***Index Terms*—collaborative filtering, data sparsity, user similarity, user interest**

I. INTRODUCTION

With the development of web2.0 technology and network services, recommendation technologies are applied to various network platforms to suppose personalized services for customers. Almost all large E-commerce platforms (e.g., *Amazon, CDNOW, eBay, and Taobao*) use recommendation systems based on various methods, among which collaborative filtering is the most successful technique in recommendation systems, without need for exogenous information about either items or users. The fundamental assumption of *CF* is that if users X and Y rate n items similarly, or have similar behaviors (e.g., buying, watching, listening), and hence will rate or act on other items similarly. CF algorithms are required to have the ability to deal with highly sparse data, to scale with the increasing numbers of users and items, to make satisfactory recommendations in a short time period, and to deal with other problems like synonymy (the tendency of the same or similar items to have different names), shilling attacks, data noise, and privacy protection problems [1], which leads to the recommendation accuracy is far behind the expectation of consumers and businesses. To solve those problems, a number of CF algorithms are put forward, such as item-based, user-based, model-based, content-based CF and so on [2]. Those studies alleviate the impact of the above problems and improve the accuracy and scalability of collaborative filter algorithm in some extent. However, as the number of online users and product items grows rapidly, data sparsity still greatly challenge the performance of CF techniques. In this paper, we mainly focus on data sparseness problem in CF algorithm. Based on the existing research, using user interesting information, an improved user-based clustering collaborative filtering algorithm is proposed to alleviate the impact of data sparseness. And the performance of the algorithm is tested by experiments.

So the rest of this paper is organized as follows: related work is introduced in section II and traditional user-based clustering *CF* method is described in section III, while the improved algorithm is described in section IV. Section V is experimental analysis. At last in selection VI, conclusions are drawn.

II. HELPFUL HINTS

To alleviate the impact of data sparsity in collaborative recommendation system, researchers have put forward a number of methods. Existing research on solving data sparseness problem can be largely divided into three branches. One stream of research is Model-based CF and hybrid recommender algorithms. Various clustering techniques, *SVD* model, sparse factor analysis, Bayesian belief nets CF, matrix reduction CF, data mining models and composite collaborative filtering algorithms are

Corresponding author: Li Zhang (zhangli_amy@hotmail.com)

doi:10.4304/jsw.9.4.999-1006

applied in CF algorithms [3-6]. Although those methods may increase complexity and expense for implementation, even lose useful information as dimensionality reduction techniques. These methods still attract attention of scholars, as they can better address the sparseness, scalability or other problems, and improve prediction performance. Another stream of research is improved user similarity calculated methods in memory-based CF algorithms, including user-based CF and item-based CF algorithms, in which the most common approach is through neighborhood-based interpolation. Sarwar B. and *et al*. (2001) proposed an item-based CF algorithm, which improve the insufficient of traditional similarity measure method when user rating data is extreme sparse [7]. While LI Cong and *et al*. (2008) improved the above algorithm and reduced the effects of data sparsity, which only predicted rating for the users who had ability to recommend [8]. Ma and *et al* (2007) proposed a CF algorithm based on the nearest neighbor set of target users and items to generate recommendation results, with adjustment parameter to control the weight of the two parts. But it needs to set the parameter value by experience [9]. Nathan and *et al*. (2010) increased time factor in user similarity calculating process and achieved relatively good recommendation results [10]. Similar works have been down by PaPagelis M.*et al* (2005), Tieli Sun,*et al* (2009) and Sun H,*et al*.(2011), who utilized item similarity to improve user similarity calculating process and achieved better results [11-13]. The other stream of research is by analyzing the characteristics of users and items comprehensively to enhance the performance of CF algorithm. Some researchers enhance the performance of CF by using extra information of users or items, such as user location, user activity and user interest [14]. Yehuda Koren (2009) proposed a collaborative filtering algorithm based on dynamic reordering within the nearest neighbor set, which dynamically adjusted the weight of users in neighbor set according to different target items based on user activity [15]. To some extent, their methods reflect the user's interest and improve the accuracy and online real-time response speed of the recommendation system. While Liu Q and *et al.* (2012) expand user interest based on the relationship between items to calculate user similarity [16]. Tsang-Hsiang Cheng and *et al.* (2011) also integrated user interest with time into user similarity calculated [17]. Jia D and *et al.* (2013) measured the user's trustworthiness and combined the computational model of trust with traditional collaborative filtering approach to generate recommendation for the target user [18]. Those studies alleviate the impact of data sparseness and improve the accuracy and scalability of collaborative filter algorithm in some extent. However there are very few researches combine them together in the same algorithm. On the other hand, some researchers extend the use-item rating matrix using user' attributes and item content information for calculating user similarity to solve cold–start problems in CF algorithm [19-20]. In fact, these methods mainly improve the calculation method of similarity in the algorithm. For example, H. J. Ahn (2008) combined item content information and popularity with user-behavior data to calculate similarity between users and get good results[21].

