Journal of Information & Computational Science 9: 12 (2012) 3421–3429
Available at http://www.joics.com

Collaborative Filtering Algorithm Incorporated with Cluster-based Expert Selection [⋆]

Weiliang Kong[*], Qingtang Liu, Zhongkai Yang, Shuyun Han

National Engineering Research Center for E-learning, Central China Normal University
Wuhan 430079, China

Abstract

In order to solve the scalability and the noise problems suffered by collaborative filtering algorithm, the researchers have proposed expert-based collaborative filtering algorithm. But, there still lacks a principled model for guiding how to select the useful experts. In this paper, firstly, we define a concept of expert which can be reduced into two components: the activity and the influence in a given domain. Secondly, we put forward cluster-based expert selection method. Thirdly, we introduce this method into expert-based collaborative filtering algorithm and propose collaborative filtering algorithm incorporated with cluster-based expert selection. Finally, experiments show that our algorithm has better performance than the existing expert-based collaborative filtering algorithm on recommendation precision (about 12% improvement) and predication accuracy (about 1.8% improvement).

Keywords: Collaborative Filtering; Expert-based Collaborative Filtering; Similarity Measure; Clustering; Expert Selection

1 Introduction

With the development of network, the explosion of information makes people have to spend too much time and energy on searching for needed information, which is well known as "Information Overload" problem [1]. Collaborative filtering technology provides an efficient way to solve this problem and it is, so far, one of the most successful technologies [2, 3]. Collaborative filtering is a technique to filter or evaluate items through the opinions of other people [4]. The principle of collaborative filtering is finding the correlations between items or users according to the users' ratings and then recommending items which users may prefer to users according to the correlations. The Nearest Neighbor algorithm (KNN), for instance, does so by finding a number of similar users (or neighbors) whose profiles can be used to produce recommendation [7].

[⋆] The research work was supported by the key projects in the national science & technology pillar program of China (No. 2009BAH51B03) and the fundamental research funds for the central universities of China (No. 120002040477 and No. Weiliang KongCCNU10B01003).

[*] Corresponding author.
Email address: byta100@yahoo.cn (Weiliang Kong).

October 15, 2012

However, finding similar users is not an easy task. Computing the similarities between users is time-consuming which limits the scalability of recommendation system, and the accuracy of similarities is also limited by noise, introduced by users when giving their feedback to the recommendation system both in the form of careless ratings [5] and malicious entries [6]. Dimensionality reduction techniques such as SVD [8] and clustering algorithm [9] are proposed to deal with the scalability problem and to quickly produce good quality recommendation; Analyzing robustness [10], removing global effects [11] and other methods are proposed to solve the noise problem. But scalability and noise are still open research problems in collaborative filtering algorithm. Literature [7] introduced a radically different approach to collaborative filtering algorithm in which recommendations are drawn from a set of domain experts rather than from the general population. However, there still lacks a principled model for guiding how to select the useful experts. In this paper, firstly, we define a concept of expert which can be reduced into two components: the activity and the influence in a given domain. Secondly, we put forward cluster-based expert selection method. Thirdly, we introduce this method into expert-based collaborative filtering algorithm and propose collaborative filtering algorithm incorporated with Cluster-based Expert Selection (CBES).

The remainder of the paper is organized as follows: Section 2, we introduce relevant work of collaborative filtering algorithm and summarize the expert-based collaborative filtering algorithm; Section 3, we describe the redefinition of expert, put forward cluster-based expert selection method and a new similarity measure, and propose our collaborative filtering algorithm incorporated with cluster-based expert selection; Section 4, the experiments are done to evaluate our algorithm; Finally, conclusions and further work are proposed in Section 5.

2 Relevant Work

2.1 Collaborative Filtering Algorithm

Traditional collaborative filtering algorithm is mainly divided into two kinds: User-based Collaborative Filtering algorithm (UBCF) [12] and Item-based Collaborative Filtering algorithm (IBCF) [13]. The UBCF algorithm assumes that the users who have the similar preference may like the same items; and the IBCF algorithm is based on the assumption that users will be interested in the items which are similar to those they have visited. In real-world application, these two algorithms usually use Nearest Neighbor algorithm (KNN): select k nearest neighbors for user or item from similar users or items; and then compute a prediction for a user-item pair based on these nearest neighbors, which can either be user-based or item-based [14]. In both cases, in order to find nearest neighbors, we need a previous step of computing similarities between all users or all items. This is a time-consuming and noise-impressionable operation that is computed in a centralized manner [15].

