A Dynamic Prediction Model for Recommender Systems Based on the Doubly Structural Network

Na Chang, Tokyo Institute of Technology, China, bacenana@gmail.com
Takao Terano, Tokyo Institute of Technology, Japan, terano@plum.plala.or.jp

Abstract: In the context of recommender systems, there are two types of enties: users and items, and three types of relationships: users' relationship, items' connection and interactions between users and items. In most literatures, one or more of these entities and relationships are used to predict users' preference or taste. In this paper, we propose a novel approach which incorporates these two entities and three relationships into one framework based on doubly structural network (DSN) . We also develop a dynamic prediction model to learn users' preference over time by focusing on the active user-item pair's influence on the corresponding neighborhood. We conduct an experiment and anylize the sensitivity of the model's parameters and compare the new approach with conventional collaborative filtering (CF) approaches and the results show that the new approach could give a better performance than user-based CF and item-based CF approaches for recommender systems.

Key-Words: *recommender systems, doubly structural network, expected preference, predictive preference, dynamic prediction model*

1. Introduction

With the development of E-commerce, personalized recommendation service becomes one important need for users. Recommender systems are information filtering systems which use users' individual information such as histories of purchasing and items' contents to predict users' preferences [1]. Based on these ideas, recommender systems will then recommend the most (or top N) favorite products or information that are most likely to be interested by users.

In the area of research of recommender systems, most of literatures focus on recommendation algorithms and their main aim is to improve the performance of recommender systems. The basic approaches for recommender systems are content-based approach (CB), collaborative filtering approach (CF) and hybrid approach. CB approach has its roots in information retrieval and recommends the right items to users through matching users' profile with items' features [2]. The main weak point of the CB approach is that it just predicts users' preference based on the past history and can't predict users' latent preference. On the other hand, CF approach predicts user's rating for item based on his/her nearest neighbors' rating for that item without knowing items' contents [3] [4]. The CF approach often suffers from the data sparse problem because it just based on the user-item rating matrix which is often very sparse. Both CB and CF methods also suffer from the cold-start problem when a new item or a new user comes to recommender system. Usually, the hybrid approach is used to integrate CB and CF approaches together to solve these problems.

Like CB, CF or hybrid approaches, most recommendation approaches treat users or items as a collection of entities that are similar to each other and use these information to predict the target user's preference. So from this point of view, there are two main entities and three main relationships in recommender systems. The two main entities are respectively users, which indicate people who use the recommender system, and items, which indicate products or information provided for users. The three main relationships mean 1) relationship among users, 2)

Received: 8 December 2011, Accepted: 27 February 2012
Available online 12 March 2012

connection among items, and 3) interaction between users and items caused by the users' preferences. The relationship between different users may be explicit or implicit [5]. The explicit connections are social relationships that indicate friendships, family relationships, colleagues, classmates and so on. The implicit connections are some indirect relationships based on preference or taste, for example, if two users major in the same department then they may be interested in the same books even though they don't know each other. Simmarly, the connections between objects could be the similar features among different items or the similar acceptance by users. For example, items belong to the same categories or are simultaneously liked by most users. And finally, users' preferences for items indicate the interactions between users and items.

Social influence is another key point we adopt in this paper. Social interacitons with another people or other groups affect people's thoughts, feelings, attitudes, or behaviors. Such phenomenon usually happenes in everyday life [6]. Similarly in recommender systems, a user's preference for an item might influence his friends or neighbors' preference for the item. And the *influence power* may depend on the closeness of the two users or some other factor.

Summing up, in this paper we integrate users, items, and the three types of relationships into one model. For the purpose, we use the concepts of Doubly Structural Network (DSN) proposed by our research group [7]. We treat users and items as nodes and relationships as edges. The DSN model consists of three networks: user-network, item-network and cross-network shown in Figure 1. Furthermore, we focus on the active user-item pair's influence on the neighborhood by obtainning the *influence power* and build a dynamic prediction model for learning users' preference based on the DSN model.

The rest of the paper is organized as follows: Section 2 surveys some related researches about recommender sysytems based on network (graph). Section 3 introduces our proposed approaches which is based on the doubly structural network. Section 4 provides experimental results and analyze the sensitivity of the parameters in our model. Finally, Section 5 gives some explanation for our future research.