Thus, on the basis of these studies, inspired by the idea of heterogeneous information network clustering method in reference [22], combining user interest information, an improved user-based clustering collaborative filtering algorithm is proposed in this paper, which is improved through two ways: improving user similarity calculating method and extending user-item rating matrix. The method considers the characteristics of user interest and items at same time. So it can reflect the similarity between users more truly.

III. USER-BASED CLUSTERING CF METHOD

In this paper, we adopt K-means clustering method in user-based CF algorithm. First, analyze user-item rating data and divided into K clusters. Then allocate the target user to the most similar cluster and generate its nearest neighbor set. At last, recommend the *top-M* items most interested by the nearest neighbors to target users based on their predicted rating for the items.

A. UserIitem Rating Matrix

CF techniques use a database of preferences for items by users to predict additional topics or products, which a new user might like. In a typical *CF* scenario, there is a list of m users $\{u_1, u_2, \ldots, u_m\}$ and a list of n items $\{I_1, I_2, \ldots, I_n\}$. Each user has rated a list of items or has preferences inferred through their behaviors. The ratings can either be explicit indications, and so forth, on a 1-5 scale, or implicit indications, such as purchases or click-throughs [1]. For example, Table I shows a list of people and the items they like or dislike, which can be denoted as user-item rating matrix. There are missing values in the matrix where users did not give their preferences for certain items.

TABLE I.
USER-ITEM RATING MATRIX

Item ID / User ID	I1	I2	I3	I4	I5	I6
U1	3		5			4
U2		3	4			4
U3				4	4	
U4		3		5	4	

B. User Similarity

Similarity computation between users is a critical step in used-based collaborative filtering algorithms. There are many different methods to compute similarity or weight between users. *Pearson coefficient* measures the extent to which two variables linearly relate with each other. Here, we also use *Pearson coefficient* to measure the degree of similarity between users [1]. Suppose $I(u)$ and $I(v)$ are the item set rated by user u and v respectively, and $I(u) \cap I(v)$ means the common item set rated by user u

and v . Then the similarity between user u and v ($sim(u,v)$) can be indicated by (1):

$$sim(u,v)=\frac{\sum_{i\in I(u)\cap I(v)}(R_{u,i}-\overline{R_u})(R_{v,i}-\overline{R_v})}{\sqrt{\sum_{i\in I(u)\cap I(v)}(R_{u,i}-\overline{R_u})^2}\sqrt{\sum_{i\in I(u)\cap I(v)}(R_{v,i}-\overline{R_v})^2}} \quad (1)$$

Where $R_{u,i}$ and $R_{v,i}$ are the rating of user u and v for item $i\ (i\in I(u)\cap I(v))$, and $\overline{R_u}$ and $\overline{R_v}$ are their respective average rating. We normalize the data in user rating matrix into [0, 1] in the algorithm.

C. Predicted Rating

To obtain predictions or recommendations is the most important step in a collaborative filtering system. In the user-based *CF* algorithm, a subset of nearest neighbors of the target user are chosen based on their similarity with him or her, and a weighted aggregate of their ratings is used to generate predictions for the target user. In the paper, we use the weighted average method to predict the rating of a target user[2], shown as (2):

$$p_{a,t}=\overline{R_a}+\frac{\sum_{i\in U_a} sim(a,i)*(R_{i,t}-\overline{R_i})}{\sum_{i\in U_a} sim(a,i)} \quad (2)$$

Where U_a is the nearest neighbor set of target user a and $p_{a,t}$ is the predicted rating of user a for item $t(t\in I(U_a)$, *which is the common item set rated by users in* U_a)

D. User-based Clustering Collaborative Filtering (UCCF) Algorithm

The steps of UCCF algorithm are described as follows:

Input：User-Item rating data

Output：Top-M items

Step1: Randomly select K users as the initial cluster center;

Step2: Allocate the remaining users to the K clusters and update the clustering centers;

Step3: Repeat Step 2,until clustering centers are no longer changes;

Step4: Allocate target user a to the most similar cluster and select KN users to generate its nearest neighbor set U_a by (1);

Step5: Generate the common item set $I(U_a)$ rated by users in U_a ;

Step6: Using (2) calculate the $p_{a,t}\ (t\in I(U_a))$, and sort them in descending order. Then recommend the Top-M items to the target user.

