2012 Fifth International Conference on Business Intelligence and Financial Engineering

Collaborative Filtering Recommendation Algorithm Based on Item Clustering and Global Similarity

Suyun Wei, Ning Ye, Shuo Zhang , Xia Huang, Jian Zhu
College of Information Science and Technology
Nanjing Forestry University
Nanjing, 210037, China
weisuyun@njfu.edu.cn

***Abstract*—Collaborative filtering is one of the most important algorithms applied in e-commerce recommendation systems. The conventional calculations of similarities are inefficient, which suffers from data sparsity and poor prediction quality problems. In order to overcome the limitations, a new collaborative filtering recommendation algorithm based on item clustering and global similarity is proposed. Firstly, K-MEANS clustering algorithm is applied to cluster items into several classes based on users' ratings on items, and the local user similarity is calculated in each cluster. In addition, the factor of overlap is introduced to optimize the accuracy of the local similarity between users. Finally, a newly global similarity between users is presented to optimize the selection of target user's neighbors and achieve better prediction accuracy. The experimental results show that this method can improve the accuracy of the prediction and enhance the recommendation quality.**

Keywords- recommendation systems; collaborative filtering; clustering; globe similarity; overlap

I. INTRODUCTION

Collaborative Filtering (CF) is a technology that has emerged in e-Commerce applications to produce personalized recommendations for users [1]. It is based on the assumption that people who like the same things are likely to feel similarly towards other things.

However, there remain important research questions in overcoming some fundamental challenges for collaborative filtering recommender systems, such as data sparsity[2], cold-start[3], and scalability widely[4] and so on. Recently, exploiting all kinds of technologies for improved recommendation has experienced an upsurge of interest in recommender systems. Item-based algorithms [5] avoid this bottleneck by exploring the relationships between items first, rather than the relationships between users. Recommendations for users are computed by finding items that are similar to other items the user has liked. Because the relationships between items are relatively static, item-based algorithms may be able to provide the same quality as the user-based algorithms with less online computation [6]. ZENG et al. [7] adopted a matrix conversion method for similarity measure, because the number of item categories is far less than the number of the items, this method can addresses the limitations of the sparsity and scalability problems. Billsus et al. [8] proposed a method based on dimensionality reduction through the singular value decomposition (SVD) of an initial matrix of user ratings in *k* dimensions. Goldberg et al. [9] advanced collaborative filtering algorithm by applying Principal Component Analysis (PCA) to the joke recommending system developed by university of California, Berkeley. Horting is a graph-based technique in which nodes are users, and edges between nodes indicate degree of similarity between two users [10]. Predictions are produced by walking the graph to nearby nodes and combining the opinions of the nearby users. Papagelis et al. [11] proposed a method for alleviating sparsity using trust inferences, which are transitive associations between users in the context of an underlying social network and are valuable sources of additional information that help dealing with the sparsity and the cold-start problems. Clustering techniques [12] work by identifying groups of users who appear to have similar preferences. Once the clusters are created, predictions for an individual can be made by averaging the opinions of the other users in that cluster.

In the collaborative filtering algorithm, the precision of the similarity between users is crucial for the quality of the recommender systems. However, the conventional calculations of similarities are inefficient because of data sparsity. To address the problem, a new collaborative filtering recommendation algorithm based on item clustering and global similarity is proposed. The kernel idea is that a new method of global similarity between users based on item clustering is adopted, by this new strategy, clustering algorithm is applied to cluster items into several classes based on users' ratings on items, and the local user similarity is calculated in each cluster. We also propose a method of adjusting the factor of overlap in order to optimize the accuracy of the local similarity between users. Finally, these techniques are applied in MovieLens data set, and the experimental results show that the new method of global similarity between users can improve the accuracy of the prediction and enhance the recommendation quality.