2.2 Expert-based Collaborative Filtering Algorithm

The traditional collaborative filtering algorithm should compute the similarities between users or items. With the increase of users or items, the computing time will increase with index level. And the accuracy of similarities is also limited by noise introduced by users, malicious attacks for instance. Expert-based collaborative filtering algorithm is a radically different approach, according to which recommendations for users are derived from ratings of domain experts rather than from peers [15]. Literature [7] has proved that this algorithm is able to predict the ratings of a large population by considering a small set of experts' ratings. It means this algorithm finds neighbors for target user in the expert set and uses these expert neighbors' ratings to predict the target user's ratings for items. This algorithm does not need to compute the every user-user similarity; instead, it builds a matrix of similarities between each user and all the experts. In order to select the nearest neighbors for a target user, firstly compute the similarities between this user and all the experts; secondly select k experts who have the biggest similarities as this user's neighbors. Formally: given the set of users $U = \{u_1, ..., u_n\}$, the set of experts $E = \{e_1, ..., e_m\}$, the rating vector $V = \{v_1, ..., v_k\}$ of a user or an expert, a pre-determined similarity measure $Sim = \{V \times V \to R\}$, a target user $u \in U$ and a value $k \in R$, find the set of experts $E' \subseteq E$ meet: $\forall e \in E' \Rightarrow Top_k(Sim_{ue})$. The processes of expert-based collaborative filtering algorithm are as follows:

Algorithm 1 : expert-based collaborative filtering algorithm

Input : user-item rating matrix M, target user u, the rated item set I_u of user u

Output : $top - N$ recommendation list for user u

Step 1 : find out all experts and build the expert set E;

Step 2 : compute the similarities between each user and all the experts according to user-item rating matrix M and the pre-determined similarity measure;

Step 3 : for target user u, get k nearest neighbors from E and build neighbor set N;

Step 4 : for every neighbor $n_i \in N$, find the rated items $I_i = \{i_1, i_2..., i_k\}$, unite all I_i and remove items which exist in I_u to produce candidate item set C.

Step 5 : for every item $j \in C$, predict $u's$ rating for item j, sort the items in C by the predicted ratings, and select the top N items to produce $u's$ recommendation list.

The experts can be found in advance, and the experts will not change frequently. Using the experts as neighbors can avoid the noise brought by users and the users' privacy ratings are not needed to share with others as everyone use the experts' ratings. In literature [7], it is confirmed that the expert-based collaborative filtering algorithm can solve the scalability problem, the malicious attacks and the privacy problem effectively, while maintaining acceptable prediction accuracy and recommendation precision.

3 Cluster-based Expert Selection and Recommendation Production

3.1 Cluster-based Expert Selection

Obviously, the expert selection is the most important step which will influence the following steps and the final recommendation effect. However, there still lacks a principled model for guiding how to select the useful experts. In literature [7], the authors define an expert as an individual that can be trusted to have produced thoughtful, consistent and reliable evaluations (ratings) of items in a given domain. In the characters defined above, some are hard clear measured in lots of applications, thoughtful for instance; some are not suitable for the real-world situation, for example, consistent. As the ratings reflect users' interests and everyone has their own interests, it is reasonable that the ratings for different items are different and the consistent of ratings may be disadvantageous for filtering items because the ratings among the experts could show less differences. So, in this paper, we put forward a redefinition of expert that can be reduced into two components: the activity and the influence in a given domain.