IV. THE IMPROVED USER-BASED CLUSTERING CF METHOD

Equation (1) requires the common item set evaluated by the two users. But with the increasing of users and items, it maybe occur extreme sparse user-item rating data. And the number of item in the common set evaluated by two users may be very small or zero. In this case, to calculate user similarity by (1) will not get effective results. Accordingly, we calculate the user similarity by user activity based on user interest categories to alleviate the impact of data sparseness in CF algorithm.

A. User Activity Based on Item Categories of User Interest

In this paper, the activity of user u is defined as (3):

$$acti_u = N_u/N \quad (3)$$

Where N_u respresents the number of items rated by user u and N is the total number of items.

Inspired by the reference[15] and [22], we calculate user activity based on item categories interseted by the user, which can be obtained by the domain knowledge or item-clustering analysis. That is to say, divide items into different categories and then calculate user activity based on her/his interesting categories. Assume C_I is the number of items categories, $G_{u,i}$ and $G_{v,i}$ are the number of items in category $i(i=1,2,\dots,C_I)$ rated by user u and v respectively, then user activity based on user interesting categories can be identified by $G_{u,i}/C_I$ and the difference between user u and v can be calculated by (4):

$$dif_{(u,v)}=\frac{\sum_{i=1}^{C_I}|G_{u,i}-G_{v,i}|}{\sum_{i=1}^{C_I}(G_{u,i}+G_{v,i})} \quad (4)$$

In the traditional user-based CF algorithm, Active users easier access to the nearest neighbours set used to generate recommended sequence[23]. To avoid this, we improved the user similarity calculated method by the difference of user activity. According to this point, the similarity between user u and v is calculated by (5):

$$sim'_{(u,v)}=\alpha\ sim(u,v)+(1-\alpha)(1-dif_{(u,v)}) \quad (5)$$

Where $sim(u,v)$ shown as (1) and $\alpha\in[0,1]$ is the parameter to adjust the weight of user activity.

B. User-Item-User Activity Matrix Expanded by User Interesting

We extend user-item rating matrix based on user activity, increasing C_I (C_I is the number of item categories) columns in the matrix and data in each column are equal to the value of $G_{u,i}/C_I$. For example, assume that item set I = *{I1, I2, I3, I4, I5, I6}* can be divided to two categories:*Class1, Class2*, and *I1, I2, I3* belong to *Class1* while *I4, I5, I6* belong to *Class2*. So the user rating matrix in TABLE I can be extened as TABLE II, identified by user-item-user activity matrix.

TABLE II
USER-ITEM-USER ACTIVITY MATRIX

	I1	I2	I3	I4	I5	I6	User Activity in Class1	User activity in Class2
U1	3		5			4	2	1
U2		3	4			4	2	1
U3	4			4	4		1	2
U4		3		5	4		1	2

C. *User Interesting-based Clustering Collaborative Filtering (UICCF) algorithm*

The steps of *UICCF* algorithm are similar as *UCCF* algorithm in section III. But there are two differences. One is the input data. The user-item-user activity matrix is used in of *UICCF* algorithm,while user-item rating matrix used in *UCCF* algorithm The other is user similarity calculated method. *UICCF* algorithm uses (5) to calculate user similarity,while *UCCF* algorithm uses (1) .

V. EXPERIMENTS

A. *Description of the Data Set*

The MovieLens dataset is used in in our experiments, which is the most popular dataset used by scholares and developers in the field of collaborative filtering research. The dataset contains real data corresponding to the movie ratings captured on the website of the MovieLens movie recommender (*http://movielens.umn.edu*)during a 7-month period (19-09-1997 to 22-04-1998). From these data, users with less than 20 ratings have been removed, giving a total of 100,000 ratings from 943 users on 1,682 movies. The dataset is divided into 5 training subsets *(u1.base-u5.base)* and 5 testing subsets *(u1.test-u5.test).* The distribution of each datasets is shown in TABLE III. All movies belong to 19 classes, such as *Children's,Comedy,Crime* and *et al*. In this paper, all data were standardized into [0, 1].

