Improved Collaborative Filtering Algorithm using Topic Model

Liu Na Lu Ying Tang Xiao-jun Wang Hai-wen Xiao Peng Li Ming-xia
School of Information Science & Engineering
Dalian Polytechnic University,
Dalian China
E-mail: liuna@dlpu.edu.cn

***Abstract*—Collaborative filtering algorithms make use of interactions rates between users and items for generating recommendations. Similarity among users or items is calculated based on rating mostly, without considering explicit properties of users or items involved. In this paper, we proposed collaborative filtering algorithm using topic model. We describe user-item matrix as document-word matrix and user are represented as random mixtures over item, each item is characterized by a distribution over users. The experiments showed that the proposed algorithm achieved better performance compared the other state-of-the-art algorithms on MovieLens data sets.**

Keywords—collaborative filtering; LDA; topic model

I. INTRODUCTION

With the emergence of Internet, there is more and more information disseminating all over this channel. The abundant amount of information, however, causes difficulty for users to locate desired information, which is referred to as the information overload problem due to our limited processing ability. Therefore, recommender systems arise to assist users to acquire useful information based on their past preferences or collaborative preferences from other sources.

Most recommendation algorithms start by finding a set of customers whose purchased and rated items overlap the user's purchased and rated items. The algorithm aggregates items from these similar customers, eliminates items the user has already purchased or rated, and recommends the remaining items to the user.

Recommender systems are often based on Collaborative Filtering (CF), which relies only on past user behavior—for example, their previous transactions or product ratings—and does not require the creation of explicit profiles[1]. Notably, CF techniques do not require domain knowledge and avoid the need for extensive data collection. In addition, relying directly on user behavior allows uncovering complex and unexpected patterns that would be difficult or impossible to profile using known data attributes. As a consequence, CF attracted much of attention in the past decade, resulting in significant progress and being adopted by some successful commercial systems[2][3]. Herlocker et al. estimated a user's preference for those items by ratings, these rating is given by similar people on an items[4]. Sarwar et al. exploited similarity of items with other items that the user has already rated to predict the user's preference on items[5]. Koren et al. made use of Singular Value Decomposition (SVD) to factorize user-item rating matrix to determine latent properties of users and items[6]. Chang et al. proposed an LDA based document recommendation system which utilized an Item Based CF algorithm with document similarity calculation based on latent topic distribution of documents[7]. Liu, Qi, et al proposed a latent factor model based on LDA to model evolution of user interests based on personalized ranking[8]. Ortega et al. pointed out that there were four stages in the CF process where the users' data could be aggregated into the data of the group. According to their finding, the system performance would be significantly improved if the aggregation was done at an earlier stage of the process[9]. Wang Z et al. present Friendbook, a novel semantic-based friend recommendation system for social networks, which recommends friends to users based on their life styles instead of social graphs[10].

Our approach utilizes topic model to infer latent properties of items and then calculates user's preferences on historical ratings. We compute a hybrid user similarity score, which make use of user similarity in the topic model along with user similarity based on cosine. This way, our approach differs from the above references to improve quality of recommendations.

The paper is organized as follows. Section 2 describes the proposed algorithm, and Section 3 presents the results of applying this algorithm to MovieLens datasets. We conclude and discuss further research directions in Section 4.

II. COLLABORATIVE FILTERING RECOMMENDER USING TOPIC MODEL

A. Collaborative Filtering Algorithms

A traditional collaborative filtering algorithm is usually represented as an m×n customer-product matrix, R, such that $r_{i,j}$ is one if the *i*th customer has purchased the *j*th product, where $U=\{u_1,u_2...u_m\}$ is the set of customers, $I=\{i_1,i_2...i_n\}$ is the set of product. It is shown as Table1. We term this m×n representation of the input data set as original representation.

978-1-5090-5081-9/16 $31.00 © 2016 IEEE
DOI 10.1109/PDCAT.2016.78

$$\begin{array}{c} \\ u_1 \\ u_2 \\ \vdots \\ u_m \end{array} \begin{array}{c} \begin{array}{cccc} i_1 & i_2 & \cdots & i_n \end{array} \\ \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} \end{array}$$

Fig. 1. user-item rate matrix

The most important step in collaborative filtering algorithm is that of computing the similarity between customers as it is used to form a proximity-based neighborhood between a target customer and a number of like-minded customers. The final step of collaborative filtering algorithm is to derive the top-N recommendations from the neighborhood of customers.

The main step of collaborative filtering algorithm is to rank each item according to how many similar customers purchased it. Either cosine or correlation is bags of words. They cannot find the relation between words deeply. Topic model is another good choice.

B. Topic Model

Topic model is a generative probabilistic model of a corpus. The basic idea of topic model is that documents are represented as random mixtures over latent topics, each topic is characterized by a distribution over words.