Definition 1 (Activity) *activity is the level of involvement of the expert in his/her domain, with A expressed. We define the activity as the ratio of the number of expert's ratings and the number of total ratings in the domain, with formula as follows:*

$A_e = R_e / \sum_G R$, where e is an expert, R_e is the number of $e's$ ratings, G is a given domain;

Definition 2 (Influence) *influence is the level of authority of the expert in his/her domain, with I expressed. We define the influence as the average of the similarities between an expert and other users in a given domain, with formula as follows:*

$$I_e = 1/n \sum\nolimits_G Sim_{eu},$$

Sim_{eu} is the similarity between expert e and user u, G is a given domain, n is the number of other users;

We also define an influence threshold δ, and a useful expert must has an influence bigger than this minimum influence. So, the experts are the most active users whose influence are bigger than δ in their own domains.

The next step is to find the domains, in real-world application, some users with common interests often form a group. Users in the same group may like the same items, and users in different groups are less probably visit the same items. In this way, a group corresponds to one or more specific items and we call the set of these specific items as a domain. In fact, the domain is relatively stable and users always can find their domains according to their interests. In the calculation, we can find out the domains through clustering users' ratings, and in this paper, we employ k-means to do this clustering. To make sure the cluster number k and start the algorithm, we do a cyclic test according to the evaluation metric put forward by literature [16] and select the initial values from the users who have relatively big numbers of ratings. The evaluation metric is composed by between-cluster separation $d(c_i, c_j)$ and within-cluster scatter $s(c_i)$, the formulas are as follows. The focus of this paper is not the optimization of the clustering algorithm; more clustering optimization algorithms refer to [17, 18]. And we define the parameter p to control the proportion of the expert in each domain, that means the top $p\%$ users who meet the conditions discussed above will be selected as experts in a given domain, formally: $Expert = \forall u \in G \Rightarrow top_{p\%}(A_u \& I_u \geq \delta)$.

$$DB = \frac{1}{k}\sum_{i,i\neq j}^{k} \max \frac{s(c_i)+s(c_j)}{d(c_i,c_j)}, \quad s(c_i) = \frac{1}{c_i}\sum_{x\in c_i} ||x - v_i||, \quad d(c_i,c_j) = ||v_i - v_j||,$$

where c_i means the cluster i, v_i means the center of c_i, the smaller the DB is, the better the cluster is, u is a user belongs to domain G.

3.2 Similarity Measure and Recommendation Production

Taking into account the disadvantage of traditional similarity measures, we employ an adjusted Pearson correlation which includes an adjustment coefficient to solve the problem brought by the number of co-rated items of both users. Given user i and j, the similarity can be calculated as follows:

$$Sim_{ij} = \frac{2|I|}{|i|+|j|} \frac{\sum\limits_{k\in I}(R_{ik}-\bar{R_i})\times(R_{jk}-\bar{R_j})}{\sqrt{\sum\limits_{k\in I}(R_{ik}-\bar{R_i})^2}\times\sqrt{\sum\limits_{k\in I}(R_{jk}-\bar{R_j})^2}},$$

where R_{ik} and R_{jk} are ratings for item k of user i and j, I is the co-rated items of user i and j, $|i|$ is the number of rated items of user i, as well as $|j|$, $\bar{R_i}$ and $\bar{R_j}$ are the average ratings of user i and j.

The new similarity measure is composed by two parts, the later one is the traditional Pearson correlation, and the former one is an adjustment coefficient which reflects the correct degree of the later one. The adjustment coefficient is proportional to the proportion of co-rated items in the total rated items of two users. The larger the proportion is, the greater the adjustment coefficient is, which means the Pearson correlation more correctly reflects the similarity between two users.

We introduce the cluster-based expert selection method and the new similarity measure into existing expert-based collaborative filtering algorithm and propose collaborative filtering algorithm incorporated with cluster-based expert selection. Once the expert neighbors (E_u) for a target user (u) have been selected, for the unrated items of user u, predict the $u's$ ratings by means of a weighted average of the ratings of each expert neighbor e in E_u, the formula is as follows. Then, sort the items by the predicted ratings and recommend the $top-N$ items as user's recommendation list.

$$R_{ui} = \bar{R_u} + \frac{\sum\limits_{k\in E_u} Sim_{uk}\times(R_{ki}-\bar{R_k})}{\sum\limits_{k\in E_u} Sim_{uk}},$$

where R_{ui} is the predication of user u for item i, $\bar{R_u}$ and $\bar{R_k}$ are the average ratings of user u and k, Sim_{uk} is the similarity between user u and user k, E_u is the set of expert neighbors of user u.