TABLE III
DISTRIBUTION OF TRAINING AND TESTING DATASET

Data Subset (training set /testing set)	Records	Users	Movies	Data density
u1.base /u1.test	80,000 /20,000	943/459	1650/1410	0.0514 /0.0309
u2.base /u2.test	80,000 /20,000	943/653	1648/1420	0.0515 /0.0216
u3.base /u3.test	80,000 /20,000	943/869	1650/1423	0.0514 /0.0162
u4.base /u4.test	80,000 /20,000	943/923	1660/1394	0.0511 /0.0155
u5.base /u5.test	80,000 /20,000	943/927	1650/1407	0.0514 /0.0153

B. *Evaluation Metrics*

The quality of a recommender system can be decided on the result of evaluation. The type of metrics used depends on the type of *CF* applications. In this paper, we use Mean Absolute Error (MAE) as the measure for performance evaluation. The MAE measures the difference, as absolute value, between the prediction of the algorithm and the real rating. Despite its limitations when evaluating systems focused on recommending a certain number of items, the simplicity of its calculation and its statistical properties have made it become one of the most popular metrics when evaluating recommender systems. It is computed over all the ratings available in the evaluation subset, using (6):

$$\mathrm{MAE} = \frac{\sum_{t=1}^{N_i} |p_{i,t} - R_{i,t}|}{N_i} \tag{6}$$

Where $N_i(N_i \leq N, total\ number\ of\ items)$ is the number of rated items by user i, $p_{i,t}(t = 1,2 \dots N_i)$ is the predicted rating of user i and $R_{i,t}(t = 1,2, \dots, N_i)$ is his/her real rating. So the smaller value of MAE means the higher quality of recommended.

C. *Experiment Results*

To test the recommendation accuracy of the algorithm proposed in this papaer, the experiment is divided into two steps:

First, explore the value of α in (5). Fig.1 shows that the average MAE of the five testing data sets varies with α while $KN = 30{,}60$ (KN respresent the number of users in the nearest neibors set). The detailed experimental results for each tesing dataset can be seen from Fig.3-Fig.4 in Appendix A.

Fig.1 shows us that the upward trend of MAE is obvious as α increase. That means user activity based on the user interest can affect the recommendation accuracy of the CF algorithm. Also from Fig.1,we can see the value of MAE changes more obviously when $\alpha > 0.5$. So in the second step,we set $\alpha = 0.5$.

Second, compare the performance of the algorithm proposed in this paper with UCCF algorithm. In this step, we test the proposed method through three experiments and Fig.2 shows that the average MAE of the five testing data sets of each experiments.

The first experiment uses UCCF algorithm to recommend based on user-item rating matrix, while calculates user similarity by (1). The result is shown in Fig.2 denoted by UCCF1.

The second experiment also uses UCCF algorithm to recommend based on user-item rating matrix,but calculates user similarity by (5). The result is shown in Fig.2 denoted by UCCF5.

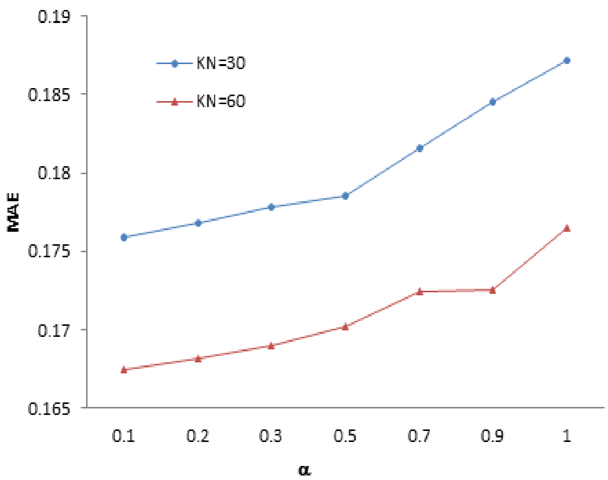

Figure 1 The Average MAE Change with α ($KN = 30{,}60$)

