International Journal of u- and e- Service, Science and Technology
Vol.9, No. 3 (2016), pp.169-178
http://dx.doi.org/10.14257/ijunesst.2016.9.3.17

An Improved Collaborative Filtering Recommendation Algorithm

Shulin Liu

Hainan College of Economics and Business,
Haikou 571127, China
[1]106228605@qq.com

Abstract

An improved CF algorithm based on the classification of items is introduced to overcome the problems caused by the data sparseness and inaccuracy of the user neighbors. The new algorithm first rates the unrated items by applying the item classification, and then calculates the user similarity within classes for nearest-neighbors, after which it could recommend the items based on the final prediction.

Keywords: *Personalized Recommendation, Collaborative filtering, Item classification*

1. Introduction

In the real e-commerce system, all commodities are divided into various categories, some of which even includes many sub-categories [1-3]. Generally speaking, consumers show interests in only three or four categories. They only browse or purchase specific commodities in which they're interested. They rate commodity categories about which they concern and are indifferent to uninteresting ones [4-5]. So it's believed that consumers focusing on common items in the same category have similar interest. When the similarity between items in the same category is identical to that between items in different category, the recommendation based on commodities in the same category will become more effective [6-9]. One user usually has a few different interests, while those with common interests (*i.e.*, two users have all similar interests) are very rare. So when user interest is unlike, the nearest neighbors shall differ as well. We divide movie item category (supposedly five categories) based on the respective interest of user A, B, C, D and E, as shown in Table 1.

Table 1. Table of User-Interest

user	Film project classification				
	Comedy	Affectional film	Martial arts film	suspense film	Cartoon film
A	1	1	0	1	0
B	1	0	1	0	1
C	0	1	0	1	0
D	1	0	0	1	0
E	0	1	1	0	1

Assume the number of the closest neighbors of user interest N=2. From Table 1, we know that regarding the interest category "comedy", user A's interest neighbors are user B and D; for "affectional film", user A's interest neighbors are user C and E; for "suspense film", user A's interest neighbors are user C and D. Hence in short, for users with different interests, their neighbors are different, which is in line with the reality.

When the idea of item classification is not adopted, we still assume the number of the closest neighbors of user interest N=2. From Table 1, we see user A's interest neighbors

ISSN: 2005-4246 IJUNESST

are user C and D who have the maximum common interests with A; in other words, for A's all interests, its closest neighbors are C and D. To make recommendations to A, we can do according to the interest of C and D. However, to recommend as per some interests of user A, again such as "comedy" and "affectional film", C or D doesn't love them and that they don't make any ratings, leading to bad recommendation result.

Based on that idea, we propose individualized recommendation algorithm, which is described as follows:

(1) Utilize item classifying information to divide user rating records as per category and put in the matrix of user-item category; calculate inter-class item similarity and complete rating prediction of unrated items to compensate data sparseness;

(2)Estimate inter-class user similarity to obtain the nearest neighbors of users with different interests and thus to improve the precision for searching such neighbors;

(3)Generate recommendation set. The improved collaborative filtering algorithm item classification needs to calculate only the category of new added items when the rating matrix is being updated. In practical application, the method can greatly increase system efficiency and scalability.

2. The Improved Collaborative Filtering Algorithm Based on Item Classification

2.1. Traditional Collaborative Filtering Algorithms

Traditional collaborative filtering algorithms have the following steps:

(1) Data pre-processing: this is designed to obtain one n*m user-item rating matrix, where column m refers to the number of user; n is number of item; the matrix element $R_{i,j}$ stands for user i's rating value of item j;

(2) Matrix completion: one method is simple fill-up, *i.e.* use a fixed value like 0, medium value or user mean value to fill up those unrated items in the matrix; the other is predictive fill-up; that is, calculate item similarity in the matrix to get similar neighbors of the item; then make predictive scores for unrated items and fill them up;

(3) Attainment of k-nearest neighbor of user interest: calculate user similarity based on user-item rating matrix and get one set of the closest neighbors for the current user according to similarity degree;

(4) Heneration of dataset: after the user interest k-nearest neighbors are obtained, we can get its interest degree and Top-N recommendation set regarding any item. To sum up in traditional collaborative filtering algorithm, the most important is to get user or item similarity.

