International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.9 (2016), pp.277-288
http://dx.doi.org/10.14257/ijmue.2016.11.9.28

Combining Neighborhood Based Collaborative Filtering with Tag Information for Personalized Recommendation

Xiaoyi Deng[1,2]

[1]*College of Business Administration, Huaqiao University, Quanzhou, 362021, China*
[2]*Research Center for Applied Statistics and Big Data, Huaqiao University, Xiamen, 361021, China*
londonbell.deng@gmail.com

Abstract

Collaborative filtering recommendation is one of the most effective recommending techniques, which provide customers with suggestions according to their interests. However, neighborhood based collaborative filtering methods confront great challenges of data sparsity and lack of accessorial information in the context of big data. To address these problems, we propose a hybrid model combining tag information and neighborhood based collaborative filtering. A folksonomy network model based on tag information is proposed to analyze the tag relevance between different items. And tag relevance is incorporated into rating prediction of neighborhood based collaborative filtering for improving the recommendation accuracy. Experiments on MovieLens and Netflix datasets are carried out to evaluate the performance of our method. The results show that our method outperforms other methods and can improve recommending quality effectively.

Keywords: *Collaborative filtering; Neighborhood based model; Tag; Personalized recommendation*

1. Introduction

For decades, the electronic commerce (e-commerce) has been regarded as an essential way of conducting business. E-commerce has made a great quantity of new products, services and related commerce more available. E-commerce also brought an increasing number of customers, products and huge amount of relevant information. These issues lead to "information overload", which makes it very difficult to provide customers with recommendations via analyzing their interests. The need for solving the information overload problem has led to the prevalence of personalized recommendation [1]. Recommender systems (RSs) are the most successful application of personalized recommendation, which receive information from users about items that they are interested in, and then recommend to them items that may fit their needs [2].

The core of recommender systems relies on well-known algorithms, collaborative filtering (CF) [3], where there are two primary approaches, neighborhood based model (NBM) and latent factor model (LFM). NBM is dependent on the availability of user ratings information, and product recommendations to a target user based on the relationship between their active neighbors, without relying on any information about the items themselves other than their ratings [4]. By contrast, LFM transforms both items and users into the same latent factor space, and then characterizes each entity with a feature vector inferred from the existing ratings. In LFM, the predictions ratings are denoted by the inner product of the corresponding vector pairs. Therefore, NBM based CF has an advantage in situations where it is hard to analyze the different aspects of the data, such as music, videos and other digital products or services. Therefore, NBM based CF has been

ISSN: 1975-0080 IJMUE

developed over decades and widely applied in many recommender systems and Internet-related fields, such as Amazon, Netflix, and Yahoo.

Despite its advances, NBM based CF suffers from several problems, such as data sparsity and cold start problem. Data sparsity is common for the user-item ratings matrix to be extremely sparse. It makes NBM based CF difficult to identify similar users and items, and produce accurate predictions or recommendations due to the lack of ratings. To solve the problem of data sparsity, many different dimensionality reduction approaches have been proposed, such as singular value decomposition (SVD) and principle component analysis (PCA) [4]. However, useful information for recommendations related to those approaches may get lost and recommendation quality may be degraded, when certain users or items are discarded [5].

The data sparsity appears when a new user or item has just entered the system. This problem can be divided into cold-start items and cold-start users. A cold-start user describes a new user that joins a CF-based recommender system and has presented few opinions. With this situation, the system is generally unable to make high quality recommendations [6]. New items cannot be recommended until someone rates it. Although attribute-aware method [5] takes into account item attributes, which are defined by domain experts. It is limited to the attribute vocabulary, and items' global attributes are essentially not helpful to generate personalized recommendations.

To address these problems, a hybrid recommendation model integrating tag information with NBM is proposed in this paper. Tags allow users to comment content with descriptive keywords, and tags usually imply users' preferences and opinions about items. The proposed approach first determines similarities between items tags and subsequently identifies the tags relevance for each item. Furthermore, the tags relevance is integrated into the similarity calculation and rating prediction for the advance of the recommendation quality.

2. Background Review

2.1. Collaborative Filtering

CF generates recommendations based on the data that store how users rated items [3]. To provide recommendations, CF tries first to search for users who have rated the same or similar items. Once the users with common tastes are found, CF will recommend the items highly rated by those users. Generally, the more items that users have rated, the more similar the users are. The procedure of CF can be stated as follows.