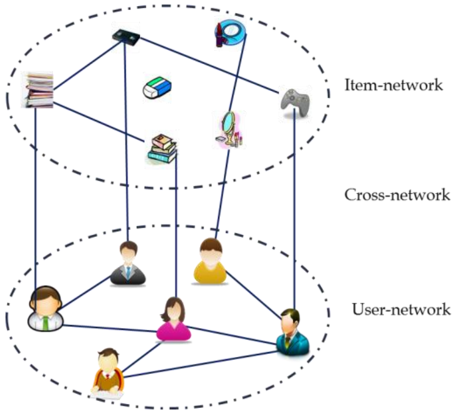

Figure 1 The doubly structural network for recommender systems

2. Related researches

With the widely use of complex network or graph theory, there are a lot of literature about complex network such as small-world networks [8] [9], scale-free networks [10] [11] and random networks [12][13]. In the field of recommender systems, there is a trend adopting complex network or graph in the research of personalization recommendation in recent years.

Papler [14] introduces a recommendation model based on a directed graph of users in which a directed link indicates that the a user's behavior is predictive of the former user's behavior. Recommendations are made by exploring short paths joining multiple users. In [15], they propose a graph-theoretic model for collaborative filtering, in which items and users are both represented as nodes and the edges represent interaction between users and items. Edges in this social network graph are induced by hammock jumps. In [16], they deal with the sparsity problem by applying an associative retrieval framework and related spreading activation algorithms to explore transitive associations among agents through their past transactions and feedback based on the bipartite graphs. In [17], they propose an integrated-graph model for users' interests in personalized recommendation, which is based on Small-World network and Bayesian network. The Integrated-Graph model also consists of two layers. One is user's layer for representing users and the other is merchandise's layer for representing goods or produce.

The relationships among users are described by Small-World network at lower layer. The implications among merchandises are represented by Bayesian network at higher layer. Directed arcs denote the interests and tendency between user's layer and merchandise's layer. Several algorithms for clustering and interest analysis based on Small-World network are introduced.

In addition, we also refer to [5],[18],[19], which are some applications of doubly structural network. The authors proposed a doubly structural network model which is the original type of our DSN model. The DSN Model consists of two levels of networks: one is inner agent-model which represents agents' beliefs or knowledge about the world and the other is inter agent-model which represents a social network among agents. The DSN model can be used to analyze some social phenomenon.

3. Proposed approach

3.1 DSN model for recommender systems

Our DSN model for recommender systems consists of three types of network: user-network, item-network and cross-network. Specifically speaking, we use nodes to denote users or items and edges to denote the relationships among them. The item-network consists of items and connections between them, the user-network consists of users and the relationship between them and the cross-network is a bipartite network which connects user-network and item-network together.

3.1.1 User-Network

User-Network is a social network which represents the relationships between different users [17]. In our research, we define the relationship between users as their implicit relationships which is preference-based and similarity-based. For example, people have the same interests on TV/films watching or people have the work of the same type or they are in the same age and so on. The definition of user - network is as follows:

$$\begin{aligned} &G^U = (U, E^U, W^U), \\ &U = \{1, \cdots, u, \cdots, v, \cdots n\}, \\ &E^U = \{e^U_{uv} \,|\, 1 \le u \le n, 1 \le v \le n\}, \\ &W^U = \{w^U_{uv} \,|\, 1 \le u \le n, 1 \le v \le n\}, \end{aligned} \tag{1}$$

where U is the set of nodes which represent users(ID) and E^U is the set of edges which represent the relationship between users and if there is an edge between user u and v then e^U_{uv} =1 else e^U_{uv}=0 and W^U is the weight of E^U. So user-network is a weighted graph. In this paper we define users' similarity as the basic criteria to measure the relationship between them and furthermore we set a *edge threshold* for user-network's construction as follows:

$$w^U_{uv} = \begin{cases} sim(u,v), & \text{if } sim(u,v) \ge \theta \\ 0 & , \text{ if } sim(u,v) < \theta \end{cases} \tag{2}$$

where $sim(u,v)$ could be the similarity between user u and v based on any similarity computations [2].