Traditional similarity measurement method. The traditional method of similarity measurement is mainly Cosine, Adjusted Cosine and Pearson Correlation, it is as follows:

(1)Cosine

User rating as n dimensional vector space, used Euclidean formula to calculate the cosine of the angle of two vectors. When the user similarity is calculated, the user's score of all items is considered as the n dimensional space vector, and the user's similarity is measured by computing the cosine angle of two vectors, it is shown in Formula 1.

$$sim(u_i, u_j) = \cos(\vec{u}_i, \vec{u}_j) = \frac{\vec{u}_i \cdot \vec{u}_j}{\left|\vec{u}_i\right| \times \left|\vec{u}_j\right|} = \frac{\sum_{c=1}^{n} R_{i,c} \cdot R_{j,c}}{\sqrt{\sum_{c=1}^{n} R_{i,c}^2} \sqrt{\sum_{c=1}^{n} R_{j,c}^2}} \quad (1)$$

(2)Adjusted Cosine

Set user u_i and u_j have a score of the items I_i and I_j, while the user u_i and u_j score of the project set with I_{ij} description.

user u_i and u_j similarity are shown in formula2 :

$$sim(u_i,u_j)=\frac{\sum_{c\in I_{ij}}\left(R_{i,c}-\overline{R_i}\right)\left(R_{j,c}-\overline{R_j}\right)}{\sqrt{\sum_{c\in I_i}\left(R_{i,c}-\overline{R_i}\right)^2}\sqrt{\sum_{c\in I_j}\left(R_{j,c}-\overline{R_j}\right)^2}} \tag{2}$$

(3) Correlation

In the related similarity computation, user u_i and u_j score of the items are I_{ij}, then the similarity of user u_i and u_j can be obtained by calculating the Pearson correlation. it is shown in Formula3.

$$sim(u_i,u_j)=\frac{\sum_{c\in I_{ij}}\left(R_{i,c}-\overline{R_i}\right)\left(R_{j,c}-\overline{R_j}\right)}{\sqrt{\sum_{c\in I_{ij}}\left(R_{i,c}-\overline{R_i}\right)^2}\sqrt{\sum_{c\in I_{ij}}\left(R_{j,c}-\overline{R_j}\right)^2}} \tag{3}$$

2.2. Method for Predicting Scores of Unrated Items Based on Item Classification

The method separates user-item rating records as per category to several category matrices; for each matrix, calculates inter-class item similarity to get the nearest neighbors of item; then based on such neighbors, predict scores of unrated items and fill them up.

If the goods are not divided into categories, you can use the clustering algorithm to divide the project into K classification. Assume that the project category $C=C_1\cup C_2\cup\ldots\cup C_K$ represents a collection of all the items. $C_j=\{I_{j,1},I_{j,2}\ldots,I_{j,k}\}$ represents j^{th} class.

When inter-class item similarity is the same with item similarity between different categories, generally the former similarity has higher degree. Based on that, we suggest calculating inter-class item similarity to get item's closest neighbors; then foresee grades of unrated items and complete them. The implementation is introduced like Algorithm 1.

Algorithm 1. The Method for Predicting Scores of Unrated Items

Input: user rating data and item information file Item

Output: predict user u's scoring value $P_{u,j}$ of unrated item j

(1)Data pre-processing: convert item information file Item into item class matrix $C_{i,j}$; process user rating Data to get user-item rating matrix $R_{m\times n}$;

(2)As per user-item rating matrix $R_{m\times n}$ and item class matrix $C_{i,j}$, divide rating records to k class matrices, $R_{m\times n}=R_1\cup R_2\cup\ldots\cup R_k$, where R_j is rating matrix of the j^{th} class; the division is done in this way: scan in proper order the ID of items in one category in matrix $C_{i,j}$; then pass such ID to user-item rating matrix $R_{m\times n}$; in $R_{m\times n}$, read out the ID-related item ID (IID), user ID (UID) and rating as to write to relative class matrix; stops till all items' rating records in

