TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol. 12, No. 8, August 2014, pp. 6324 ~ 6331
DOI: 10.11591/telkomnika.v12i8.4617

Cliques-based Data Smoothing Approach for Solving Data-Sparsity in Collaborative Filtering

Yang Yujie*, Zhang Zhijun, Duan Xintao
College of Computer and Information Engineering, Henan Normal University,
Xinxiang, Henan, China, 453007
*Corresponding author, e-mail: yujieyangyujie@gmail.com

Abstract

Collaborative filtering (CF), as a personalized recommending technology, has been widely used in e-commerce and other many personalized recommender areas. However, it suffers from some problems, such as cold start problem, data sparsity and scalability, which reduce the recommendation accuracy and user experience. This paper aims to solve the data sparsity in CF. In the paper, cliques-based data smoothing approach is proposed to alleviate the data sparsity problem. First, users and items are divided into many cliques according to social network analysis (SNA) theory. Then, data smoothing proceeding is carried out to fill the missing ratings in user-item rating matrix based on the user and item cliques. Finally, the traditional user-based nearest neighbor recommendation algorithm is used to recommend items for users. The experiments show that the proposed approach can effectively improve the accuracy and performance on sparse data.

Keywords*: data sparsity, collaborative filtering, clique, social network analysis, data smoothing*

1. Introduction

Collaborative filtering (CF) can help toovercome "information overload" and to provide personalized services in social networking web site. Generally speaking, collaborative filtering can be categorized into memory-based algorithmand model-based algorithm [1]. The CF has been applied in many areas successfully, such as book sites, movie sites and some e-commerce sites [2]. However, the CF also suffers from a lot of issues, such as cold start problem, data sparsity and scalability [3].

This paper aims to solve the data sparsity problem. Data sparsity problem will occur when either few ratings are available for the active user, or for the target item that prediction refers to, for the entire user-item rating matrix in average [4]. The existing solutions of data sparsity include dimensionality reduction technique [5], data smoothing technique[6] and associative retrieval technique [7] etc. Paper [8] proposed a novel goal-based hybrid approach to overcome the cold-start problem in e-learning internet. And it also helps to improve collaborative filtering using k-nearest neighbor as neighborhood collaborative filtering (NCF) and content-based filtering as content-based collaborative filtering (CBCF). Paper [9] proposed a social recommender system that follows user's preferences to provide recommendation based on the similarity among users participating in the social network. And the approach which it proposed was based on integration of major characteristics of content-based and collaborative filtering techniques.

In this paper, cliques-based data smoothing approach is proposed to solve the data sparsity problem in collaborative filtering. Firstly, user social network and item social network are built. And then, users and items are divided into many cliques respectively according to social network analysis (SNA). In order to fill the missing ratings, data smoothing operation is carried out by using these cliques. Finally, the traditional user-base nearest neighbor recommendation algorithm is used to recommend items for users. The experiments indicate that this novel approach can effectively improve the recommendation accuracy and performance.

The remainder of this paper is organized as follows. In section 2, we give a brief survey of the related work on the solution of data sparsity in CF. Section 3 describes the proposed algorithm in detail. Section 4 presents the experimental results and analysis. Finally, section 5 gives the conclusion.

Received October 3, 2013; Revised March 7, 2014; Accepted April 2, 2014

2. Related work

Collaborative filtering recommends items to users according to their preferences. Therefore, a history database of users' preferences must be available. However, the database is always very sparse. This leads to the reduction of recommendation accuracy and performance. Data sparsity is an inevitable problem with all kinds of CF algorithms. Data sparsity includes two aspects. On the one hand, the number of user rating is very small compared to the number of items. On the other hand, the overlapping number of two user rating is very few. There are many researchers who have focused on the data sparsity problem and proposed some solutions.