3.1.2 Item-Network

Item-Network represents the connection between different items. Items may be in the same category or be liked by users at the same degree. In this paper, we define items connection as the acceptance by users and we use items' similarity to measure the connection between them. The definition of item-network is as follows:

$$\begin{aligned} &G^I = (I, E^I, W^I), \\ &I = \{1, \cdots, i, \cdots j, \cdots m\}, \\ &E^I = \{e^I_{ij} \,|\, 1 \le i \le m, 1 \le j \le m\}, \\ &W^I = \{w^I_{ij} \,|\, 1 \le i \le m, 1 \le j \le m\}, \end{aligned} \tag{3}$$

where I is the set of nodes which represent items(ID) and E^I is the set of edges which represent the connections between items and if there is an edge between item i and j then e^I_{ij}=1 else e^I_{ij} =0 and W^I is the weight of E^I. So item-network is also a weighted graph. And the definition for W^I is as follows:

$$w^I_{ij} = \begin{cases} sim(i,j), & \text{if } sim(i,j) \ge \pi \\ 0 & , \text{ if } sim(i,j) < \pi \end{cases} \tag{4}$$

where $sim(i, j)$ could be the similarity between item i and j based on any similarity computations [2].

3.1.3 Cross-network

In recommender systems, users giving rating to an item represents the degree of their preferences for the item. We define users' preference for items connect the two networks together. We also call the interaction between user-network and item-network cross links/edges and an user-item pair corresponds to a cross link. The definition for cross-network is as follows:

$$\begin{aligned} G^C &= (G^U, G^I, R, \hat{R}), \\ R &= \{r_{ui} \mid u \in U, i \in I\}, \\ \hat{R} &= \{\hat{r}_{ui} \mid u \in U, i \in I\}, \end{aligned} \tag{5}$$

where R is the the set of ratings to items given by users and is observed based on users' activity and means users' real preference. And with respect to $\hat{R}$, we define it as users' predictive preference for items, which is not observed. The aim of recommender systems is to predict users' preference and make it close to users' real preference.

3.2 Learning a dynamic prediction model for recommender systems

In order to build the dynamic prediction model for recommender systems, first, we define the sub-network of DSN as an arbitrary user-item pair's neighborhood which is made up of the user u together with his neighbors and the item i together with its neighbors and the relationship among them (see Figure 2). If two users and two items are neighbors respectively, then we call the cross links between them are neighbor cross link.

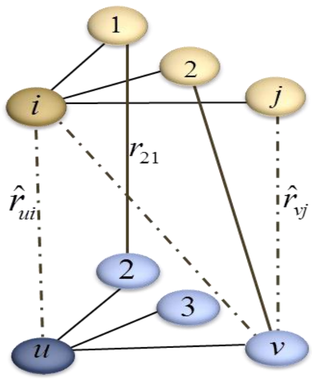

Figure 2 A sub-network of DSN model

Second, we take time step into account in our research for the obtaining of active cross link's influence on the neighborhood or sub-network. Consider a real recommender system, after a user coming to the system, buying or rating an item, another user comes to the system and repeats the same action (in this paper, we don't consider about the simultaneous actions). When a user buys or rates an item, we call the action one time step and the user-item pair as active user-item pair (cross link). Traditional recommendation approaches predict a user' preference just based on the current state of the whole users or items when the active user comes to the system and most recommendation methods do not take into account active user-item's local influence. For example, in a movie recommender system, user *A*'s neighbors are *B, C, D* and movie *a*'s neighbors are *b, c, d, e, f*. We assume user *A* rates movie *a* with rating 5 which means the user likes the movie and we can predict user *B, C, D*'s preference for moive *a, b, c, d, e, f* based on user-item pair *A*-*a*'s influence on them.

In addition, we assume some rules for our research as follows:

i) A cross link is similar to its neighbor cross links;

ii) The more closer users or items' relationship are, then the more similar the corresponding cross links are;

iii) When a user rates an item, which means there is a new solid cross link between the user-network and item-network, the neighbor cross links of the new cross link will be influenced.

In our dynamic prediction model for recommender systems, we defined two important aspects: one is the expected value of users' preference (we use expected preference for short in the following parts) which indicates users' average preference for items and it is relatively stable. The other aspect is the predictive value of users' preference (we use predictive preference for short in the following parts), which is our aim and is based on the expected preference and the local infulence of active cross link.