$C_{i,j}$ are assigned to relative category; when one item belongs simultaneously to several categories, in each relative class matrix, there will have its IID, UID and rating;

(3)Based on k class matrices got in step ②, calculate the similarity between unrated item j and any item in the same category with formula (4) in each class matrix; suppose item i and j, whose similarity is:

$$sim(i,j) = \cos(\vec{i},\vec{j}) = \frac{\vec{i},\vec{j}}{|\vec{i}|,|\vec{j}|} = \frac{\sum_{c=1}^{m} R_{u,i} \cdot R_{u,j}}{\sqrt{\sum_{c=1}^{m} R_{u,i}^{\ 2}} \sqrt{\sum_{c=1}^{m} R_{u,j}^{\ 2}}} \tag{4}$$

(4)Search the neighbor set of unrated item j in each inter-class matrix space, which is made as $M_j = \{I_1, I_2, \ldots, I_v\}$; the similarity between item $I_1, I_2, \ldots, I_v$ and j degrades; calculate the number n of all neighbors in neighbor set M; make N the number of item j's nearest neighbors; when N>n, item j's nearest neighbor set is M_j ; when N<n, choose the first N in $M_j = \{I_1, I_2, \ldots, I_v\}$ as the item j's closest neighbor set M_j ;

(5)For the nearest neighbor set M_j of unrated item j got from each class matrix, use the method stated in [10] to predict user u's marks on item j;

$$P_{u,j} = \frac{\sum_{n \in M_j} sim_{j,n} \cdot R_{u,n}}{\sum_{n \in M_j} (|sim_{j,n}|)} \tag{5}$$

(6)When item j belongs to a few categories, calculate the mean value of $P_{u,j}$ and use it as the final prediction rating $P_{u,j}$ of user u for item j; finally choose rounding numbers of $P_{u,j}$ and enter into the rating matrix $R_{m \times n}$.

2.3. Improved Collaborative Filtering Algorithm Based on Item Classification

Algorithm 2 the improved collaborative filtering algorithm based on item classification

Input: user rating record Data and item information file Item

Output: target user UID's prediction rating $P_{UID,IID}$ of item IID

(1)Data pre-treatment

Treat user rating record Data to get user-item rating matrix $R_{m \times n}$; meanwhile, utilize item class matrix $C_{i,j}$ to divide user-item rating matrix $R_{m \times n}$ into k categories;

(2)Obtain inter-class nearest neighbors of user

Regarding k class matrices, employ formula (1) respectively to calculate the similarity between any items in the same class and form inter-class user similarity matrix;

(3)Produce recommendation value

According to the weighted mean of each user's rating of item IID in the interest nearest neighbor C_{UID} , $P_{UID,IID}$ is acquired in the following way:

$$P_{UID,IID} = \overline{P_{UID}} + \frac{\sum_{u \in C_{UID}} sim(UID,u) x (R_{u,IID} - \overline{R_u})}{\sum_{u \in C_{UID}} |sim(UID,u)|} \tag{6}$$

(4)Judgment of multiple classes

When target item IID belongs simultaneously to multiple classes, in each its belonged matrix, calculate user UID's predictive rating $P_{UID,IID}$ of item IID in the same class; then get mean value as the final predictive scores of UID for IID;

(5)Make recommendation

Take the specified strategy and select user UID's neighboring users in C_{UID}; then recommend their preferences to target user as to help it find out new possible interest.

3. Experiment Design and Discussion

3.1. Experimental Dataset

The dataset for this experiment is built about film sites by research team GroupLens in University of Minnesota, USA for studies on personalized recommendation, which is 100k open dataset provided by the site MovieLens. This set includes 100,000 rating records about 1682 movies by 943 users, of which each user comments at least 20 movies. According to the recent statistics, MovieLens has 40,000 registered users because of truthful and accurate data and abundant contents and at least 3,500 movies were evaluated. MovieLens dataset is widely applied for studies on various algorithms of personalized recommendation system. It is authoritative data source in the field [11].