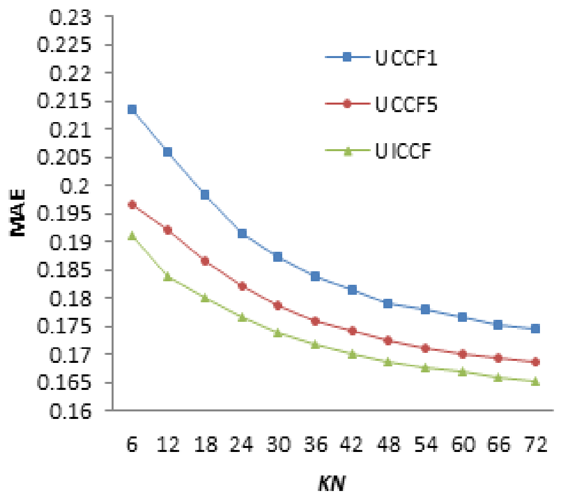

Figure 2 The Average MAE Comparison of the Algorithms

Those two experiments mainly verify the effect of different user similarity calculated method in CF algorithm.

The third experiment uses UICCF algorithm based on user-item-user activity matrix and calculates user similarity by (5), whose result denoted by UICCF in Fig.2.

In the experiments, set user-based clustering number $K = 12$, the number of nearest neighbor $KN = 6,12,18,24,\ldots 72$ and $\alpha = 0.5$ in (5). Also based on the domain knowledge acquired, divide all movies into 19 categories, i.e. $C_I = 19$. The experimental results are shown in Fig.2, which shows the average MAE of 5 testing data subsets in TABLE III. Also the detailed experimental results are shown as Fig.5-Fig.9 for each testing dataset in Appendix A).

From Fig.2, we can know user similarity calculated method (Eq.(5)) proposed in the paper improves the accuracy of CF algorithm significantly and extending user-item rating matrix based on user interesting further improves the quality of CF algorithm.

As the main process of the CF algorithm is to find out the most similar neighbors to collaborate with the target user. The experimental results show that the algorithm could describe the user similarity more accurately. Although, analyzing items rated by a user will increase the time complexity of the algorithm. We can perform the statistics operation offline, so it will not affect the real-time of CF algorithm. On the other hand, in our algorithm we do not think about user interesting shift with time, which has been veritied can improved the accuracy of CF algorithm in reference [24]. So the future research we will combine this in our method.

VI. Conclusions

In order to reduce the impact of data sparseness in personalized recommendation system, an improved user-based clustering collaborative filtering algorithm is proposed, which improves the algorithm by two ways: user similarity calculating method and user-item rating matrix extended. In user similarity calculated process, user activity is calculated by user interesting information. The experimental results show that it could significantly improve the accuracy of the collaborative recommendation algorithm. Although the method in this paper could alleviate the impact of data sparseness, it did not consider the problem of cold-start and user interest shifted with time. Further research will be focused on the user interest shifted and cold-start problems in collaborative filtering algorithms.

The main process of the user-based CF algorithm is to find out who can collaborate with the target user. All existing studies try to find the nearest neighbors of the target user, without differentiating their recommendation ability. Because of the restrictions placed on the similarity (e.g., by picking the closest neighbors to the target user), some neighbors with better recommendation ability may not be selected. Especially, some key users who play an essential role in the diffusion of information are not considered in the recommendation algorithm [25]. On the other hand, with the development of social network, some researchers improve CF algorithm by integrate the attributes of social network into rating data, such as friendship, membership or tag data of social network. So another future work is to select neighbors by the structural characteristics of the social network and the information of key users in a social network.

Appendix A. Detailed EXPEROMENT Results

In this section, the detailed experiment results are given.

Fig 3 and Fig.4 show the MAE change with α for each testing dataset (u1.test-u5.test), while $KN = 30$ and $KN = 60$.

Fig 5-9 shows the MAE comparison of the algorithms for each testing dataset (u1.test-u5.test).