It is assumed that $U=\{u_i|i=1,2,\ldots,m\}$ is a set of m users and $I=\{I_j|j=1,2,\ldots,n\}$ is a set of n distinct items. The set of user ratings is denoted by $R=\{(u_i, I_j)|u_i \in U, I_j \in I\}$ which is a $m\times n$ matrix, as shown in equation (1).

$$R=\left(r_{u_i,I_j}\right)_{m\times n},\quad r_{u_i,I_j}=\begin{cases} S & \text{if } u_i \text{ rated } I_j \\ \varnothing & \text{if } u_i \text{ not rated } I_j \end{cases} \tag{1}$$

where $r_{u,I}$ is the rating of the item I by user u, which indicates the user's preference for different items. Usually, $r_{u,I}$ is equal to a real number denoted by S ($S\neq\varnothing$). When $r=\varnothing$, it means that user u_i does not rate a certain item I_j.

After the data preparation, CF needs to select a similarity function to measure how similar two users are. Two of the most well-known similarity measures are Cosine-based similarity and Pearson correlation coefficient [1] defined in equations (2) and (3).

$$Sim_{Cosine}\left(I_i,I_j\right)=\frac{\sum_{i\in I(u_i,u_j)} r_{u_i,I}\cdot r_{u_j,I}}{\sqrt{\sum_{i\in I(u_i,u_j)} r_{u_i,I}^{2}}\cdot\sqrt{\sum_{i\in I(u_i,u_j)} r_{u_j,I}^{2}}} \tag{2}$$

$$Sim_{PCC}\left(I_i, I_j\right) = \frac{\sum_{i \in I(u_i, u_j)} (r_{u_i,I} - \overline{r}_{u_i})(r_{u_j,I} - \overline{r}_{u_j})}{\sqrt{\sum_{i \in I(u_i, u_j)} (r_{u_i,I} - \overline{r}_{u_i})^2} \sqrt{\sum_{i \in I(u_i, u_j)} (r_{u_j,I} - \overline{r}_{u_j})^2}} \quad (3)$$

where $r_{u,I}$ is the rating of item I by user u; $\overline{r}_u$ is mean rating of user u, and $I(u_i, u_j)$ represents the items co-rated by users u_i and u_j

Once similarity calculation has been done, prediction of a rating of an item I_j by user u_i can be obtained for conventional CF methods [5], as shown in the following equation (4).

$$PR\left(u_i, I\right) = \overline{r}_{u_i} + b_u + b_i \quad (4)$$

2.2. Related Works

The traditional CF approaches predict the rating of items for target users only based on the user-item rating matrix. Although CF is a very successful recommending technology, there are still some potential problems:

(1) With the rapid development of mobile commerce and e-commerce, the magnitudes of users, commodities and services grow rapidly, while users' rating information is of insufficiency. This resulted in extreme sparsity of user rating data [4].

(2) Cold start problem usually causes bad performance on new users and new items because there is few or no rating for them. It is significant and challenging for existing methods to deal with the increment of new users and new items.

To solve the sparsity problem, Deng [8] proposed a CF algorithm based on item rating prediction (ErrR-CF), which takes use of a new similarity measure to find target users' neighbors for better prediction results. Lee et al [9] presented a CF recommendation methodology based on both implicit ratings and less ambitious ordinal scales to enhance the quality of collaborative recommendation. Anand and Bharadwaj [10] proposed various sparsity measure schemes based on local and global similarities for achieving quality predictions. Many other researchers employed different dimensionality reduction approaches to produce accurate predictions, such as SVD, PCA and LDA. However, many useful information for recommendations related to those approaches may get lost and recommendation quality can be degraded, when certain users or items are discarded

Due to the cold start problem, Leung et al [4] utilized association rules to integrate domain items information into traditional CF, and introduced a preference model to comprise user-item relationships and item-item relationships. Ahn [11] applied a heuristic similarity measure method that focuses on improving the recommendation performance under the cold-start conditions. Kim et al [12] designed an error-reflected model derived from explicit ratings for the accurate predictions. Shinde and Kulkarni [13] introduced a novel centering-bunching-based clustering algorithm (CBBC) to overcome information overload for a better rating prediction. Deng and Jin [14] employ user access sequence for similarity measurement to search target users nearest neighborhoods and reduce the impact of cold start problem on prediction quality

These previous researches have made several improvements on traditional CF algorithms, and they partially reduced the effect of data sparsity on the rating prediction. These previous researches have made several improvements on conventional CF algorithms, and they partially reduced the effect of data sparsity on the rating prediction. However, it is assumed in most existing CF approaches that all items have the same weight to rating data when measuring similarity, and items attributes are essentially not helpful to generate similarity [15]. This results in a lower accuracy of prediction results, so the quality recommendation is reduced

3. Hybrid NBM Model Based on Tag

The proposed hybrid recommendation model is composed of two phases: the first phase constructs folksonomy network to analyze the tag relevance between items by using tag information; the second phase incorporates folksonomy information into conventional similarity calculation for enhancing the prediction. At last, the recommendations are made by computing the weighted average of the rating of items.