4 Experiments and the Analyses

4.1 Dataset and Evaluation Metrics

The dataset that we have employed is the MovieLens provided by GroupLens group of Minnesota University [19]. In this paper, we choose part of the data, including 100000 ratings of 943 users for 1682 movies. The ratings are integer of 1-5, 5 means "perfect" while 1 means "bad", users express their interests by the different ratings for different movies [20]. We divide our dataset into 80% training - 20% testing sets and report the average results of a 5-fold cross-validation. Fig. 1 shows the number of ratings per user. From the figure, we can see that there is a long tail where about half of the users rate less than 100 items, while the most active user rates about 700 items. Fig. 2 depicts the CDF (Cumulative Distribution Function) of the number of ratings. We can see the last 30% users have about 70% ratings, while the first 400 users only 10%.

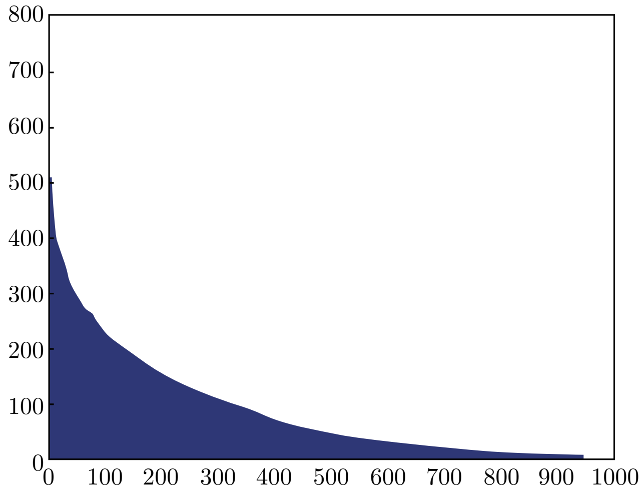

Fig. 1: Number of ratings per user

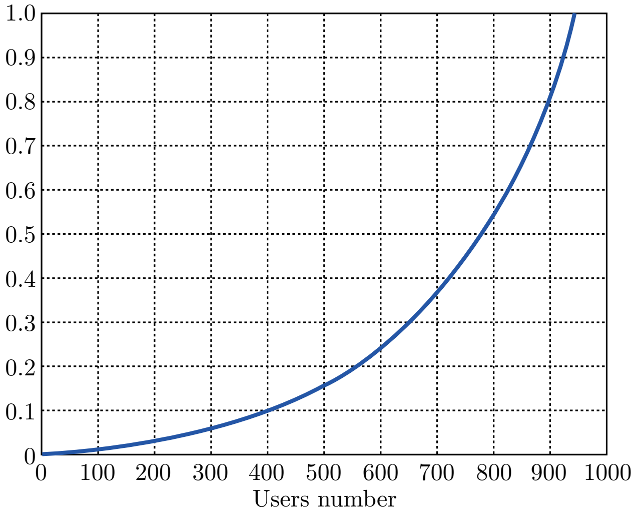

Fig. 2: CDF of the number of ratings

In order to evaluate the prediction accuracy of collaborative filtering algorithm, it is common to measure the Mean Average Error value (MAE), which calculates the deviation between the predicted ratings and the user's real ratings [21]. The smaller the MAE is, the higher the predication accuracy is. If the predicted ratings for items are $p_1, p_2 ... p_n$, and the corresponding user's real ratings are $r_1, r_2 ... r_n$, the MAE is defined as follows:

$$MAE = \frac{\sum\limits_{i=1}^{n} |p_i - r_i|}{n}.$$

In real-world application, collaborative filtering algorithm takes top-N items to produce recommendation list for target user. That is recommending some good items to target user, but not recommending all good items to target user. So, another evaluation metric our interested in is not the recall but the precision which is popular in information retrieval field. For a target user, if a recommended item i also exists in testing set, it means a correct recommendation, otherwise a wrong one. Suppose that tp means the number of correct recommendations, and fp means the wrong number, then the precision is defined as follows:

$$precision = \frac{tp}{tp + fp}$$

4.2 Experimental Results and the Analyses

Experiment 1: we firstly look at the correlation between the number of experts and the influence threshold δ. In this experiment, we take $p = 0.1$, which means the number of experts is about tenth of the number of total users, of course the value of p can be others. Fig. 3 shows the curve of the number of experts with the influence threshold δ, the horizontal axis is the value of δ, which increases from 0.02 to 0.11 at the interval 0.01. It can be seen the number of experts remains stable until $\delta = 0.06$, then it starts to decrease sharply as δ moves to bibber values. The number of experts remains stable when $\delta \leq 0.06$, because there are enough users meeting the conditions discussed in Section 3.1; with the increase of δ, though, we have fewer and fewer users meeting the conditions.

Experiment 2: we compare the number of ratings per expert selected by our CBES algorithm and the one used in literature [7] (paper [7] expressed in the following figures). In this experiment, we take $p = 0.1$, $\delta = 0.06$, the number of cluster $k = 20$. With our CBES algorithm, we get 95 experts. At the same time, we choose the same number of experts from the GroupLens web site according to the method used in literature [7]. We sort the experts by the number of ratings and number them 1-95, with the horizontal axis expressed in Fig. 4. We can see that all the experts selected by the later method have more than 200 ratings, which means this method selects experts focusing on the users who have lots of ratings. On the contrary, the number of ratings of experts selected by our CBES algorithm is more even, which means our method not only focuses on the users with lots of ratings but also the users with a few of ratings.