II. RELATED WORK

Collaborative filtering algorithm is processed in item-user rating matrix. User-item matrix usually is described as a $m \times n$ ratings matrix R_{mn}, shown as formula (1), where row represents m users and column represents n items. The element of matrix r_{ij} means the score rated to the user i on

978-0-7695-4750-3/12 $26.00 © 2012 IEEE
DOI 10.1109/BIFE.2012.23

the item j, which commonly is acquired with the rate of users' interest.

$$R_{mm}=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} \tag{1}$$

One critical step in user-based collaborative filtering is to compute the similarity between users and then to select the nearest neighbors. There are a number of different ways to compute the similarity between users. Here we present two such methods. These are cosine-based similarity, correlation-based similarity.

Cosine-based similarity: In this case, two users are thought of as two vectors in the n dimensional user-space. The similarity between them is measure by computing the cosine of the angle between these two vectors. Formally, in the $m \times n$ ratings matrix, similarity between users u and v, denoted by $sim(u, v)$ is given by

$$sim(u,v)=\cos(\vec{u},\vec{v})=\frac{\vec{u}\cdot\vec{v}}{\|\vec{u}\|\times\|\vec{v}\|}=\frac{\sum_{i=1}^{n} r_{ui}\cdot r_{vi}}{\sqrt{\sum_{i=1}^{n} r_{ui}^{2}}\sqrt{\sum_{i=1}^{n} r_{vi}^{2}}} \tag{2}$$

Where "." denotes the dot-product of the vectors.

Correlation-based similarity: In this case, similarity between two users u and v is measured by computing the *Pearson-r* correlation $corr_{u,v}$. To make the correlation computation accurate we must first isolate the co-rated cases (i.e., cases where the items rated by u and v). Let the set of items which both rated by u and v are denoted by I_{uv} then the correlation similarity is given by

$$sim(u,v)=\frac{\sum_{i\in Iuv}(r_{ui}-\bar{r}_{u})(r_{vi}-\bar{r}_{v})}{\sqrt{\sum_{i\in Iuv}(r_{ui}-\bar{r}_{u})^{2}}\sqrt{\sum_{i\in Iuv}(r_{vi}-\bar{r}_{v})^{2}}} \tag{3}$$

Here $\bar{r}_u$ is the average rating of the u-th user. i.e.

$$\bar{r}_{u}=\frac{1}{|I_{uv}|}\sum_{i\in Iuv} r_{ui}\ ,\ \ \bar{r}_{v}=\frac{1}{|I_{uv}|}\sum_{i\in Iuv} r_{vi} \tag{4}$$

The most important step in a collaborative filtering system is to generate the output interface in terms of prediction. Once we isolate the nearest neighbors set base on the similarity measures, the next step is to look into the target users' ratings and use a technique to obtain predictions. Here the prediction on an item i for a user u, denoted by P_{ui} is give by

$$P_{ui}=\bar{r}_{u}+\frac{\sum_{v\in Nu} sim(u,v)(r_{vi}-\bar{r}_{v})}{\sum_{v\in Nu} sim(u,v)} \tag{5}$$

Where N_u is the *top-N* nearest neighbors set of target user u.

III. COLLABORATIVE FILTERING ALGORITHM BASED ON GLOBAL SIMILARITY

A. clustering item

The K-Means Clustering method creates k clusters each of which consists of the items which have similar ratings among themselves. In this method we first select arbitrarily k items as the initial center points of the k clusters, respectively. Then each item is assigned to a cluster in such a way that the distance between the item and the center of a cluster is minimized. The distance is calculated using the similarity between the items. Then, for each cluster, we recalculate the mean of the cluster based on the items which currently belong to the cluster. After finding new centers, we compute the distance for each item as before in order to find to which cluster the item should belong. Recalculating the means and computing the distances are repeated until a terminating condition is met.

Algorithm 1 clustering item

```
Input: rating matrix $R_{mn}$, the items $I=\{i_1,i_2,...,i_n\}$ .
Output: item clusters $C=\{c_1,c_2,...,c_k\}$.
Method:
a) Initialize $k$ prototypes $(w_1,...,w_k)$ such that $w_j=i_l$, $j\in\{1,...,k\}$, $l\in\{1,...,n\}$;
b) Each cluster $c_j$ is associated with prototype $w_j$;
c) Repeat
     for each input vector $i_l$, where $l\in\{1,\ldots,n\}$, do
       Assign $i_l$ to cluster $c_{j*}$ with nearest prototype $w_{j*}$
       (i.e., $sim(i_l, w_{j*})\ge sim(i_l, w_j)$, $j\in\{1,\ldots,k\}$ )
     for each cluster $c_j$, where $j\in\{1,\ldots,k\}$, do
       Update the prototype $w_j$ to be the centroid of all samples currently in $c_j$, so that $w_j=\sum_{il\in cj} i_l / |c_j|$
     Compute the error function $E=\sum_{j=1}^{k}\sum_{il\in cj}|i_l-w_j|^2$
   Until E does not change significantly
```