3.1. Folksonomy Network Model

Tagging technique is popularized by websites associated with Web 2.0. Tagging is the only feasible way to organize multimedia data structured and to make it searchable. And tags can be freely chosen by a user and are not restricted to any taxonomy [16].

With the advent of tagging technique, users are enabled to share opinions on various types of internet resources using arbitrary tags according to their tastes [17]. Those tags created by users can represent item relevance and user preference, which could be capable to enhance the recommendation quality [18]. In consequence, a folksonomy network model (FNM) based on tag information is constructed to analyze the item relevance, and then FNM is integrated into the NBM-based CF model for more accurate rating prediction.

In the FNM, three kinds of item relevance described by tags are defined: strong link, medium link and weak link.

(1) Strong link: if two items are assigned same/similar tags by the same user, the corresponding tag link is a strong link.

(2) Medium link: if two items are assigned same/similar tags by different users, the corresponding tag link is a medium link.

(3) Weak link: if two items are assigned dissimilar tags by the same user, the corresponding tag link is a weak link.

To illustrate a simple example of three types of tag links, two users' tag assignments on five items are shown in Figure 1.

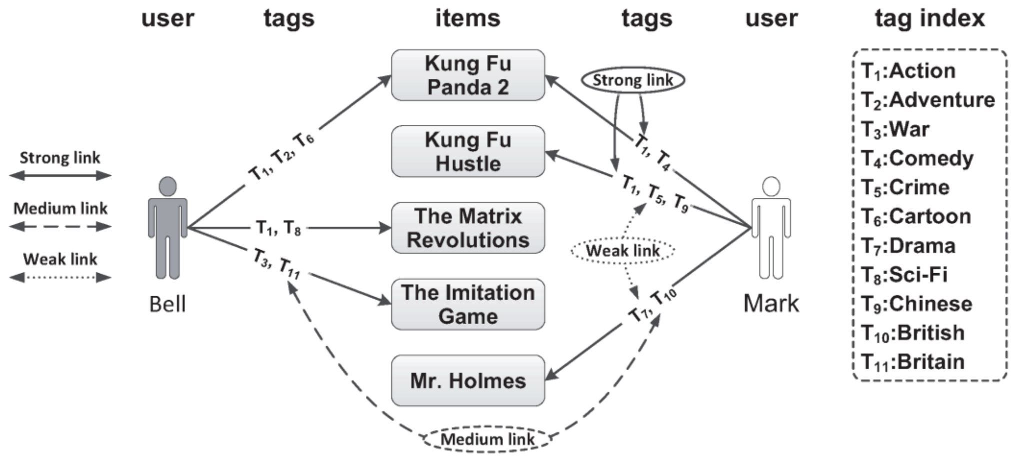

Figure 1. An Example of Tag Links

In Figure 1, there are weak tag links between each pair of items inside items tagged by user Mark, such as T_4(Comedy) & T_5(Crime), T_4(Comedy) & T_7(Drama) and T_9(Crime) & T_5(Drama). The item "The Imitation Game" is tagged by user Bell with the 'Britain', and the item "Mr. Holmes" is tagged by user Mark with 'British', these two words are very similar, so there is a medium tagging link between "The Imitation Game" and "Mr. Holmes". The items "Kung Fu Panda 2" and "Kung Fu Hustle" are both tagged with 'Action' by user Mark, so there is a strong link between them.

After the definition of tag links, a weight measurement should be selected to describe the importance of each tag link. In this paper, the weight on a tag link is considered from two aspects: the tag similarity and the tag link category. For tag similarity, because the formats of tags in folksonomy are usually arbitrary, each pair of tags can be simply regarded as word sets. And tag similarity (*TS*) between tagged item I_i and I_j can be calculated by using the Jaccard similarity, as shown in in equation (5).

$$TS\left(I_i, I_j\right) = Sim_{JS}\left(I_m, I_n\right) = \left|T_m \cap T_n\right| / \left|T_m \cup T_n\right| \tag{5}$$

where T_m and T_n denote the tags belonging to item I_i and I_j, respectively.