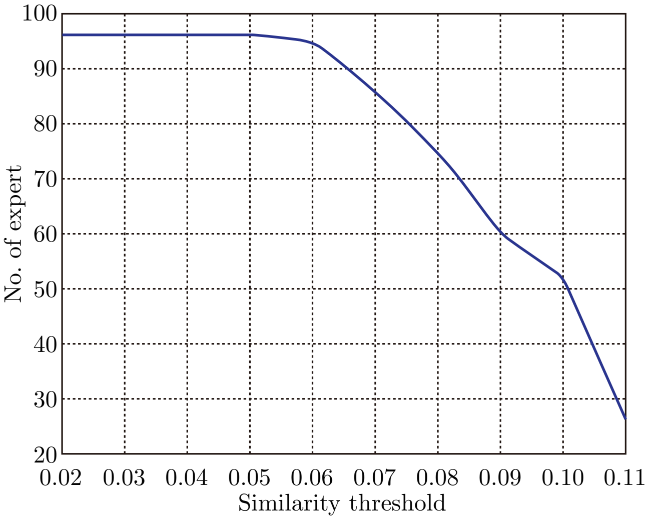

Fig. 3: Number of experts with influence threshold

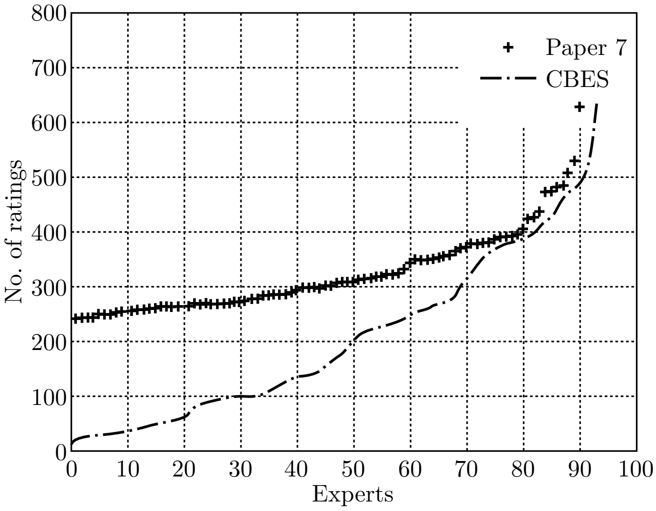

Fig. 4: Number of ratings per expert

Experiment 3: in order to verify the effect of our CBES algorithm, we compare our CBES algorithm, traditional UBCF algorithm and the one used in literature [7] on the predication accuracy and recommendation precision. In this experiment, we use the experts selected in Experiment 2. Fig. 5 depicts the comparison on MAE, the horizontal axis is the number of expert neighbors N_e, which increases from 10 to 90 at the interval 10. We can see the MAE of our CBES algorithm is inversely proportional to the number of expert neighbors until $N_e = 30$, then it remains stable. The MAE decreases rapidly when the number of neighbors is below 30, as we are taking into account more and more useful experts. It also can be seen that the traditional UBCF algorithm has lower MAE than these two expert-based algorithms and our CBES algorithm has a MAE 1.8% lower than the one used in literature [7].

Fig. 6 shows the comparison on recommendation precision, the horizontal axis is the value of N in top-N, which increases from 1 to 20 at the interval 2. In this experiment, we take the number of neighbors $N_e = 30$. From the figure, we can see the precisions of three algorithms are decreasing with the increase of N, because with the increase of N, there are more and more recommended items that users are not interested in. And we can see our CBES algorithm has a precision 12% higher than the one used in literature [7]. That is because our experts can take account of all

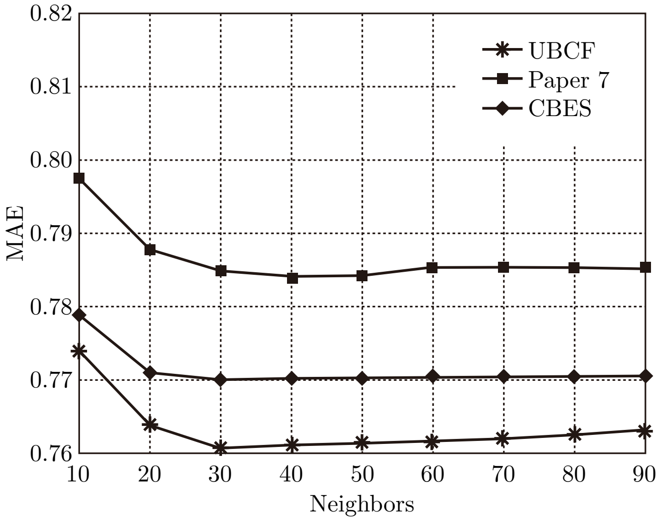

Fig. 5: Comparison on MAE

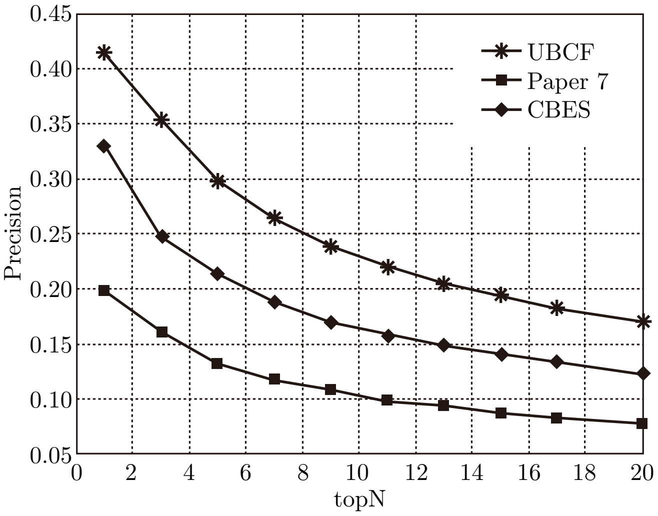

Fig. 6: Comparison on precision

types of users, not only the ones with lots of ratings but also the ones with a few of ratings, and the users with a few ratings play an important role in recommendation system according to the long tail theory.

5 Conclusions

Expert-based collaborative filtering algorithm provides an efficient way to solve the scalability problem and the malicious attacks, while maintaining acceptable prediction accuracy and recommendation precision. But there is not an effective method to select the useful experts. In this paper, we firstly define a concept of expert which can be reduced into two components: the activity and the influence. Basing on the definition, we put forward cluster-based expert selection method by clustering users' interests and propose collaborative filtering algorithm incorporated with cluster-based expert selection. In order to validate our algorithm, we compare our algorithm with the one used in literature [7], and the result shows that our algorithm outperform the exist one both in the respect of predication accuracy and recommendation precision.