3.2.1 Expected preference

A user' expected preference indicates the average preference of a user for an item and it is based on the current user-item rating matrix. In a recommender system, there is a rating distribution for each user and each item if the user has given rating and if the item has been rated by users. For example, in GroupLens data set, the overall rating distribution, a random user's rating distribution and a random item's rating distribution are shown in Figure 3, Figure 4 and Figure 5. From the overall rating distribution and a user and an item's rating distribution, we could get the overall average rating, the user's average rating and the item's average rating. We use r_u denotes user u's average rating and r_i denotes item i's average rating given by users and $\overline{r}$ denotes the overall average rating. Based on this, we could get a user's expected preference for an item. For example, if rating scale is from 1 to 5 and $\overline{r}$ =3, r_u=4, r_i= 4, then user u's preference for item i may be expected to be about 4. And if $\overline{r}$ =3, r_u=2, r_i= 2, then user u's preference for item i may be expected to be about 2. We define the expected preference $\overline{r}_{ui}$ of user u for item i as the function of $\overline{r}$, r_u and r_i as follows:

$$\overline{r}_{ui} = f(\overline{r}, r_u, r_i) \tag{6}$$

where *f(x)* could be decided by machine leaning based on the current user-item rating. When we take time step t into account, then $\overline{r}^t_{ui} = f(\overline{r}^t, r^t_u, r^t_i)$.

In this paper, we set *f(x)* is a sigmoid function $f(x) = 1/(1+e^{-x})$ and we use two layer artificial nueral network to learn *f(x)*. The learning task is to find optimal parameters of *f(x)* which make sure the expected preference is close enough to user's rating to item, i.e. to minimize the squared error:

$$\min_f \sum_{u,i \notin T} (r_{ui} - \overline{r}_{ui})^2 \tag{7}$$

where T indicates the training set. There is one thing should be noticed is that we normalize the ratings to 0-1 scale before learning process.

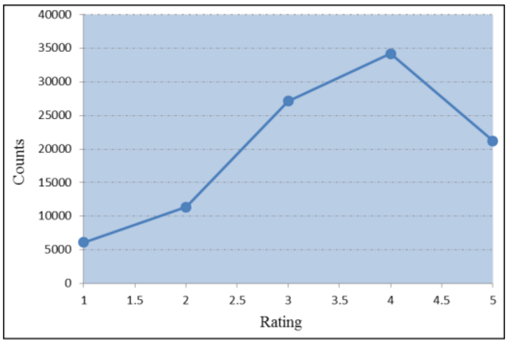

Figure 3 The overall rating distribution

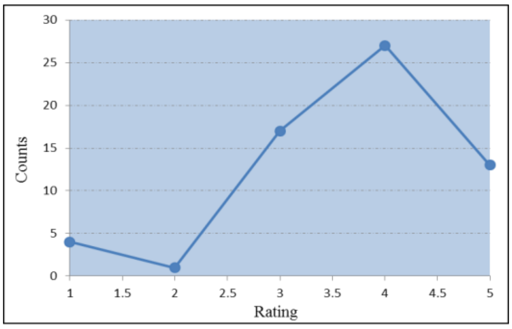

Figure 4 A random user's rating distribution

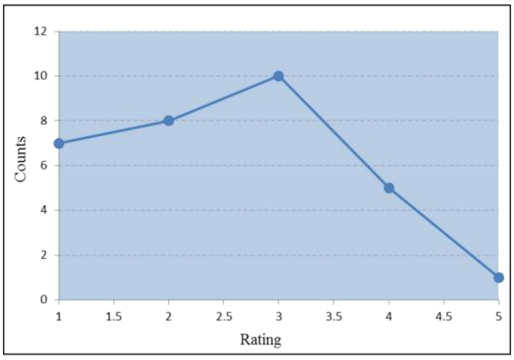

Figure 5 A random item's rating distribution

3.2.2 Predictive preference

As mentioned above, by taking into account time step in recommender systems, we can explore the local influence of active user-item pair and it may influence the whole state of recommender systems. In this paper, we assume local influence is only within the scope of active user-item pair's sub-network. This means that when a user rated an item then there is a cross link between the user and the item and the relationship between the user and his neighbors may change as well as the item and its neighbors and furthermore the cross links inside the sub-network will also change.

The core issue of our dynamic prediction model for recommender systems is: when a user u rates an item i with rating r_{ui} in one time step, how does it influence the corresponding sub-network, especially the neighbor cross links. The scope of the influence include: the relationship between u and his neighors, the connection between item i and its neighbors, and the cross links in the sub-network.