Then, the occurrence probability of three tagging link categories is considered as an adjusting coefficient, which is introduced into the weight calculation. Therefore, the occurrence probability of strong link, medium link and weak link are denoted by P_s, P_m and P_w, respectively. So the tag relevance (*TR*) between tagged items *i* and *j* can be computed in equation (6).

$$\begin{gathered} TR\left(I_i, I_j\right) = P_s^{-1} \cdot \sum TS_s\left(I_i, I_j\right) + P_m^{-1} \cdot \sum TS_m\left(I_i, I_j\right) + P_w^{-1} \cdot \varepsilon \\ P_s = N_s / N,\ P_m = N_m / N,\ P_w = N_w / N \end{gathered} \tag{6}$$

where TS_s denotes the tag similarity of tags pair with strong link, TS_m denotes the tag similarity of tag pair with medium link. Because there is no comparison between a tag pair with weak links, a constant ε with the minimal value of the tag similarities, is applied to restrict the importance of the weak links; N_s, N_m and N_w respectively denote the count numbers of the strong, medium and weak links in folksonomy data, and $N = N_s+N_m+N_w$.

Then, the folksonomy network model (FNM) is defined based on tag relevance. A folksonomy network (FN) is an undirected weighted graph. Each node in FN denotes a specified item and the weight on each edge is the tag relevance between the corresponding two items. For instance, a simple FNM is constructed based on the folksonomy data between user Bell and user Mark shown in Figure 1, as shown in Figure 2.

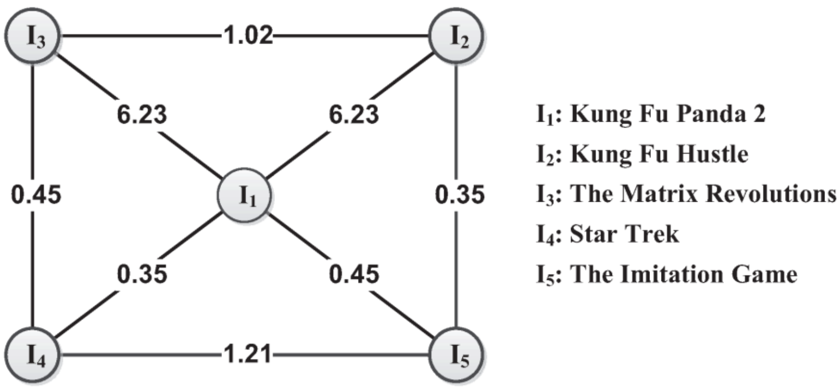

Figure 2. An Example of FNM

Apparently, the FNM in Figure 2 can also be depicted as an adjacency matrix denoted by *TR*, as shown in Table 1.

Table 1. Tag Relevance in the Example FNM

	I_1	I_2	I_3	I_4	I_5
I_1	---	6.23	6.23	0.35	0.45
I_2	6.23	---	1.02	---	0.35
I_3	6.23	1.02	---	0.45	---
I_4	0.35	---	0.45	---	1.21
I_5	0.45	0.35	---	1.21	---

3.2. Model Integration

After the constriction of FNM, the folksonomy information F is integrated into the rating prediction process proposed by Koren [19], as shown in equation (7).

$$PR(u_i, I) = \overline{r}_{u_i} + b_u + b_i + \frac{\sum_{j \in R(u)} \omega_{ij}\left(r_{u_j} - b_{u_j}\right)}{\left|R(u)\right|^{\alpha}} + \frac{\sum_{k \in N(u)} c_{ik}\left(r_{u_j} - b_{u_j}\right)}{\left|N(u)\right|^{\alpha}} + F \tag{7}$$

where $\overline{r}_{u_i}$ is the overall average rating; b_u and b_i indicate the observed deviations of user u and item i, respectively; ω_{ij} and c_{ik} denote the relevance weight of items and the implicit user preference bias, respectively; $R(u)$ is the item set which contains ratings by u, and $N(u)$ contains all items with implicit rating provided by u; α is a constant with an usual set at 0.5, which controls the extent of normalization parameter controlling the extent of normalization.

In equation (7), the item-oriented rating interactions can be marked by E, as shown in equation (8). And the rating prediction equation can be reduced to another equation as shown in equation (9).

$$E = \frac{\sum_{j \in R(u)} \omega_{ij}\left(r_{u_j} - b_{u_j}\right)}{\left|R(u)\right|^{\alpha}} + \frac{\sum_{k \in N(u)} c_{ik}\left(r_{u_j} - b_{u_j}\right)}{\left|N(u)\right|^{\alpha}} \tag{8}$$

$$PR(u_i, I) = \overline{r}_{u_i} + b_u + b_i + E + F \tag{9}$$

Because the rating data and the folksonomy data are parallel data sources, there must be a coefficient to balance the importance of E and F. In this paper, a coefficient β is introduced in equation (9), as shown in equation (10)

$$PR(u_i, I) = \overline{r}_{u_i} + b_u + b_i + \beta E + (1-\beta)F \tag{10}$$

With consideration on the FNM, three factors are utilized in the F calculation process for prediction:(1) the item set tagged by user u, $T_p(u)$, which indicates the active user's tagging preference; (2) the item set having tag relevance with item i, $T_n(i)$, which contains the items connected to the tagged item;(3) the tag relevance $TR(I_i,I_j)$ between item i and item j. Therefore, the folksonomy information F can be computed in following equation (11).