The 100k MovieLens data set consists of the following files, file name and file contents as follows:

(1)u.data: user id | item id | rating | timestamp

The paper mainly includes the user serial number UID, the project serial number IID, the user's score for the project Rating, the timestamp of four. In the experiment, data preprocessing is obtained by the user - item score matrix $R_{m\times n}$, which is composed of $R_{m\times n}$, , $R_{m\times n}$ is is a matrix of m row n column, and the format is as follows:

$$R_{mxn}=\begin{bmatrix} R_{1,1} & R_{1,2} & \cdots & R_{1,n} \\ R_{2,1} & R_{2,2} & \cdots & R_{2,n} \\ \cdots & \cdots & \cdots & \cdots \\ R_{m,1} & R_{m,2} & \cdots & R_{m,n} \end{bmatrix}$$

(2) u.item: movie id | movie title | release date | video release date | IMDb URL | unknown | Action | Adventure | Animation | Children's | Comedy | Crime | Documentary | Drama | Fantasy | Film-Noir | Horror | Musical | Mystery | Romance | Sci-Fi | Thriller | War | Western |

The file is project information. Five fields: movie ID (IID), title of the movie, the movie released video time, release time, customers; 19 field behind the movie category list. The corresponding field for the 1 to show the film belongs to the category.

(3) u.genre: genre title | genre id

This paper mainly lists the project category name and the corresponding category code, namely, the 0-18 to represent the 19 categories, the order of the corresponding sequence of documents u.item.

(4) u*.base 和 u*.test

The dataset contains the u1.base~u5.base five training sets and the corresponding u1.test~u5.test five test sets. The training set and test set are randomly divided according to the proportion of 80% and 20%.

(5) u.user: user id| age | gender | occupation | zip code

This file lists the main information of the participating users, such as user UID, age, gender, occupation and zip code *etc.*.

User score is 1,2, 5, 3, 4, 5 grades, the score is higher, show that the user is more like the movie, and vice versa, that is, 5 said the most like, 1 said the least favorite.

The experiment mainly uses the data to focus on the first four files, and then uses the data. The hardware configuration of the machine: Core Duo (TM) 2 CPU 2.00GHz Intel, RAM 2 GB, hard disk 500G. Running environment: operating system isWin7, development platform is MATLAB 2010.

3.2. Evaluation Criteria

The quality of recommendation is a decisive factor for the sustainable development of system: good recommendation quality can help attract new users in addition to keeping users' higher loyalty; on contrary, poor recommendation quality will lead to fewer users because of bad user experience.

Indicators for evaluating the quality of system recommendation are differing for different system targets. However, there're mainly two criteria: Statistical Accuracy Metrics and Decision Support Accuracy Metrics [12].

Mean absolute error, as a statistical precision method, is the most often used to measure the quality of recommendation. By calculating errors between system's predictive recommendation value and user actual evaluating value, it examines the accuracy of recommendation. Normally, people would get predictive recommendation value through training; then do testing with MAE; the smaller MAE value is, the higher the recommendation quality of system becomes.

Suppose in the testing dataset, the set of items rated by user U_i is $\{p_1, p_2, ..., p_N\}$. With the proposed recommendation algorithm, we can predict relative rating set is $\{r_1, r_2, ..., r_N\}$; and that MAE can be reached by the following equation:

$$MAE = \frac{\sum_{i=1}^{N} | p_i - r_i |}{N} \tag{7}$$

3.3. Experiment and Result Analysis

Among traditional similarity measuring methods, cosine similarity measuring method is easily implemented and has rapid speed of prediction, along with high precision of prediction. So here we use cosine similarity measuring method to calculate both item and user similarity during the testing with the improved collaborative filtering algorithm based on item classification.

Experiment One: variation of MAE value for different values of parameters

1) Experiment content:
(1) Fix the number of item neighbors and observe the change of MAE value when the number increases from 10 to 50;
(2) Fix the number of user neighbors and observe the change of MAE value when the number grows up from 10 to 50;
2) The experimental results are shown in Figure1.