Dimensionality reduction technique, such as principle component analysis (PCA) [10] and singular value decomposition (SVD) [11], is commonly used to alleviate data sparsity. Reference [5] combined the SVD and item-based recommender in CF. It utilized the results of SVD to fill the missing ratings and then used the traditional item-based method to recommend. This combination method can increase the accuracy of system. Reference [12] investigated a hybrid recommendation method which was based on two-stage data processing-dealing with content features describing items and handing user behavioral data. This hybrid method combined random indexing (RI) technique and SVD to preprocess the content features. The experiments improved the recommendation accuracy without increasing the computational complexity.

Data smoothing technique is the most used method to solve the data sparsity problem in CF. Various sparsitymeasures [13] were used to enhance accuracy of CF. These sparsity measures were computed based on local and global similarities. Then, an estimating parameter scheme for weighting the various sparsity measures was proposed. The experimental results demonstrated that the proposed estimate parameter outperform the schemes for which the parameter is kept constant on accuracy of prediction ratings. Reference [14] proposed a partial missing data prediction algorithm, in which the information of both users and items was taken into account. In this algorithm, similarity threshold for users and items was set respectively, if and only if the intersection of the neighbor of user and the neighbor of item is not empty, the missing data will be predicted. An iterative prediction method [15] was proposed to alleviate the sparsity problem in CF.This method clusters the user and item respectively by using spectral clustering algorithm. Then, the iterative prediction technique is used to convert user-item sparse matrix to dense one based on the explicit ratings. Moreover, cluster-based smoothing method [16], support vector machine(SVM) [17], BP neural networks [18] and zero-sum reward and punishment mechanism[19] are also applied to smooth the missing ratings for the solution of data sparsity in CF.

With the development of social network, social network analysis (SNA) [20] theory has been applied to recommender systems. Reference [21] proposed to use social network to solve data sparsity problem in one-class CF. It compared social networks belong to specific domains and the ones belong to more generic domains in terms of their usability in one-class CF problems. Associative retrieval technique was applied to alleviate the sparsity problem in CF. [22] gives a social network representation for CF recommender systems. It shows some of the advantages and results that can be obtained applying SNA. Reference [23] gave a book recommendation based on web social network. It analyzed the problem of trust in social network and proposed a recommender system model based on social network trust. Reference [24] presented a framework of recommendations based on information network analysis. Reference [25] proposed a new weighting method in network-based recommendation. This method presents a new expression of initial resource distribution and takes into account the influence of resource associated with receiver nodes.

In this paper, cliques-based data smoothing technique is proposed to solve the data sparsity problem in CF. First, the similarity of user and item is computed respectively and the user and item social networks are built based the similarity. Then, all users and items are divided into many cliques according to SNA theory. The missing ratings of testing users will be predicted. This prediction will take into account both user and item. The prediction values from user and item are weighted together as the smoothing value. Finally, the traditional user-base nearest neighbor recommendation algorithm is used to recommend items for user. The experiments demonstrate that the proposed algorithm is effectively improving the recommendation accuracy.

3. Cliques-based Data Smoothing Algorithm

The application of social network analysis theory in recommender systems is becoming more and more important. The potential user relationship can be mined to recommend information or items for users. The most contribution of this paper is performming data smoothing by means of cliques of user and item together. The smoothing rating matrix is used to the recommendation.This recommendation algorithm can improve the recommendation performance effectively.

3.1. Building Social Network

A network is composed of nodes and the relations among nodes. Formally, let us consider a network as a graph *G=(U,E)* in which *U* represents nodes and *E* represents links. In this paper, the users or items represent the nodes and, the similarities among users or items denote the relations. We will build the user social network and item social network respectively.