$$F = \frac{\sum_{j \in T_n(I)} TR\left(I_i, I_j\right)\left(r_{u_j} - b_{u_j}\right)}{\left|T_n(I)\right|^{\alpha}} + \frac{\sum_{k \in T_p(u)} c_{ik}\left(r_{u_j} - b_{u_j}\right)}{\left|T_p(u)\right|^{\alpha}} \tag{11}$$

Thus, equation (9) can be transformed to another equation as shown in equation (12).

$$PR\left(u_i, I\right) = \overline{r}_{u_i} + b_u + b_i + \beta E + \left(1-\beta\right) F$$
$$E = \frac{\sum_{j \in R(u)} \omega_{ij} \left(r_{u_j} - b_{u_j}\right)}{\left|R\left(u\right)\right|^{\alpha}} + \frac{\sum_{k \in N(u)} c_{ik} \left(r_{u_j} - b_{u_j}\right)}{\left|N\left(u\right)\right|^{\alpha}}$$
$$F = \frac{\sum_{j \in T_n(I)} TR\left(I_i, I_j\right)\left(r_{u_j} - b_{u_j}\right)}{\left|T_n\left(I\right)\right|^{\alpha}} + \frac{\sum_{k \in T_p(u)} c_{ik} \left(r_{u_j} - b_{u_j}\right)}{\left|T_p\left(u\right)\right|^{\alpha}} \tag{12}$$

4. Experiments and Results

In this section, a numerical experiment is designed to test and evaluate T-NBM. The experiment on three real-world datasets is carried out on a computer with Intel E3-1230 3.2GHz CPU, 16GB RAM and Windows 2003 operation system. And the other three CF algorithms are used as the benchmarks in this experiment.

4.1. Experiment Design

All the experiments are carried out on two real world datasets for completeness and generalization of results, as shown in Table.2. These three datasets are publicly open for research purpose and provided by GroupLens Research Group at University of Minnesota. The sizes of the three datasets are given in Table 2. MovieLens datasets provide ratings on movies in the scale of 1 to 5, and the tags are labeled arbitrarily by users.

For all the experiments, all datasets are randomly divided into two groups: 80% of the data is used as a training set and 20% of the data is used as a test set. In the other word, 80% of the users are utilized as the reference for similarity calculation, and actual recommendation is conducted to 20%; similarly, 80% of the movies are used for similarity calculation, while 20% are actually recommended to users.

Table 2. Characteristics of Two Movielens Datasets

dataset	*user*	*movie*	*rating*	*tag*	*sparsity*
MovieLens-10M	71567	10681	10M	95580	1.31%
MovieLens-20M	138493	27278	20M	465564	0.53%

In order to evaluate the performance of our approach, the following metrics are selected: Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

(1) MAE is the most widely used metric for measuring the deviation of predictions generated by the recommender system from the user rating. The lower MAE is, the better prediction performance is. It is defined in equation (13).

$$MAE = \frac{\sum_{i=1}^{N} \left|P_i - Q_i\right|}{N} \tag{13}$$

where P_i is the rating prediction, Q_i is corresponding real rating and N is the number of user rating in rating matrix.

(2) RMSE is a statistical accuracy metric representing the accuracy of predicted rating, an important metric for customers (The lower the better). RMSE is defined in equation (14).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\left(P_i - Q_i\right)^2}{N}} \tag{14}$$

where P_i is the rating prediction, Q_i is corresponding real rating and N is the number of user rating in rating matrix.

To compare the performance of our algorithm, three other typical CF algorithms are implemented: an item-based CF algorithm proposed by Sarwar and et al (denoted by KNN-CF) [2], an item-based CF approach based on item rating prediction (ErrR-CF) [8], and a tag-based CF method named L2R-CF[20]. KNN-CF applies Cosine-based similarity to predict rating; ErrR-CF employs Cosine-based similarity to perform rating prediction; and L2R-CF utilizes rating deviation to predict user rating. Our proposed T-NBM is evaluated by comparing with the three benchmark algorithms with the parameters α and β both set at 0.5.

4.2. Experimental Results

The experimental results from four algorithms on MovieLens-10M and MovieLens-20M are respectively shown in Figures 3~6.