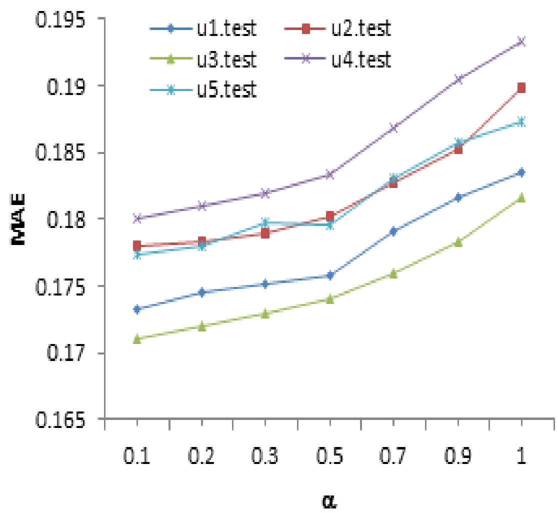

Figure 3 MAE Change with α $(KN = 30)$ for Each Testing Dataset

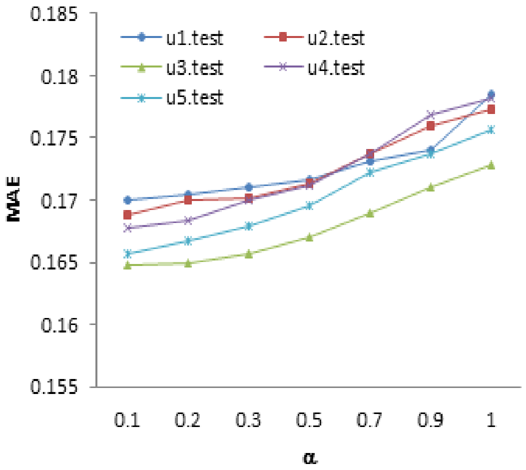

Figure 4 MAE Change with α $(KN = 60)$ for Each Testing Dataset

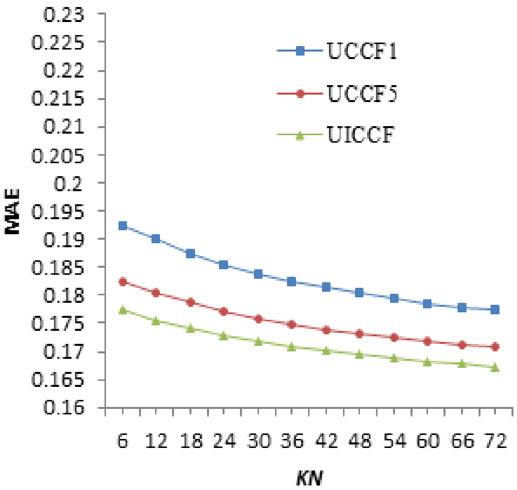

Figure 5 MAE Comparison of the Algorithms for u1.test

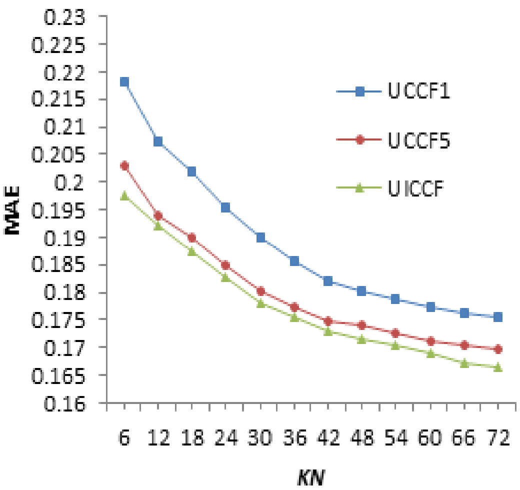

Figure 6 MAE Comparison of the Algorithms for u2.test

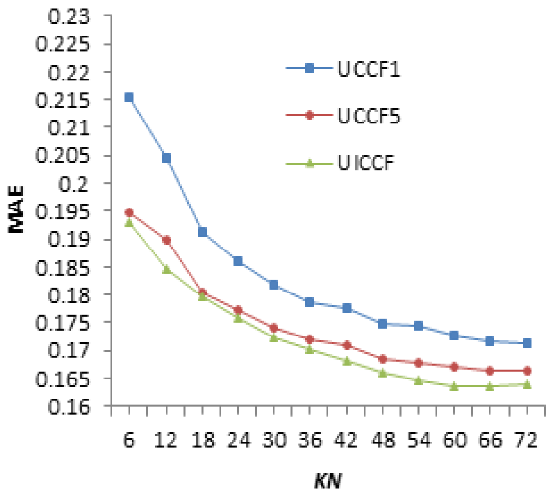

Figure 7 MAE Comparison of the Algorithms for u3.test

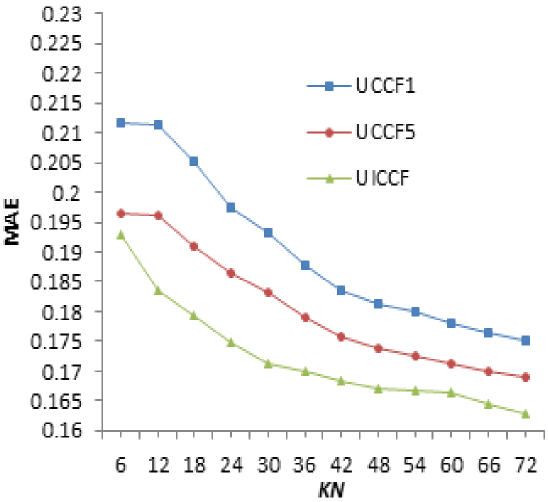

Figure 8 MAE Comparison of the Algorithms for u4.test

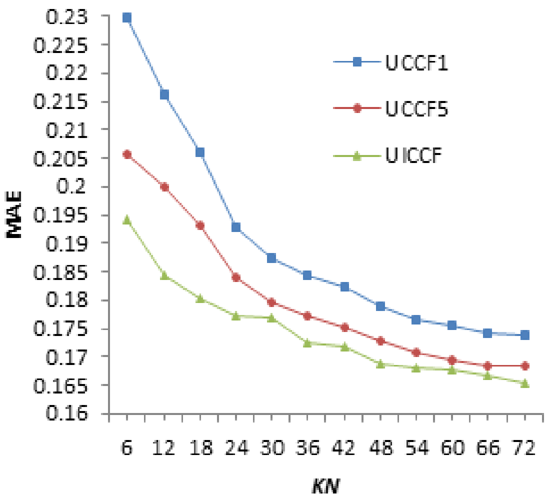

Figure 9 MAE Comparison of the Algorithms for u5.test

ACKNOWLEDGMENT

This work was supported by National Social Science Foundation of China (Grant No. 13BTQ027), Beijing Philosophy and Social Science Planning Foundation (Grant No. 12JGB034) and University of International Business and Economics Research Fund(No. X10042). We are grateful to the editors and the reviewers whose comments have improved this paper considerably.

REFERENCES

[1] Xiaoyuan Su&Taghi M. K. A Survey of Collaborative Filtering Techniques[J]. Advances in Artificial Intelligence, Vol.2009(1),pp.1-19,2009.

[2] FIDEL C.,V´ICTOR C.,DIEGO F.& VREIXO O. Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposals for Scalable, High-Performance Recommender Systems[J]. ACM Transactions on the Web, Vol. 5(1),pp.2-33,2011.