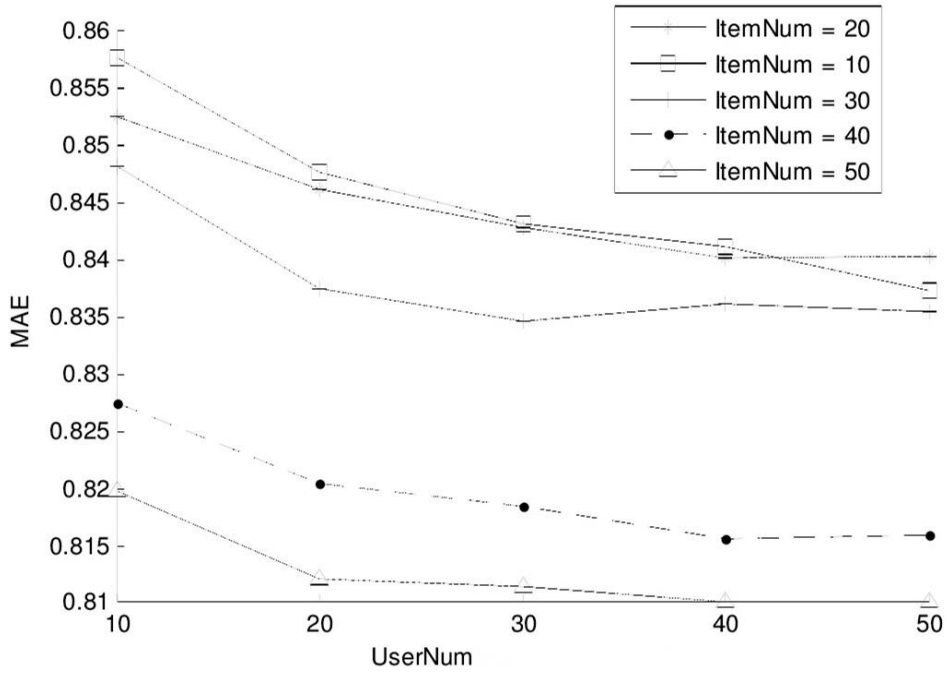


Figure 1. Comparison of Collaborative Filtering Recommendation Algorithms of Different Nearest Neighbors

3) Result analysis

(1) Fix the number (ItemNum) of item neighbors: when the number (UserNum) of user neighbors changes in [50,10], MAE value is descending with increasing UserNum; and when ItemNum>20, UserNum>40, MAE value becomes stable;

(2) Fix the number (UserNum) of user neighbors: when the number (ItemNum) of item neighbors varies in [50,10], MAE value is reducing with bigger ItemNum; and when ItemNum=50, MAE reaches the best value.

It's concluded that variation of item neighbors and user neighbors in a certain range will have great impacts on the recommendation quality of system. Therefore, it's rather important to obtain accurate the closest neighbors of user interest.

Experiment two: comparison of this testing result and other findings

1)Experiment content:

With identical dataset and both the training set and testing training of similar percentage, we compare results of recommendation by the proposed algorithm and Cosine method, improved collaborative filtering method [13] , and the similarity measuring method based on user-interest collaborative filtering [14] .

2)The experimental results are shown in Figure2.

3) Result analysis

(1)For all recommendation algorithms, MAE value goes down with bigger user neighbors (UserNum);

(2)MAE value of the method in [13] varies a lot and of other three methods changes a little; the proposed method has the least changing MAE value, which is 0.02;

The conclusion is the method here controls MAE in [0.80,0.82]; while the other three's MAE value is >82.0, with big fluctuation. It implies that the improved collaborative filtering algorithm based on item classification gets less changing nearest neighbors of interest, who have very similar interest as the user. It realizes better recommendation effect of the system.