The topic model is represented as a probabilistic graphical model in Figure 1. The boxes are "plates" representing replicates. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document. As the figure makes clear, there are three levels to the topic model representation. The parameters α and β are corpus level parameters, assumed to be sampled once in the process of generating a corpus. The variables θ_d are document-level variables, sampled once per document. Finally, the variables Z_{dn} and W_{dn} are word-level variables and are sampled once for each word in each document[11].

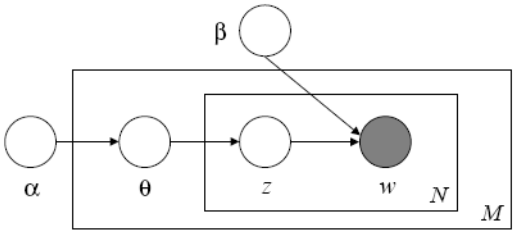

Fig. 2. Graphical model representation of LDA.

A k-dimensional Dirichlet random variable θ can take values in the $(k-1)$-simplex, and has the following probability density on this simplex:

$$p(\theta \mid \alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1} \quad (1)$$

where the parameter α is a k-vector with components $\alpha_i > 0$, and where $\Gamma(x)$ is the Gamma function. The Dirichlet is a convenient distribution on the simplex, has finite dimensional sufficient statistics, and is conjugate to the multinomial distribution. Given the parameters α and β, the joint distribution of a topic mixture , a set of θ topics **z**, and a set of N words **w** is given by:

$$p(\theta, z, w \mid \alpha, \beta) = p(\theta \mid \alpha) \prod_{n=1}^{N} p(z_n \mid \theta) p(w_n \mid z_n, \beta) \quad (2)$$

where $p(z_n \mid \theta)$ is simply θ_i for the unique i. Integrating over θ and summing over z, we obtain the marginal distribution of a document:

$$p(w \mid \alpha, \beta) = \int p(\theta \mid \alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n \mid \theta) p(w_n \mid z_n, \beta) \right) d\theta \quad (3)$$

Finally, taking the product of the marginal probabilities of single documents, we obtain the probability of a corpus:

$$p(D \mid \alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn} \mid \theta_d) p(w_{dn} \mid z_{dn}, \beta) \right) d\theta_d \quad (4)$$

C. Collaborative Filtering Recommenders using Topic Model

1) Item-Topic Distribution

As mentioned in section 3.1, topic model treats documents as mixtures of topics, and topics as mixtures of words. Inspired by this, we can regard that users(documents) are represented as random mixtures over latent topics, each topic is characterized by a distribution over items(words). It is shown as Figure 3.Given users documents, we adopt topic model to discover the distribution of topic and distribution of item-topic.

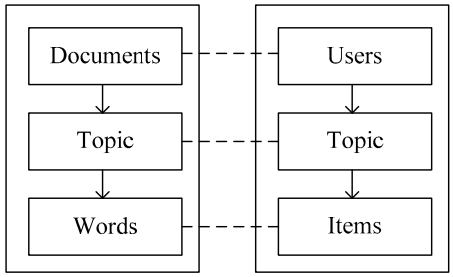

Fig. 3. A analogy between user and item of collaborative filtering algorithm

2) User-Topic Distribution

In collaborative filtering algorithm, the input data is m×n matrix as shown in Table1. The matrix is the input of topic model. The matrix is computed as Table 2 using LDA, where θ_{ij} is distribution of item i over topic t.

$$\begin{array}{c} \\ i_1 \\ i_2 \\ \vdots \\ i_n \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & \cdots & t_k \end{array} \\ \begin{bmatrix} \theta_{i_1}^{t_1} & \theta_{i_1}^{t_2} & \cdots & \theta_{i_1}^{t_k} \\ \theta_{i_2}^{t_1} & \theta_{i_2}^{t_2} & \cdots & \theta_{i_2}^{t_k} \\ \vdots & \vdots & \vdots & \vdots \\ \theta_{i_n}^{t_1} & \theta_{i_n}^{t_2} & \cdots & \theta_{i_n}^{t_k} \end{bmatrix} \end{array}$$

Fig. 4. item-topic distribution matrix

As collaborative filtering algorithm includes user-base and item-based, we have to calculate user-topic distribution according to item-topic. The user-topic distribution is determined by rating under users and item-topic distribution. It is calculated as:

$$\theta_{u_p}^{t_q} = \sum_{k \in M} r_{ik} \times \theta_{i_k}^{t_q} \times \varphi_q \tag{5}$$

where φ_q is distribution of topic q.