In order to build the user relation network, firstly, we need compute the similarity among each pair users. Assume that$U=\{u_1,u_2,...,u_N\}$denotes the set of users, $P=\{p_1,p_2,...,p_M\}$ for the set of items, and$R$ as an $N\times M$ matrix of ratings $r_{i,j}$, with $i \in 1,...,N$, $j \in 1,...,M$. There are many algorithms to determine the similarity among users: Pearson's correlation coefficient, cosine similarity, and adjusted cosine measure [26] and so on. In the paper, Pearson's correlation coefficient is used. So, the similarity between user u_i andu_j is as follows:

$$sim(u_i,u_j)=\frac{\sum_{p\in P_{u_i}\cap P_{u_j}}[(r_{i,p}-\overline{r_i})*(r_{j,p}-\overline{r_j})]}{\sqrt{\sum_{p\in P_{u_i}\cap P_{u_j}}(r_{i,p}-\overline{r_i})^2*(r_{j,p}-\overline{r_j})^2}} \quad (1)$$

Where $\overline{r_i}$ and $\overline{r_j}$ corresponds to the average rating of user u_i and $u_j$respectively. P_{u_i} denotes the item set of user$u_i$rating. P_{u_j} denotes the item set of user$u_j$rating. In practice, because the amount of items is very large, users may only rate few items. The number of overlapping item among users may be very few. This leads to the inaccurate similarity. For more accuracy of the similarity, a parameter γ which denotes the overlapping number of rating between two users will be added to adjust. So, the improved formula is as follows:

$$sim'(u_i,u_j)=\begin{cases}sim(u_i,u_j),\gamma\geq T\\0,\gamma<T\end{cases} \quad (2)$$

Where T is a threshold value. The larger of the value$T$,the more accuracy of the similarity.

The building of item network is as same as the user network. The difference is that the similarity of item, rather than the similarity of user, will be computed. For the same reason, a parameter γ which denotes the overlapping number of rating between two items will be added to adjust. So, the improved formula is as follows:

$$sim'(p_i,p_j)=\begin{cases}sim(p_i,p_j),\gamma\geq T\\0,\gamma<T\end{cases} \quad (3)$$

After computing the similarity, the similarity value needs binary processing in order to building the social network. Considering the cliques division (describing next section), the binary threshold requires suitable in order to obtain appropriate cliques.

3.2. Cliques Division

According to SNA theory, cliques are some sub-structures of the network. From the view of social structure, clique focuses attention on how solidarity and connection of social network. The general definition of a clique is simply a sub-set of nodes which are more closely tied to each other than they are to nodes which are not part of the group. More accurately, it

insists that every member have a direct tie with each and every other member. In our approach, the existing users and items will be divided into many cliques respectively.

UCINET is a kind of network analysis software. It can make all kinds of network analysis, such as network structure, centralization and so on. We make cliques division by means of UCINET in this paper.

Compared to k-means cluster algorithm [16], clique has many advantages. On one hand, k-means cluster divides the similar users into the same cluster, however for oneuser,he/she is divided only one cluster. Intuitively, each user may have many interests and they may join a few of communities. So, the user should belong to several clusters. Cliques can avoid this obstacle. On other hand, k-means algorithm requires the *k* less *n*, which *k* denotes the number of clusters, *n* is the number of users. In fact, the cluster number may be more than that of the users. However, clique number can more than users. Finally, clique can also present the user relation better. It is not only considering the direct relationships, but also the transmission relationships. Sowe cluster users and items by using clique theory rather than k-means cluster algorithm.

3.3. Cliques-based Data Smoothing

Because the overlapping number of rating items between users is small or none, it leads the accuracy of similarity is very low. In order to enhance the accuracy, it is necessary to smooth the missing rating of user-item rating matrix.

In this paper, the predictive value of missing rating is from two aspects: user cliques and item cliques. First, the clique's members of user and item are collected respectively. Then, the predictive rating will be computed based on user's cliques and item's cliques respectively. Finally, the weighted value of the two predictive rating will be the final predictive value of the missing rating. The weighted formula is as follows:

$$S_{mis}(r_{ij}) = \beta \cdot S_{u_i} + (1-\beta) \cdot S_{p_j} \tag{4}$$

$$S_{u_i} = \sum_{u_k \in C_{u_i}} (w_{ik} \cdot r_{kj}) \tag{5}$$

$$S_{p_j} = \sum_{p_k \in C_{p_j}} (w_{jk} \cdot r_{ik}) \tag{6}$$

Where S_{u_i} is the predictive value which relevant to the cliques of user u_i. C_{ui} represents the cliques set of useru_i. w_{ik} is the similarity between user $u_i$and u_k. S_{p_j} denotes the predictive value according to the cliques of item p_j.C_{pj}represents the cliques set of item p_j.$w_{jk}$is the similarity between item $p_j$and p_k. β is a significance weighting factor. $S_{mis}(r_{ij})$ is the final smoothing value of missing ratingr_{ij}.

3.4. Prediction for Active User

After the missing ratings are predicted in the user-item matrix, we can recommend items for the active users. In this paper, the traditional user-based nearest neighbor recommendation algorithm is adopted. For the active user ui, the predictive value of item pj can be computed as follows:

$$pre(u_i, p_j) = \frac{\sum_{k=1}^{M} (w_{ik} \cdot r_{kj})}{\sum_{k=1}^{M} w_{ik}} \tag{7}$$

Where *M* is the number of neighbor of user u_i. The predictive values are sorted according to the descending. The *TopN* items will be selected to the user u_i.

The whole step of the proposed approach is as follows:

(1) Computing the similarities among users and items respectively according to the user-item rating matrix, then building user relation network and item relation network based on thesesimilarities.

(2) Dividing all users and items into many cliques respectively.

(3) Smoothing the missing rating according to user and item cliques together.

(4) Finally, user-based nearest neighbor recommendation is used to predict items for new users.

4. Experimental Results

4.1. Dataset

The MovieLens (http://www.movielens.umn.edu) dataset is used in this paper. In MovieLens, there are 100,000 ratings with 943 persons and 1682 movies. And each person had rated at least 20 movies. The user information includes age, sex, and occupation and so on. The movie includes 19 types. The density of the user-item matrix is 6.3%.

First, the dataset is divided into two parts, 20% of all persons are selected to be testing set, and the remaining as training set. For measuring accuracy, we conducted a 5-fold cross validation by uniformly choosing different training and test sets. In order to better evaluate the performance of new approach, we take the testing users from dataset uniformly. That is, the degree of all users is firstly computed and sorted by order. Then, the testing set is selected by equal intervals. So, the testing set includes all kinds of users. For the training set, first, the similarities among each pair users and items will be computed respectively according to Equation (1), (2) and (3). In order to improve the accuracy, in the experiment, the parameter T will be set as 2 in Equation (2) and 5 in Equation (3). Each users and items represents nodes, and similarities form the relations among the user network and item network respectively.

4.2. Performance Evaluation

In order to estimate the performance of the proposed approach, the precision of prediction is measured with three different metrics.

Recall: The recall score is the average proportion of items from test set that appear among *TopN* of the ranked list from the training set [27]. This measure should be as high as possible for good performance. Assume N is the number of items which are in the testing set and liked by users, n is the amount of items which the testing user likes and appears in the recommended list. So, the recall is computed as follows:

$$Recall = \frac{n}{N} \tag{8}$$

Precision: The precision is the proportion of recommended items that the testing users actually liked in the test set [28]. This measure is also as high as possible for good performance. The precision is computed as follows:

$$Precision = \frac{n}{TopN} \tag{9}$$

F-measure: It is also known as the F_1measure, which combines precision and recall into a single metric by taking the harmonic mean of them[28]. So, the F-measure is computed as follows:

$$F_1 = \frac{(2 * Recall * Precision)}{(Recall + Precision)} \tag{10}$$

4.3. Results and Analysis

For building the user and item social network respectively, first, utilizing Pearson correlation coefficient algorithm, each pair user's and each pair item's similarity will be computed. Then, binary all similarities are constructed as follow: we set a relation threshold R_T

(it may be different as for user and item network), similarity values greater or equal than are set 1, that is, the user pair or item pair has link, otherwise 0.