[3] Truong, K., Ishikawa, F., and Honiden, S. Improving accuracy of recommender system by item clustering. IEICE - Trans. Inf. Syst., E90-D(9),pp.1363－1373,2007.

[4] Dhoha A., Ghadeer S., Lamia A., Mona K., Romy M. A Survey Paper on Recommender Systems. www.arxiv.org/computer science/information retrival,2010.11

[5] G. Adomavicius & A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions[J]. IEEE Transactions On Knowledge And Data Engineering, 17(6),pp.734–749, 2005

[6] Hu L, Wang W, Wang F, et al. The Design and Implementation of Composite Collaborative Filtering Algorithm for Personalized Recommendation[J]. Journal of Software, 7(9),pp.2040-2045,2012.

[7] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C].Proceedings of the 10th international conference on World Wide Web. ACM, pp.285-295,2001.

[8] Li Cong, Liang Changyong, M a Li. A collaborative filtering recommendation algorithm based on domain nearest neighbor[J]. Journal of Computer Research and Development, 45(9),pp.1532 – 1538,2008(in Chinese).

[9] Ma Hao, King Irwin, Lyu Michael. Effective missing data prediction for collaborative filtering[C].Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Amsterdam, Netherlands, pp.39-46,2007.

[10] Nathan N. Liu, Min Zhao, Evan Xiang, and Qiang Yang. Online evolutionary collaborative filtering. Fourth ACM Conf. on Recommender Systems, pp.95-102, 2010.