According to the research talked above, the definition for dynamic prediction model for recommender systems based on DSN is as follows (we assume that active user u rates item i with rating r^t_{ui} in time step t+1):

$$\hat{r}^{t+1}_{vj} = \hat{r}^t_{vj} + \eta \cdot \frac{w^{U(t+1)}_{uv} \cdot w^{I(t+1)}_{ij}}{(1+\sigma_1+\sigma_2)} (r^{t+1}_{ui} - \hat{r}^t_{vj}) \tag{8}$$

Where the second term in the right side indicates the active user-item pair's influence power on target user(v)-item(j) pair and η is learning rate and the term $\frac{w^{U(t+1)}_{uv} \cdot w^{I(t+1)}_{ij}}{(1+\sigma_1+\sigma_2)}$ is influencial coefficient and $\sigma_1 = \left| r^{t+1}_{ui} - r^t_{vj} \right|$ denotes the predictive deviation of active user-item pair and target user-item pair and $\sigma_2 = \left|\overline{r}_{ui}^{t+1} - \overline{r}_{vj}^{t+1}\right|$ denotes the expected deviation of them. And furthermore we set two influence threshold $\sigma_1 < \lambda_1$ and $\sigma_2 < \lambda_2$ as constraints for the predicition formula. The constraints here means only the target user-item pair which are close or similar enough to the active user-item pair could be influenced by the latter one. And the term $w_{uv}^{U(t+1)} \cdot w_{ij}^{I(t+1)}$ means that the influence happens only within the scope of active user-item's sub-network. The dynamic model has some adaptive characteristic that adjust users' predictive preference based on the active user's action.

4. Experimental analyses

In this paper, we used the data set from GroupLens to verify our research. The data set has 100000 records (user-item-rating) and contains 943 users and 1682 movies and the rating scale is 1-5. We used 80% of the data set as training data and the rest was test data. And furthermore, we randomly selected 1/8 (10000 records) from the training data and set it as the set of active user-item pair and selected 7/8 (70000 records) as initial data which was used to get the initial DSN model and learned users initial expected preference for items $\overline{r}_{vj}^{0}$. At the beginning *t*=0 , we set $\hat{r}_{vj}^{0} = \overline{r}_{vj}^{0}$. We used *MAE* (Mean Absolute Error) , *Precision*, and *Recall* [20] as the evaluation metrics in our experiment. In this paper, we set user *u* prefers item *i* if $r_{ui}^{t} \geq 3$ which means the user *u* like item *i* and user *u* may prefer item *i* if $\hat{r}_{ui}^{t} \geq 3$ which means we predict user *u* may like item *i* and unlike most the other literatures in recommender systems area, we adopted a none-fixed recommendation list lengths, rather than using a fixed length. And their definition as follows:

$$\mathrm{MAE} = \frac{\sum_{u,i \in testset} \left| r_{ui} - \hat{r}_{ui} \right|}{|testset|}$$

$$\mathrm{Precision} = \frac{\{\text{Items user prefers}\} \cap \{\text{Items user may prefer}\}}{\{\text{Items user may prefer}\}}$$

$$\mathrm{Recall} = \frac{\{\text{Items user prefers}\} \cap \{\text{Items user may prefer}\}}{\{\text{Items user prefers}\}}$$

We firstly studied the sensitivity of parameters in our model. The *MAE* at different learning rate as Figure 6 shows. From the *MAE* trend line at different learning rate, we can see that the value of *MAE* keeps on decreasing, which means that the prediction of our dynamic model is more close to users' real preference for items. And we notice that the value of *MAE* decreases very quickly at the beginning and then the change is slower. And from the performance at different learning rate, we can see that the decline rate and convergence rate of *MAE* are quicker with the bigger learning rate.