For convenience, we use TCF as the traditional user-based nearest neighbor recommendation algorithm and the proposed algorithm is express as Clique-CF. Figure 1, Figure 2 and Figure 3 gives the comparisons of two algorithms in recall, precision and F-measure with different *TopN* values. In these figures, the number of neighbor *M* is selected as 100. The parameter β is set 0.5.From Figure 1, we can know that the recall value of TCF and Clique-CF is gradually increasing with the increasing of *TopN*. However, the performance of Clique-CF is better than the TCF. The recall of Clique-CF is larger than TCF constantly. Further, with the increasing of *TopN*, the gap is becoming larger. Figure 2 depicts the comparison in precision. The figure shows that the precision value of both TCF and Clique-CF is gradually decrease with the increasing of *TopN*. However, theperformance of Clique-CF is better than TCF. The precision of Clique-CF is larger than TCF always. But, it is different from Figure 1, the gap between Clique-CF and TCF is becoming smaller with the increasing of *TopN*.

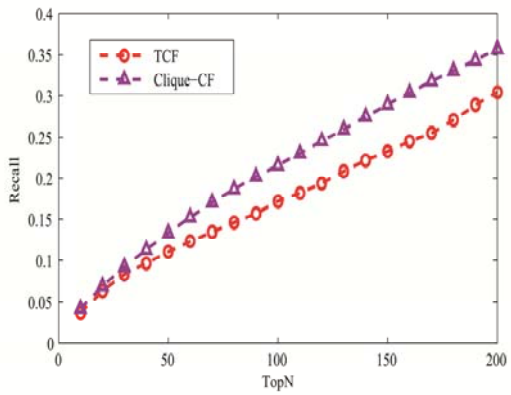

Figure 1. The Comparison of Recall between TCF and Clique-CF with Different *TopN*

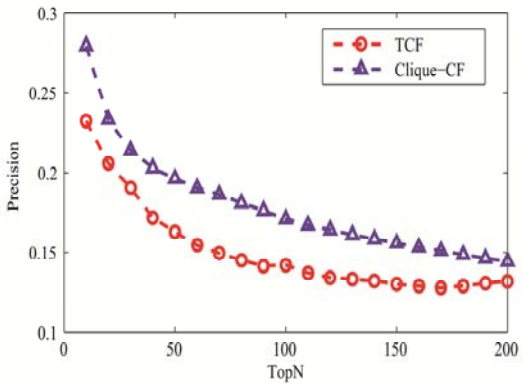

Figure 2. The Comparison of Precision between TCF and Clique-CF with Different *TopN*

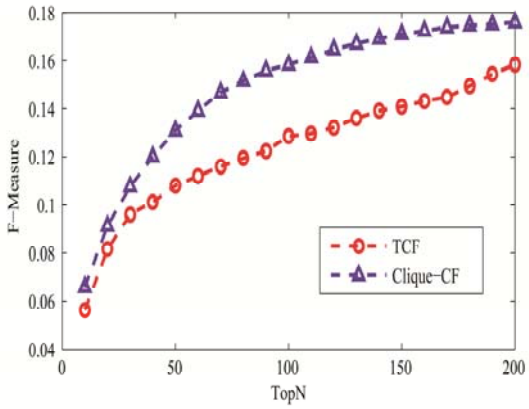

Figure 3. The Comparison of F-measure between TCF and Clique-CF with Different *TopN*

Figure 3 describes the changing of F-measure about Clique-CF and TCF with different *TopN*. From the figure, we can see that the F-measure value of both Clique-CF and TCF is also gradually increasing with the increasing of *TopN*. As same Figure 1and Figure 2, the performance of Clique-CF is better than TCF. From the analysis, we can see that the proposed algorithm is better than the traditional collaborative filtering algorithm in recall, precision and F-

measure.In the above, we set the number of neighbor as a solid value. Figure 4 gives the performance of Clique-CF with different neighbors *M*. And the *TopN*is set 100. From Figure 4, the performance is becoming better with the increasing of*M*. Further, the performance increases fast when *M* is less than 50 and then it becomes stable with more *M*.