[11] PaPagelis M,Plexousakis D,Rousidis I, et al. Qualitative analysis of user-based and item-based Prediction algorithm for recommendation agents[J]. Engineering Application of artificial intelligence,18(7),pp781-789,2005.

[12] Tieli Sun, Lijun Wang, Qinghe Guoin. A Collaborative Filtering Recommendation Algorithm Based on Item Similarity of User Preference[C]. The 2th International Workshop on Knowledge Discovery and Data Mining, Volume 1,pp.60-63,2009.

[13] Sun H, Peng Y, Chen J, et al. A new similarity measure based on adjusted euclidean distance for memory-based collaborative filtering[J]. Journal of software, 6(6),pp. 993-1000,2011.

[14] Akihiro Y., Hidenori K.&Keiji Suzuki. Adaptive Fusion Method for User-Based and Item-Based Collaborative Filtering[J]. Advances in Complex Systems, 14(2),pp.133-149,2011.

[15] Yehuda Koren. Collaborative filtering with temporal dynamics. 15th ACM SIGKDD'2009,pp. 447-456,2009.

[16] Liu, Q., Chen, E., Xiong, H., Ding, C. H., & Chen, J. Enhancing collaborative filtering by user interest expansion via personalized ranking. Systems, Man, and Cybernetics[J].Part B: Cybernetics, IEEE Transactions on,42(1), pp.218-233,2012.

[17] Tsang-Hsiang Cheng, Hung-Chen Chen, Wen-Ben Lin& Yen-Hsien Lee.Collaborative Filtering With User Interest Evolution. Pacific Asia Conference on Information Systems, Paper 45,2011.

[18] Jia D, Zhang F, Liu S. A robust collaborative filtering recommendation algorithm based on multidimensional trust model[J]. Journal of Software, 8(1),pp.11-18,2013.

[19] Iaquinta L, Semeraro G. Lightweight approach to the cold start problem in the video lecture recommendation[C]. Proceedings of the ECML/PKDD Discovery Challenge Workshop, CEUR Workshop Proceedings. CEUR Workshop Proceedings.770,pp.83-94,2011.

[20] Qiu T, Chen G, Zhang Z K, et al. An item-oriented recommendation algorithm on cold-start problem[J]. EPL (Europhysics Letters), 95(5): 58003,2011.

[21] H. J. Ahn. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Sciences 178,pp.37-51,2008.

[22] Sun, J.Han, P.Zhao, Z.Yin, H.Cheng, and T.Wu. RankClus: Intergating Clustering with Ranking for Heterogeneous Information Network Analysis[C].In Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, pp.565-576,2009.

[23] Zeng W, Shang M S, Zhang Q M, et al. Can Dissimilar Users Contribute to Accuracy and Diversity of Personalized Recommendation?[J]. International Journal of Modern Physics C, 21(10),pp.1217-1227,2010.

[24] Bin Li, Xingquan Zhu, Ruijiang Li, Chengqi Zhang, Xiangyang Xue & Xindong Wu. Cross-Domain Collaborative Filtering over Time. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence,pp.2293-2298,2011.

[25] Bauer, F., & Lizier, J. T. Identifying influential spreaders and efficiently estimating infection numbers in epidemic models: A walk counting approach. EPL (Europhysics Letters), 99(6), 68007,2012.

Li Zhang was born in Qingdao city in 1972. She received the Ph.D. degree in signal and information processing from Beijing University of Post and Telecommunication, Beijing, China.

She is currently an associate professor of the School of Information Technology and Management Engineering, University of International Business and Economics. Her general research areas are business intelligence, data mining and recommendation systems. She also works on credit risk management and E-commerce. Her research is supported by National Social Science Foundation, Beijing Philosophy and Social Science Planning Foundation grants and University of International Business and Economics Research Fund.

She has published over 30 research papers in peer-reviewed journals and conference proceedings and two books about business intelligence.

Tao Qin received the B.E. degree in information management in 2012.She is currently working toward the Master degree from the School of Information Technology and Management Engineering, University of International Business and Economics. Her main research interests include E-commerce and recommendation systems.

Piqiang Teng received the B.E. degree in computer science in 2012.He is currently working toward the Master degree from the School of Information Technology and Management Engineering, University of International Business and Economics. His main research interests include intelligent data analysis and recommendation systems.