3) Similarity using LDA

By LDA, the count of user purchase item in matrix is denoted as distribution. The similarity of users or item is calculated by KL divergence:

$$sim_{i,j}^{\mathrm{LDA}} = \exp^{-KL(u_i,u_j)} = \exp^{-\left(KL(u_i\|u_j)+KL(u_j\|u_i)\right)} = \exp^{-\left(\sum_{k\in m}\ln\left(\frac{\theta_i}{\theta_k}\right)\theta_i + \sum_{k\in m}\ln\left(\frac{\theta_j}{\theta_k}\right)\theta_j\right)} \tag{6}$$

D. Proposed Algorithm

Collaborative filtering recommenders using topic model is described as follows:

Input: user-item rate matrix

Output: Top-N recommender

(a)Compute similarity as follows

$$sim_{i,j} = \lambda\left(\frac{1}{3}\left(sim_{i,j}^{\mathrm{c}} + sim_{i,j}^{\mathrm{p}} + sim_{i,j}^{\mathrm{ac}}\right)\right) + (1-\lambda) sim_{i,j}^{\mathrm{LDA}} \tag{7}$$

(*b*)Find neighbor according to similarity and nearest number

(*c*)Predict users as Equation (8)where M is set of neighbor.

$$\hat{r}_{u,i} = \sum_{j \in M_i} sim_{i,j} \bullet r_{u,j} \Big/ \sum_{j \in M_i} \left| sim_{i,j} \right| \tag{8}$$

(*d*)Recommender the Top-N users.

III. Experiments

We evaluated our algorithms on the MovieLens data sets. This data set consists of 100,000 ratings (1-5) from 943 users on 1682 movies. In order to evaluate our algorithm, we use Mean Absolute Error(MAE) as measure. MAE is a common measure in recommender system. It is an average of the absolute errors between predictions of target user and eventual outcomes. MAE is given by

$$MAE(u_i) = \sum_{u \in U} \left| r_{u,i} - \hat{r}_{u,i} \right| / n \tag{9}$$

A. Result of collaborative filtering using LDA with different cluster

In our experiments, we set cluster is 5,10,20,30,40,50 respectively and neighbor is 10,20,30,40,50,60,80,100,130 respectively. The number of topic is 20. We first run our proposed algorithm under above parameter. The result is shown in Figure 5 and Figure 6.

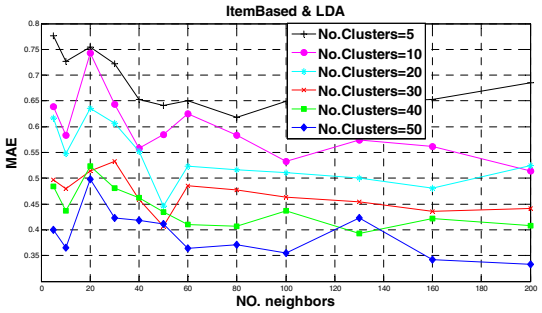

Fig. 5. Item-based LDA Collaborative Filtering algorithm

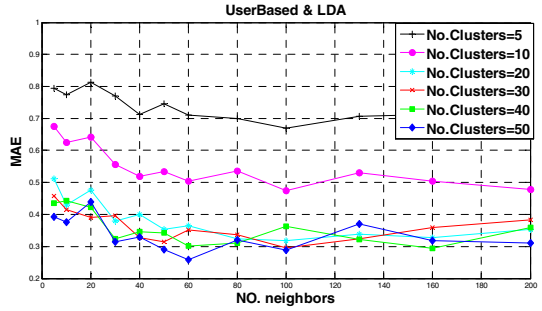

Fig. 6. User-based LDA Collaborative Filtering algorithm

From Figure 6, we can see that the MAE is lower with cluster is greater. Therefore, we compare our proposed algorithm with other three algorithm when cluster is 50.

B. Result of compared with baseline under MAE

To compare, we run the user-based and item-based collaborative filtering algorithm with baseline under MAE. The baseline includes of cosine, pearson correlation and adjusted cosine method. The compared result is shown in Figure 7 and Figure 8.

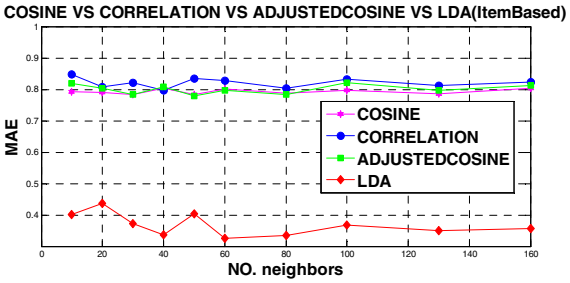

Fig. 7. Results of four item-based Collaborative Filtering algorithm with MAE

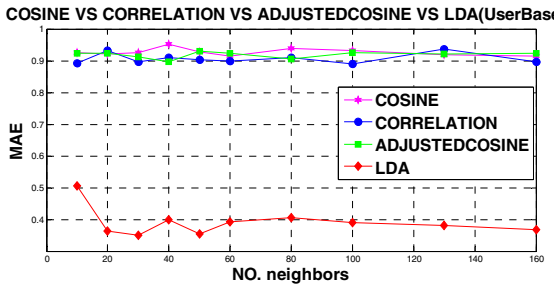

Fig. 8. Results of four user-based Collaborative Filtering algorithm with MAE

C. *Result of compared with baseline under RMSE*

We also compare our algorithm under RMSE which is root-mean-square error. The compared result under RMSE is shown in Figure 9 and Figure 10.