B. Local similarity Calculation

Assuming that items $I=\{i_1,i_2,...,i_n\}$ could be clustered into k groups $C=\{c_1,c_2,...,c_k\}$. We present correlation-based similarity on clusters to compute the local similarity between users. The local similarity between user u and user v on the item cluster c_j is defined as:

$$lsim(u,v,j)=\frac{\sum_{i\in I_{uv}^{j}}(r_{ui}-\bar{r}_{u}^{j})(r_{vi}-\bar{r}_{v}^{j})}{\sqrt{\sum_{i\in I_{uv}^{j}}(r_{ui}-\bar{r}_{u}^{j})^{2}}\sqrt{\sum_{i\in I_{uv}^{j}}(r_{vi}-\bar{r}_{v}^{j})^{2}}} \tag{6}$$

Where I_{uv}^{j} is the items set co-rated by users u and v in the item cluster c_j, i.e., $I_{uv}^{j}=I_u\cap I_v\cap c_j$, and $\bar{r}_u^{j}$ is the average rating of the u-th user in the item cluster c_j.

In order to avoid the inaccuracy in conventional collaborative filtering algorithm that the similarity between user co-rated less, almost no items is high, then the factor of overlap is introduced to optimize the accuracy of the local similarity calculation, represented as formula (7).

$$lsim'(u,v,j)=\frac{Min(|I_u\cap I_v\cap c_j|,\gamma)}{\gamma}lsim(u,v,j) \tag{7}$$

Where $\left|I_u \cap I_v \cap c_j\right|$ denotes the number of items co-rated by users *u* and *v* in the item cluster c_j, and *γ* is a parameter that takes a value increases with the number of the items co-rated. The improved local similarity can guarantee effectively only if the users who have enough items co-rated are able to come into the nearest neighbors.

C. Global similarity Calculation

We can calculate *k* adjusted local similarities *lsim'*(*u*,*v*,*j*) (*j*=1,2,…,*k*) between users *u* and *v* in the *k* items clusters, thus the global similarity between user *u* and *v* is formalized as formula(8).

$$sim(u,v) = \sum_{j=1}^{k} lsim(u,v,j) \tag{8}$$

We now describe the proposed recommendation algorithm in detail.

```
Algorithm 2 Collaborative filtering algorithm based on item clustering and global similarity(ICGS)
Input: user-item rating matrix $R_{mn}$, the first $N$ items of the top-N recommended set.
Output: the top-N recommended set.
Method:
a) Create $k$ clusters $C=\{c_1,c_2,\dots,c_k\}$ with Algorithm 1;
b) for any two users $u$ and $v$, do
      Calculate $k$ local similarities $lsim'(u,v,j)(j=1,2,\dots,k)$;
      Obtain global similarity $sim(u,v)$ according to the formula (10);
      Add $sim(u,v)$ into the similarity matrix $R_{sim}$ ,i.e., $R_{sim}=R_{sim} \cup sim(u,v)$;
c) for each user u, do
      Obtain the K-nearest neighbors $N_u=\{v_{i1},\dots,v_{iK}\}$,
      $u \notin N_u$ and $sim(u,v_{i1}) \geq \dots \geq sim(u,v_{iK})$ according to
      the corresponding similarities in matrix $R_{sim}$,
d) for each user u, do
      Compute the prediction of non-purchased items
       according to the formula (6);
       Sort items in non-increasing order with respect to
       that prediction, and the first $N$ items are selected as
       the top-N recommended set.
```

IV. Experimental Results and Analysis

A. data set

We use data set provided by MovieLens. The website is created by GroupLens Research at University of Minnesota.100K of public data set is selected in this paper, one hundred thousand records, including 1628 movies rated by 943 users. Each user at least rates 20 movies; rate them from one to five stars, five being the best. Users show their interest by number they rated. We divided the database into a training set and a test set which 80% of the data was used as training set and 20% of the data was used as test set.