Finally, in time performance, the computing of similarities between each pair users and each pair users, the building of user relation network and item relation network and dividing cliques can be implemented in offline. So, its speed is almost at the same level compared with the traditional recommender systems.

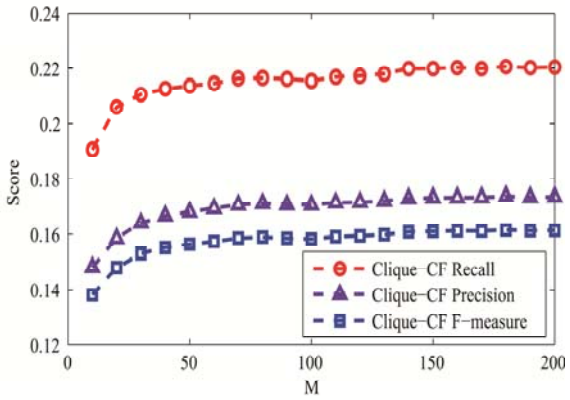

Figure 4. The Performance of Cliques-based Data Smoothing Algorithm with Different Neighbor *M*

5. Conclusion

This paper proposes a cliques-based data smoothing algorithm to solve the data sparsity problem in collaborative filtering. First, the similarities of users and items are computed respectively and the user social network and item social network can be built. Then, all users and items are divided into many cliques according to social network analysis theory. The missing rating of the user-item rating matrix will be filled according to the predictive value from user cliques and item cliques. Finally, we proposed the traditional user-based nearest neighbor recommendation algorithm. The experimental results show that the proposed algorithm performs better than the traditional collaborative filtering algorithm.

References

[1] Adomavicius G, Tuzhilin A. 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.* 2005; 17(6): 734-749.
[2] Linden G, Smith B, York J. *Amazon comrecommendations: item-to-item collaborative filtering*. Report number. 7(1): 2003.
[3] Grcar M, Mladenic D, Fortuna B, Grobelnik M. *Data Sparsity Issues in the Collaborative Filtering Framework*. Advances in Web Mining and Web Usage Analysis.Berlin, Heidelberg. 2006: 58-76.
[4] DanEr Chen. *The Collaborative Filtering Recommendation Algorithm Based on BP Neural Networks.* International Symposium on Intelligent Ubiquitous Computing and Education .ChengDu. 2009: 234-236.
[5] Song Jie Gong, Hong Wu Ye, YaE Dai. *Combining Singular Value Decomposition and Item-based Recommender in Collaborative Filtering.* International Workshop on Knowledge Discovery and Data Mining. MOscow. 2009: 769-772.
[6] Rong Hu, Yansheng Lu. *A Hybrid User and Item-based Collaborative Filtering with Smoothing on Sparse Data.* Proceedings of the International Conference on Artificial Reality and Telexistence-Workshops. Hangzhou. 2006: 184-189.
[7] Zan Huang, Hsinchun Chen, Daniel Zeng. 2004. Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering. *ACM Transaction on Information System*. 2004; 22(1): 116-142.