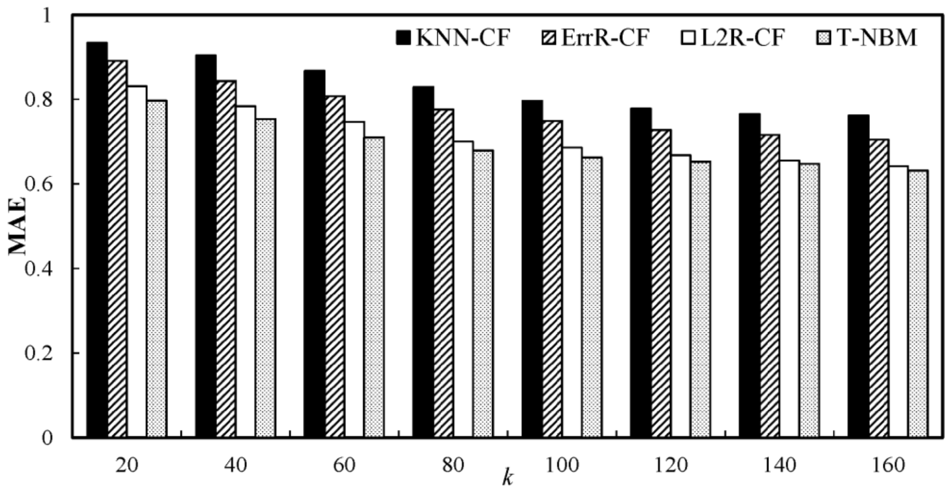

Figure 3. Comparisons of Four Algorithms' Maes on Movielens-10M

In Figure 3, the minimal MAE of T-NBM on MovieLens-20M is 0.7981, and the MAE values of KNN-CF, ErrR-CF and L2R-CF are 0.7619, 0.705, 0.6419 and 0.6318, respectively. The value of RMSA of T-NBM is only 82.92%, 89.61% and 98.43% of that of KNN-CF, ErrR-CF and L2R-CF, respectively. We can find that T-NBM outperforms KNN-CF, ErrR-CF and L2R-CF. The result means that T-NBM has the minimum MAE on MovieLens-10M.

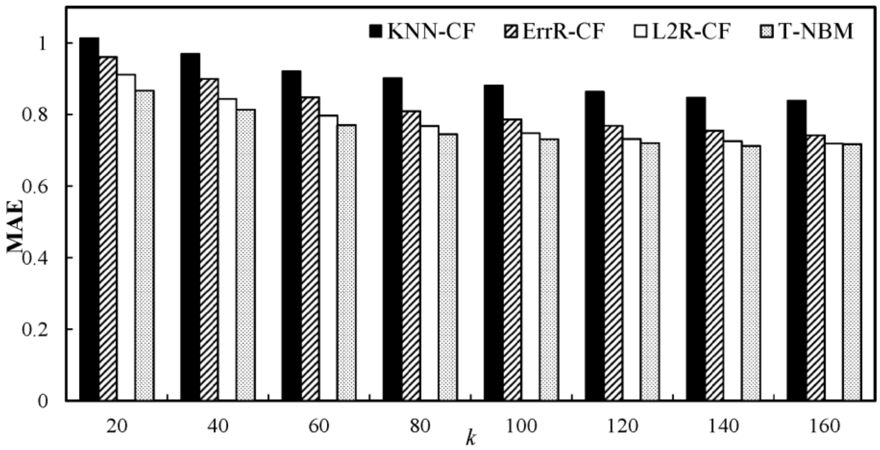

Figure 4. Comparisons of Four Algorithms' Maes on Movielens-20M

In Figure 4, the T-NBM's minimal MAE value on MovieLens-20M is 0.7981, and the MAE values of KNN-CF, ErrR-CF and L2R-CF are 0.839, 0.7419, 0.7189 and 0.717, respectively. Similar to results in Figure4, The value of RMSA of T-NBM is about 85.46%, 96.64% and 99.73% of that of KNN-CF, ErrR-CF and L2R-CF, respectively. We can find that T-NBM outperforms KNN-CF, ErrR-CF and L2R-CF. The result means that T-NBM has the minimum MAE on MovieLens-10M.