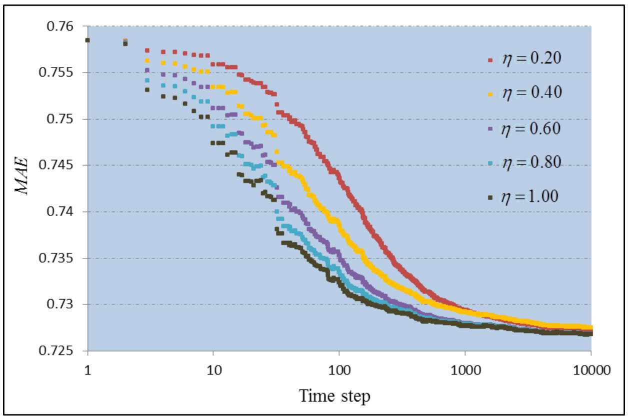

Figure 6 The *MAE* over time step at different learning rate (when $\theta = 0.98, \pi = 0.98, \lambda_1 = 0.6, \lambda_2 = 0.6$)

About the sensitivity of edge threshold for user-network and item-network is as Figure 7 shows. The figure indicates that at the beginning (about *t*<1000) with smaller edge threshold, which means the scope of active user-item pair is bigger, the *MAE* declines quickly and when about *t*>1000 with bigger edge threshold, which means the scope of active user-item pair is smaller, the *MAE* declines quickly.

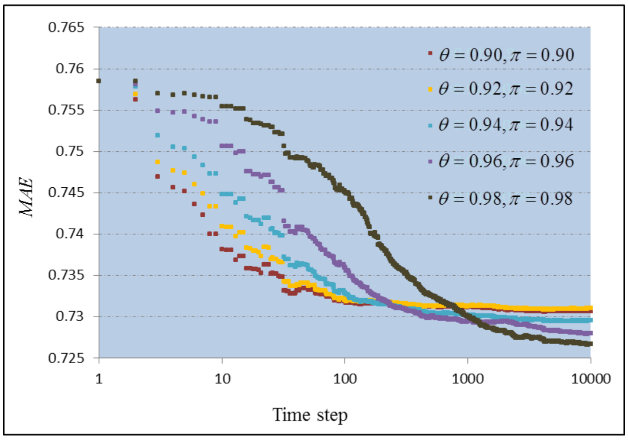

Figure 7 The *MAE* at different edge threshold (when $\eta = 1.0, \lambda_1 = 0.60, \lambda_2 = 0.60$)

We also studied the sensitivity of influence threshold for recommender systems' performance as Figure 8 shows. The results tell us that with bigger or smaller influence threshold the performance is not as good as a moderate value of that. This indicates if influence threshold is smaller then the scope of active user-item pair's influence is smaller and only a few target user-item pair could be influenced. On the other hand, if influence threshold is bigger then the scope of active user-item pair's influence is bigger and this will lead to worse performace such as $\lambda_1 = 1.00, \lambda_2 = 1.00$.

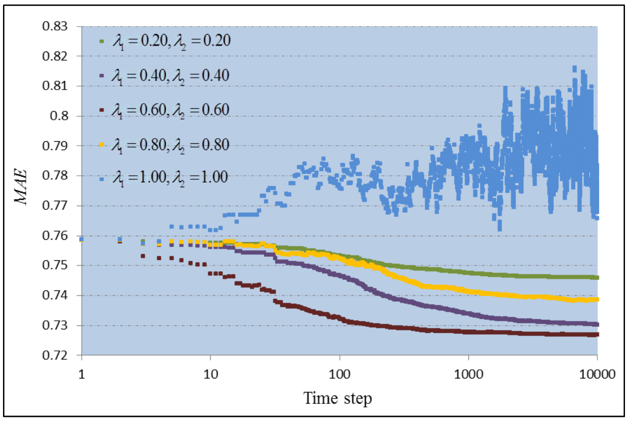

Figure 8 The *MAE* at different influence threshold

(when $\eta = 1.0, \theta = 0.98, \pi = 0.98$)

In addition, we compared our approach to traditional collaborative filtering algorithm in terms of *MAE*, *Precision*, *Recall* and *F-Measure* as table 1 shows. The results were computed by 5-flod cross validation and we used the value of *MAE* at *t*=10000 for the proposed approach. And table 1 gives the best results of the three approaches. The results shows that our proposed approach outerperformed user-based CF and item-based CF except in terms of *Precision*. And in terms of *MAE* and *Recall*, the proposed approach has a obvious advantage than the conventional CF approaches. This indicates that it may give accurate and comprehensive predicition for users' preference by using our method.