B. Performance Measurement s

Mean Absolute Error (MAE) between ratings and predictions is widely used to evaluate the quality of collaborative filtering methods. The MAE is a measure of the deviation of recommendations from their true user-specified values.For each ratings-prediction pair<p_i, q_i > this metric treats the absolute error between them, i.e., | p_i, q_i | equally. The MAE is computed by first summing these absolute errors of the *n* corresponding ratings-prediction pairs and then computing the average. Formally,

$$MAE = \frac{\sum_{i=1}^{n} \left|p_i - q_i\right|}{n} \quad .(9)$$

C. Adjusting parameters

In our algorithm, we have two parameters to choose: number of clusters *k*, degree of overlapping *γ*.

It is significant to select number of clusters, when we use K-means method to cluster. In order to have proper number of clusters, we use different value of *k* (10, 20, 30, 40, and 50) to test the prediction. From Figure 1, we can know that number of clusters will have influence on the prediction. When the number of clusters is quite small, the cluster is too similar to display the difference between items. When the number of clusters is too big, the clustering shows excessive individuation that also cannot show the difference between items. The predictability is quite well when number of cluster is 30.

The overlap parameter *γ* is to adjust the similarity between users, making the calculation more reasonable. We can handle users who rated a few but have a high level of similarity by taking advantages of *γ*. Figure 2 shows the *γ* influence on the prediction. Results show that with increasing of *γ*，MAE is gradually decreasing ,this account for the predicted rate is more close to the real rate.

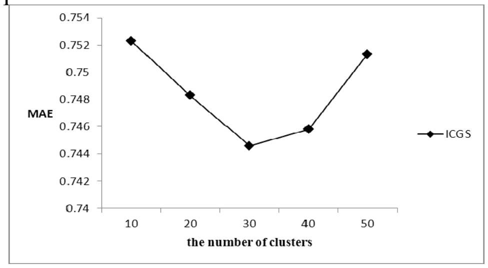

Figure 1. Impact of the number of clusters on MAE.

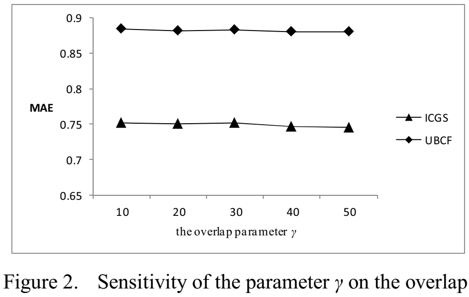

Figure 2. Sensitivity of the parameter *γ* on the overlap.

D. *Comparing the experimental results*

The size of the nearest neighborhood has a significant effect on the prediction quality. In the experiment, the size of the neighborhood is increasing from 3 to 40.We calculate the MAE to compare similarity by using Item Clustering for the Global Similarity (ICGS), User Based Collaborative Filtering (UBCF), Item Based Collaborative Filtering (IBCF), test parameters are set before, experiment result shows in Figure 3.

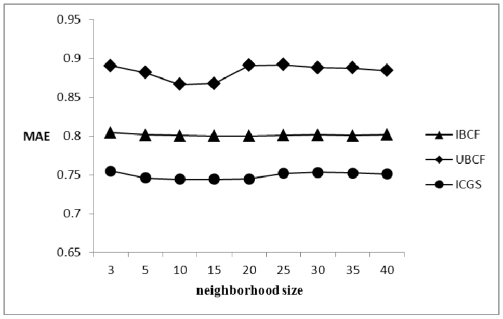

Figure 3. Comparison of prediction quality of several collaborative filtering algorithms.