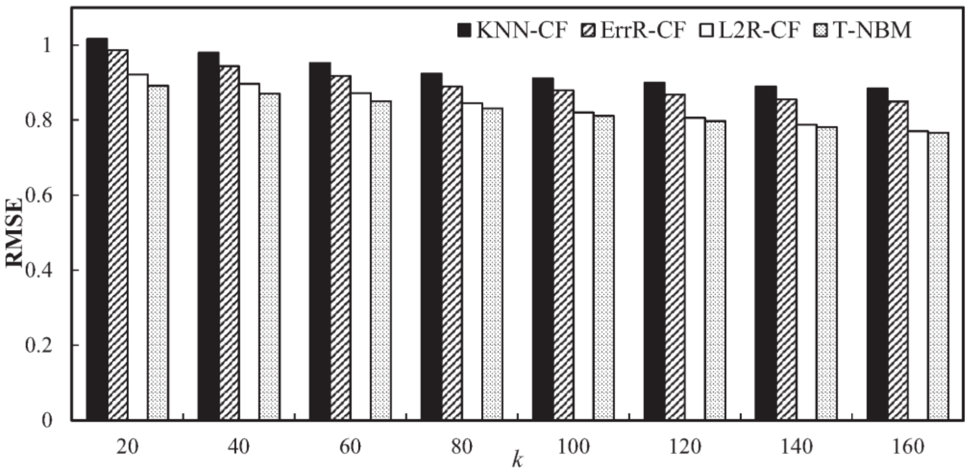

Figure 5. Comparisons of Four Algorithms' RMSEs on MovieLens-10M

In Figure 5, RMSE values of four algorithms under MovieLens-10M are shown. The minimal RMSE of four algorithms are 0.8847, 0.8499, 0.7705 and 0.7662, respectively. L2R-CF has lower RMSE values than KNN-CF and ErrR-CF, which benefits from tag information in MovieLens-10M. The minimal RMSE of T-NBM is 0.1185, 0.0837 and 0.0043 lower than that of KNN-CF, ErrR-CF and L2R-CF, respectively. It is clear that T-NBM possesses the minimum values of RMSE on MovieLens-10M.

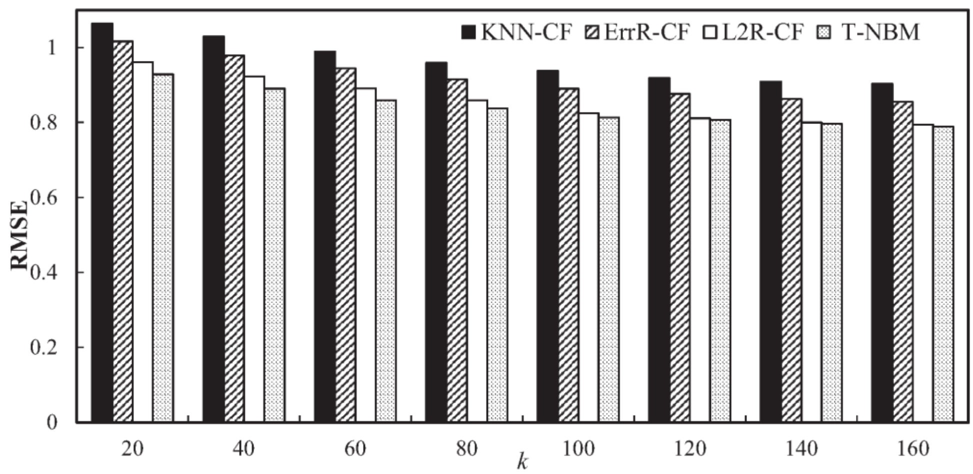

Figure 6. Comparisons of Four Algorithms' Rmses on Movielens-20M

In Figure 6, the RMSE of four algorithms on MovieLens-10M are presented. The minimal RMSE of four algorithms are 0.9031, 0.8547, 0.7945 and 0.7892, respectively. Similar to results in Figure5, T-NBM has lowest RMSE values than that of KNN-CF, ErrR-CF and L2R-CF, which benefits from tag information in MovieLens-10M. The minimal RMSE of T-NBM is 0.1139, 0.0655 and 0.0053 lower than that of KNN-CF, ErrR-CF and L2R-CF, respectively. It is clear that T-NBM possesses the minimum values of RMSE on MovieLens-20M.

From Figures 3 to 6, it is clear that the proposed T-NBM has the minimum values of MAE and RMSE on two different datasets compared with other three CF methods. Therefore, our T-NBM outperforms the other three typical CF approaches, and it can effectively improve the quality of collaborative recommendation.

5. Conclusions

This paper presents an improved collaborative filtering method T-NBM to enhance the prediction quality of collaborative recommendation. T-NBM employs tag information to build a FNM obtain tag relevance between item pairs, and integrates tag relevance with NBM based collaborative filtering for improving the recommendation accuracy. The experimental results have shown T-NBM succeeds in advancing the quality of rating prediction. Compared with other three algorithms, T-NBM has both the minimum RMSE and the minimum RSME. This means that T-NBM outperforms the other three typical CF approaches in terms of quality. This indicates that T-NBM is more applicable in situations where context and relationship are critical to the success of the application, just like in e-commerce.