Table 1 The comparision of proposed approach with CF

Algorithms	Precision	Recall	MAE
Proposed Approach	1.93%	**95.5%**	**0.730**
User-based CF	**2.42%**	89.67%	0.763
Item-based CF	1.67%	86.80%	0.764

5 Conclusions

In this paper, we proposed a novel approach for recommender systems, which incorporates users, items and the relationships between them into one framework based on doubly structural network, and built a dynamic prediction model by focusing on active user-item pair's influence on neighborhood for predicting users' preference over time. From the experiment we can see that the novel method could give a good performance for recommender systems. In the future we'll keep on studying the proposed dynamic prediction model and improve the performance of the model . And furthermore, we will use the other data sets to verify it.

References

[1] A. Ansari, S. Essegaier, R. Kohli, Internet Recommendations Systems, Journal of Marketing Research, Vol.37,No.3, 2000,pp. 363-375.

[2] P. Lops, M. Gemmis, G. Semeraro, Content-based Recommender Systems: State of the Art and Trends, Recommender Systems Handbook, Springer Science and Business Media. 2011.

[3] B. Sarwar, Kar Ypis G. , J. Konstan, et al., Item-based collaborative filtering recommendation algorithms, Proceedings of the 10th International World Wide Web Conference, New York , N. Y. , USA: ACM , 2001.

[4] G. Adomavicius, A. Tuzhili, Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, IEEE Transaction on Knowledge and Data Engineering, Vol.17, No.6, 2005, pp. 734-749.

[5] D. Frey, A. Jegou, A. M. Kermarrec. Social Market: Combining Explicit and Implicit Social Networks, International Symposium on Stabilization, Safety, and Security of Distributed Systems, 2011.

[6] K. Y. Liu, M. King, P. S. Bearman, Social influence and the Autism Epidemeic, American Journal of Sociology, Vol.115, No.5, 2010, pp. 1387–1434.

[7] M. Kunigami, M. Kobayashi, et al., On the emergence of money through a complex doubly structural network model, The 22nd Annual Conference of the Japanese Society for Artificial Intelligence, 2008.

[8] A. Barrat, M. Weigt, On the properties of small-world network models, The European Physical Journal B 13, 2000, pp. 547-560.

[9] J. D. Watts, H. S. Strogatz, Collective dynamics of 'small-world' networks, Nature, Vol.393, 1998, pp. 440-442.

[10] S. Boccalettia, V. Latora. Complex networks: Structure and dynamics, Physics Reports 424, 2006, pp. 175 – 308.

[11] R. Albert, A,L. Barabasi, Statistical mechanics of complex networks, Reviews of Modern Physics, Vol.74 ,2002, pp. 47-97.

[12] A.L. Barabasi, Scale-Free Networks: Adecade and beyond, Science, Vol.325, 2009.

[13] K. Christensen, P. Donangelo, B. Koiller, K. Sneppen, Evolution of Random Networks, Physcial Review Letters, Vol.81, No. 11, 1998, pp. 2380-2383.

[14] C. Aggarwal, Horting hatches an egg: A new grph-theoretic approach to collaborative filtering, Proceedings of the 5th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'99) (San Diego, Calif.), ACM, New York, 1999, pp. 201–212.

[15] B. J. Mirza, , B. J. Keller, and N. Ramkrishnana. 2003. Studying recommendation algorithms by graph analysis. J. Intel. Inf. Syst., Vol.20, No.2, 2003, pp. 131–160.

[16] Z. Hhuang, H. Chen, D. Zeng, Applying Associative Retrieval Technique to Alleviate the Sparsity Problem in Collaborative Filtering. ACM Transactions on Information Systems, Vol. 22, No. 1, 2004, pp. 116–142.

[17] S. Zhang, D. Chen, Analysis of user interests based on integrated-graph model, Sixth international conference on fuzzy systems and knowledge discovery, Vol.2, 2009, pp. 203-208.

[18] M. Kunigami, M. Kobayashi, S. Yamadera, T. Terano, On emergence of money in self-organizing micro-macro network model. Proceedings of ESSA, 2007, pp. 417–425.

[19] M. Kunigami, M. Kobayashi, et al. A Doubly Structural Network Model and Analysis on the Emergence of Money, Agent-Based Social Systems, 2010, Vol.7, Part II, pp. 137-149.

[20] J. L. Herlocker, J. A. Konstan, L. G. Terveen, J. T. Riedl, Evaluating Collaborative Filtering Recommender systems, Vol.22, No.1,2004, pp. 5-53.