[8] Muhammad Waseem Chughtai, Ali Selamat, Imran Ghani. Goal-based hybrid filtering for user-to-user personalized recommendation. *International Journal of Electrical and Computer Engineering.* 2013; 3(3): 329-336.
[9] Abeer El-Korany, Salma MokhtarKhatab. Ontology-based Social Recommender System. *IAES International Journal of Artificial Intelligence.* 2012; 3(1): 127-138.
[10] Goldberg K, Roeder T, Gupta D, Perkins C. Eigentaste: A Constant Time Collaborative Filtering Algorithm. *Information Retrieval*. 2001; 4(2): 133-151.
[11] Sarwar BM, Karypis G, Konstan JA, Riedl J. *Application of Dimensionality Reduction in Recommender System - A Case Study.* ACM WebKDD Workshop. Report number: 0704-0188. 2000.
[12] Szwabe A, Ciesielczyk M, Janasiewicz T. *Semantically Enhanced Collaborative Filtering Based on RSVD.* Proceedings of the Third International Conference on Computational Collective Intelligence. Gdynia. 2011: 10-19.
[13] Anand D, Bharadwaj KK. Utilizing Various Sparsity Measures for Enhancing Accuracy of Collaborative Recommender Systems Based on Local and Global Similarities. *Expert Systems with Applications: An International Journal*. 2011; 38(5): 5101-5109.
[14] Hao Ma, King I, Michael RL. *Effective Missing Data Prediction for Collaborative Filtering*. Proceedings of the annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Amsterdam. 2007; 39-46.
[15] Abdelwahab A, Sekiya H, Matsuba I, Horiuchi Y, Kuroiwa S. *Collaborative Filtering Based on an Iterative Prediction Method to Alleviate the Sparsity Problem*. Proceedings of the International Conference on Information Integration and Web-based Applications & Services. Kuala Lumpur. 2009; 375-379.
[16] Guirong Xue, Chenxi Lin, P Qiang Yang, Wensi Xi, P Hua Jun Zeng, P Yong Yu, P Zheng Chen. *Scalable Collaborative Filtering Using Cluster-based Smoothing*.Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Salvador. 2005; 114-121.
[17] Grcar M, Mladenic D, Fortuna B, Grobelnik M. *Data Sparsity Issues in the Collaborative Filtering Framework.* Advances in Web Mining and Web Usage Analysis. Chicago. 2006; 4198: 58-76.
[18] DanEr Chen. *The Collaborative Filtering Recommendation Algorithm Based on BP Neural Networks*. Proceedings of the International Symposium on Intelligent Ubiquitous Computing and Education. Chengdu. 2009; 234-236.
[19] Nan Li, Chunping Li. *Zero-Sum Reward and Punishment Collaborative Filtering Recommendation Algorithm.* Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence Agent Technology. Milan. 2009; 548-551.
[20] Hanneman RA, Riddle M. Introduction to Social Network Methods. California: University of California. 2005: 17-41.
[21] Kaya H, Alpaslan FN. *Using Social Networks to Solve Data Sparsity Problem in One-Class Collaborative Filtering.* Proceedings of the Seventh International Conference on Information Technology: New Generations. Las Vegas. 2010; 249-252.
[22] Perez LG, Chiclana F, Ahmadi S. *A Social Network Representation for Collaborative Filtering Recommender System.* International Conference on Intelligent Systems Design and Applications. *Córdoba.* 2011; 438-443.
[23] Mingjuan Zhou. *Book Recommendation Based on Web Social Network*. International Conference on Artificial Intelligence and Education. Hangzhou. 2010; 136-139.
[24] Xue Li, Ling Chen. *Recommendations based on Network Analysis*. International Conference on Advanced Computer Science and Information System. Jakarta. 2011; 9-16.
[25] Chun Xiao Jia, Run Ran Liu, Duo Sun, Bing Hong Wang. A New Weighting Method in Network-based Recommendation. *Physica A: Statistical Mechanics and its Applications*. 2008; 387(23): 5887-5891.
[26] Herlocker JL, Konstan JA, Borchers A, Riedl J. *An Algorithmic Framework for Performing Collaborative Filtering*. Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Berkeley. 1999; 230-237.
[27] Francois F, Alain P, Jean-Michel R, Marco S. Random-Walk Computation of Similarities between Nodes of a Graph with Application to Collaborative Recommendation. *IEEE Transactions on Knowledge and Data Engineering*. 2007; 19(3): 355-369.
[28] Cane WL, Stephen CC, Chung F. An Empirical Study of a Cross-level Association Rule Mining Approach to Cold-Start Recommendations. *Knowledge-Based Systems*. 2008; 21(7): 515-529.