Our future research will focus on the trust propagation in social networks, and we will try to make our computations more extensible and faster, for example, by developing paralleled algorithms.

Acknowledgements

The author gratefully acknowledges the financial support from the National Natural Science Foundation of China (No.71401058 and No.61300139).

References

[1] X. Y. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques", Advances in Artificial Intelligence, **(2009)**, pp. 1-19.
[2] G. Linden, B. Smith and J. York, "Amazon.com recommendations: item-to-item collaborative filtering", IEEE Internet Computer, vol. 7, no. 1, **(2003)**, pp. 76-80.
[3] B. Sarwar, G. Karypis, J. Konstan and J Riedl, "Item based collaborative filtering recommendation algorithms", Proceedings of the 10th international conference on World Wide Web, New York, USA. **(2001)**, pp. 285-295.
[4] C. W. Leung, S. C. Chan and F. Chung, "An empirical study of a cross-level association rule mining approach to cold-start recommendations", Knowledge Based System, vol. 21, no, 7, **(2008)**, pp. 515-529.
[5] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions", IEEE T. Knowledge Data En., vol. 17, no. 6, **(2005)**, pp. 734-749.
[6] P. Massa and P. Avesani, "Trust-aware collaborative filtering for recommender systems", Proceedings of International Conference on Cooperative Information Systems, Agia Napa, Cyprus, **(2004)**, pp. 492-508.
[7] J. S. Bobadilla, F. Ortega, A. Hernando and J. Bernal, "A collaborative filtering approach to mitigate the new user cold start problem", Knowledge Based Syst., vol. 26, **(2012)**, pp. 225-238.
[8] A. L. Deng, Y. Y. Zhu and B. Shi, "A collaborative filtering recommendation algorithm based on item rating prediction", Journal of Software, vol. 14, no. 9, **(2003)**, pp. 1621-1628.
[9] S. K. Lee, Y. H. Cho and S. H. Kim, "Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations", Inform. Sciences, vol. 180, no. 11, **(2010)**, pp. 2142-2155.
[10] D. Anand and K. K. Bharadwaj, "Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities", Expert Syst. Appl., vol. 38, no. 5, **(2011)**, pp. 5101-5109.
[11] H. J. Ahn, "A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem", Inform. Sciences, vol. 178, no. 1, **(2008)**, pp. 37-51.
[12] H. N. Kim, A. El-Saddik and G. S. Jo, "Collaborative error-reflected models for cold-start recommender systems", Decis. Support Syst., vol. 51, no. 3, **(2011)**, pp. 519-531.
[13] S. K. Shinde and U. Kulkarni, "Hybrid personalized recommender system using centering-bunching based clustering algorithm", Expert Syst. Appl., vol. 39, no. 1, **(2012)**, pp. 1381-1387.
[14] X. Y. Deng and C. Jin, "A novel recommendation model to mitigate new user cold start problem in mobile e-commerce", ICIC Express Letters, Part B: Applications, vol. 6, no. 7, **(2015)**, pp. 1829-1836.
[15] K. H. L. Tso and L. S. Thieme, "Empirical analysis of attribute-aware recommender system algorithms using synthetic data", Journal of Computers, vol. 1, no. 4, **(2006)**, pp. 18-29.
[16] Y. Zhao, C. Feng, J. Yang and L. Wang, "Literature review of network public opinion about the e-commerce", Review of Computer Engineering Studies, vol. 2, no. 2, **(2015)**, pp. 25-30.
[17] A. K. Milicevic, A. Nanopoulos and M. Ivanovic, "Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions", Artif. Intell. Rev., vol. 33, no. 3, **(2010)**, pp. 187-209.
[18] Z. K. Zhang, T. Zhou and Y. C. Zhang, "Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs", Physica A., vol. 389, no. 1, **(2010)**, pp. 179-186.
[19] Y. Koren, "Factor in the neighbors: Scalable and accurate collaborative filtering", ACM Transactions on Knowledge Discovery from Data, vol. 4, no. 1, **(2010)**, pp. 1-24.
[20] F. M. Belem, E. F. Martins, J. M. Almeida and M. A. Goncalves, "Personalized and object-centered tag recommendation methods for Web 2.0 applications", Information Processing and Management, vol. 50, no. 4, **(2014)**, pp. 524-553.

Author

Xiaoyi Deng, he is an associate Professor at Huaqiao University. He received his M.Sc. degree in Software Engineering (2006) and Ph.D. degree in Management Science and Engineering (2012) from Dalian University of Technology. His research focuses on data mining and recommender system. Since 2009, he is a member of the Institute of Electronics, Information and Communication Engineers, and the Systems Engineering Society of China.