$$RMSE(u_i) = \sqrt{\sum_{u \in U} (r_{u,i} - \hat{r}_{u,i})^2 / n} \tag{10}$$

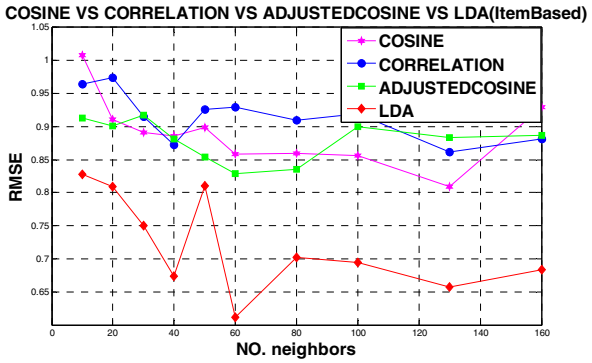

Fig. 9. Results of four item-based Collaborative Filtering algorithm with RMSE

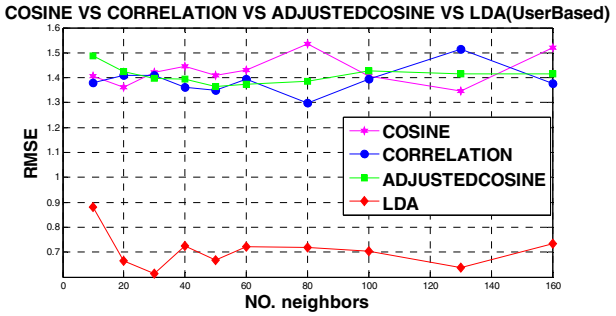

Fig. 10. Results of four user-based Collaborative Filtering algorithm with RMSE

- The result of Collaborative filtering using LDA is influenced by cluster. The cluster is lager, the MAE is lower.
- Compare with cosine, pearson correlation and adjusted cosine, neither MAE or RMSE of our proposed method is lower obviously. It indicates the precision of recommender is higher.

IV. Conclusion

Collaborative Filtering algorithms make use of interactions between users and items in the form of implicit or explicit ratings alone for generating recommendations. Similarity among users or items is calculated purely based on rating overlap in this case, without considering explicit properties of users or items involved, limiting their applicability in domains with very sparse rating spaces. In this paper, we proposed collaborative filtering algorithms using topic model, which can improved the similarity between users and items.

Acknowledgment

This work was supported by National Natural Science Foundationof China (NO.61402069, NO.61272369, NO.61175053), General project of Liaoning Provincial Department of Education (NO.L2015047).

Reference

[1] GOLDBERG, D., NICHOLS, D., OKI, B. M., AND TERRY, D. 1992. Using collaborative filtering to weave an information tapestry. Comm. ACM 35, 12, 61–70.

[2] LINDEN, G., SMITH, B., AND YORK, J. 2003. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Comput. 7, 1, 76–80.

[3] ALI, K. AND VAN STAM, W. Tivo: making show recommendations using a distributed collaborative filtering architecture. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, NY, 2004, 394–401.

[4] Herlocker, Jonathan L., Joseph A. Konstan, Al Borchers, and John Riedl. "An algorithmic framework for performing collaborative filtering." In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 230-237. ACM, 1999.

[5] Sarwar, Badrul, George Karypis, Joseph Konstan, and John Riedl. "Item-based collaborative filtering recommendation algorithms." In Proceedings of the 10th international conference on World Wide Web, ACM, 2001, 285-295

[6] Koren, Yehuda. "Factorization meets the neighborhood: a multifaceted collaborative filtering model." In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2008 ,426-434.

[7] Chang, Te-Min, and Wen-Feng Hsiao. "LDA-based Personalized Document Recommendation." , 2013.

[8] Liu, Qi, et al. "Enhancing collaborative filtering by user interest expansion via personalized ranking." Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 42.1 ,2012,218-233.

[9] Ortega, F., Bobadilla, J., Hernando, A., and Guti'errez, A. Incorporating group recommendations to recommender systems: Alternatives and performance. Information Processing & Management, 2013, 49(4): 895–901.

[10] Wang Z, Liao J, Cao Q, et al. Friendbook: A Semantic-Based Friend Recommendation System for Social Networks[J]. IEEE Transactions on Mobile Computing, 2015, 14(3):538-551.

[11] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of Machine Learning Research, 2003, 3:993-1022.