When the number of nearest neighborhood is 3, our algorithm is much better than the user based collaborative filtering, because user based collaborative filtering cannot show similarity accurately between users when the dataset is too sparse to get enough neighborhood. In this case, it cannot distinguish predicted effective users by known users, so that cannot distribute proper weight when predicting. In our algorithm, by introducing factor of overlapping, calculating the local similarity, we can compensate for the user based collaborative filtering‘s weakness to improve the accuracy of the prediction. With the increasing number of neighborhood, our proposed algorithm is better than the user based collaborative filtering and item based collaborative filtering.

V. CONCLUSIONS

In collaborative filtering algorithm, similarity measurement direct influence the prognostication precision and quality of recommendations of the algorithm, this paper proposed a personalized recommendation approach joins the item clustering technology and the global similarity between users. Meanwhile, we introduce the factor of the overlapping to calculate the local similarity in order to make the result more precise. From the result of the experiment, the algorithm proposed in this paper can improve the accuracy of the prediction and enhance the recommendation quality.

ACKNOWLEDGMENT

The work is partially supported by National Basic Research Program of China (No. 2012CB114505), China National Funds for Distinguished Young Scientists(No. 31125008) ,Natural Science Foundation of Jiangsu Province(No.BK2009393), Jiangsu Qing Lan Project and Scientific and Technological Innovation Foundation of Jiangsu (No. CXLX11_0525), Technological Innovation Foundation of Nanjing Forestry Scientific (No. 163070079), the Innovation Fund Project for College Student of Jiangsu (No. 164070742).

REFERENCES

[1] D. Goldberg, D. Nichols, B.M. Oki , et al. “Using collaborative filtering to weave an information tapestry,” Communications of the ACM , vol.35(12) ,pp. 61-70, 1992,.

[2] B.M. Sarwar, G. Karypis, J.A. Konstan, et al. “Application of dimensionality reduction in recommender system -- A Case Study,” Proceedings of the ACM Web KDD Web Mining for E-Commerce Work shop. Boston, MA, United States , 2000,pp.82-90.

[3] P. Massa and P. Avesani, “Trust-Aware Collaborative Filtering for Recommender Systems,” Lecture Notes in Computer Science, vol.3290, pp. 492-508, 2004.

[4] S.Z. Vincent and F. Boi, “Using Hierarchical Clustering for Learning the Ontologies used in Recommendation Systems,” Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San-Jose, California, USA , 2007,pp.599-608.

[5] B.M. Sarwar, G. Karypis, J.A Konstan , et al. “Item-based collaborative filtering recommendation algorithms,” Proc. 10th International Conference on the World Wide Web. New York, ACM Press, 2001, pp.285-295.

[6] B.M. Sarwar, “Sparsity, scalability, and distribution in recommender systems,” Minneapolis, MN: University of Minnesota, 2001.

[7] C. Zeng, C.X. Xing and L.Z Zhou. “Similarity measure and instance selection for collaborative filtering,” In Proceedings of the 12th International Conference on World Wide Web. New York, ACM Press, 2003, pp.652-658.

[8] D.Billsus and M.J Pazzani, “Learning collaborative information filters,” In Proceedings of the 15th International Conference on Machine Learning. San Francisco, Morgan Kaufmann Publishers, 1998, pp.46-54.

[9] K. Goldberg, T. Roeder, D. Gupta, et al. “Eigentaste: a constant time collaborative filtering algorithm,” Information Retrieval, vol.4(2),pp. 133-151, 2001.

[10] C.C.Aggarwal, J.L.Wolf, K.L.Wu, et al. “Horting hatches an egg: a new graph-theoretic approach tocollaborative filtering,” Proc. 5th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, New York, ACM Press, 1999. pp.201-212.

[11] M. Papagelis, D. Plexousakis and T. Kutsuras, “Alleviating the sparsity problem of collaborative filtering using trust inferences,” In Proceedings of the 3rd International Conference on iTrust 2005, LNCS3477. Berlin: Springer-Verlag, 2005, pp.224-239.

[12] L.H Ungar and D.P Foster, “Clustering methods for collaborative filtering,” In Proceedings of the Workshop on Recommendation Systems at the 15th National Conference on Artificial Intelligence. Menlo Park, CA: AAAI Press, 1998, pp.112-125.