Although the algorithm proposed in this paper has achieved good results, there is also possibility for future improvement. We can improve the recommendation precision by tracking the changes of user's interest. Meanwhile the clustering method will produce outlier, which means a particular user with special interest or with too few ratings. How to deal with this user is another research content.

References

[1] D. Bawden, C. Holtham, N. Courtney, Perspectives on information overload, Aslib Proceedings: New Information Perspectives, 51(8), 1999, 249-255

[2] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 2005, 734-749

[3] A. L. Deng, Y. Y. Zhu, B. O. Shi, A collaborative filtering recommendation algorithm based on item rating prediction, Journal of Software, 14 (9), 2003, 1621-1628 (in Chinese)

[4] J. B. Schafer, D. Frankowski, J. Herlocker et al., Collaborative filtering recommender systems, Lecture Notes in Computer Science, 4321, 2007, 291-324

[5] M. P. O'Mahony, N. J. Hurley, G. C. M. Silvestre, Detecting noise in recommender system databases, Proceedings of the 11th International Conference on Intelligent User Interfaces, 2006, 109-115

[6] Z. P. Cheng, N. Hurley, Effective diverse and obfuscated attacks on model-based recommender systems, Proceedings of the Third ACM conference on Recommender Systems, 2009, 141-149

[7] X. Amatriain, N. Lathia, J. M. Pujol et al., The wisdom of the few: A collaborative fltering approach based on expert opinions from the web, Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2009, 532-539

[8] H. Polat, W. L. Du, SVD-based collaborative filtering with privacy, Proceedings of the 2005 ACM Symposium on Applied Computing, 2005, 791-795

[9] G. Y. Xue, C. X. Lin, Q. Yang et al., Scalable collaborative filtering using cluster-based smoothing, Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2005, 114-121

[10] M. O'Mahony, N. Hurley, N. Kushmerick et al., Collaborative recommendation: A robustness analysis, ACM Transactions on Internet Technology, 4(4), 2004, 344-377

[11] R. M. Bell, Y. Koren, Improved neighborhood based collaborative filtering, KDD Cup and Workshop at the 13th Association for Computing Machinery Special Interest Group on Knowledge Discovery in Data International Conference on Knowledge Discovery and Data Mining, 2007, 7-14

[12] P. Resnick, N. Iacovou, M. Suchak et al., GroupLens: An open architecture for collaborative filtering of netnews, Proceedings of the ACM CSCW'94 Conference on Computer Supported Cooperative Work, 1994, 175-186

[13] G. Linden, B. Smith, J. York, Amazon.com recommendations: Item-to-item collaborative filtering, IEEE Internet Computing, 7(1), 2003, 76-80

[14] H. L. Xu, X. Wu, X. D. Li et al., Comparison study of internet recommendation system, Journal of Software, 20(2), 2009, 350-362 (in Chinese)

[15] J. W. Ahn, X. Amatriain, Towards fully distributed and privacy-preserving recommendations via expert collaborative filtering and RESTful linked data, AIEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2010, 66-73

[16] D. L. Davies, D. W. Bouldin, A cluster separation measure, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1(4), 1979, 224-227

[17] J. G. Sun, J. Liu, L. Y. Zhao, Clustering algorithms research, Journal of Software, 19 (1), 2008, 48-61 (in Chinese)

[18] C. Carpineto, S. Osiski, G. Romano et al., A survey of web clustering engines, ACM Computing Surveys, 41(3), 2009, 1-38

[19] GroupLens, Movielens, http://www. grouplens. org/node/73. [2011-08-24]

[20] J. Chen, J. Yin, A collaborative filtering recommendation algorithm based on influence sets, Journal of Software, 18(7), 2007, 1685-1694 (in Chinese)

[21] L. Herlocker, J. A. Konstan, L. G. Terveen et al., Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems, 22(1), 2004, 5-53