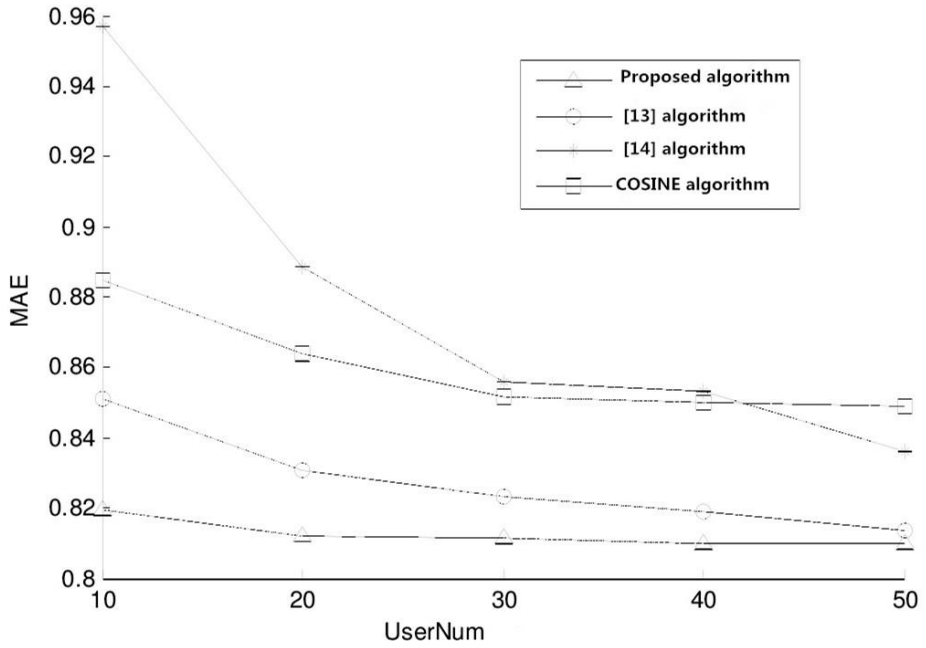


Figure 2. Comparison of MAE of Recommendation Algorithms

4. Conclusion

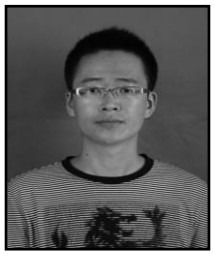
With item classifying information, different nearest neighbors can be acquired for different user interests, which is quite in accordance with the category of user interest in reality. Through testing, we find that we can obtain more accurate nearest neighbors of user interest with item classification information, and thus to produce precise recommendation, reduce errors of recommendation and enhance the quality of the system.

References

[1] X. Zhongyang and L. Wentian, "Improved algorithm for collaborative filtering based on project classification", Computer application research, vol. 02, **(2012)**, pp. 493-496.
[2] S. Zhang Yang, "Based on neighbor users and neighboring projects to improve the collaborative filtering algorithm", Journal of Shenyang Normal University (Natural Science Edition), vol. 03, **(2012)**, pp. 382-385.
[3] X. Zhongyang and Z. Yufang, "A collaborative filtering recommendation algorithm based on project classification and cloud model", Computer application research, vol. 10, **(2012)**, pp. 3660-3664.
[4] K. Zhongrong, "Research on Collaborative Filtering Recommendation Algorithm Based on project classification and prediction", Beijing University of Chemical Technology, **(2013)**.
[5] S. Nanjun and L. Tianshi, "Collaborative filtering algorithm based on project classification and user interest", Computer engineering and applications, vol. 10, **(2015)**, pp. 128-131.
[6] K. Zhongrong, "Research on the collaborative filtering algorithm based on feature classification and filling of the project", Henan science and technology, vol. 12, **(2013)**, pp. 3-5.
[7] Z. Yimeng and X. Yinghua, "The improved algorithm of collaborative filtering by the neighbors", The application of computer system, vol. 06, **(2015)**, pp. 132-137.
[8] T. Jun and Z. Ning, "A collaborative filtering recommendation algorithm based on user interest classification. Computer system application", vol. 05, **(2011)**, pp. 55-59.
[9] X. Hong, P. Li, G. Aiyin and X. Yunjian, "Research on the improvement of collaborative filtering strategy based on user interest", Computer technology and Development, vol. 04, **(2011)**, pp. 73-76.
[10] D. Ailin, Z. Yangyong and S. Bole, "Recommendation algorithm of collaborative filtering algorithm based on project score prediction", Journal of software, vol. 14, no. 9, **(2003)**, pp. 1621-1628.
[11] W. Ting, "Application and research of collaborative filtering technology in the electronic commerce recommendation system", Wuhan: Wuhan University of Technology, **(2009)**.
[12] W. Huiping, "Based on item category similarity and the user interest in personalized recommendation algorithm research", Shanxi: Taiyuan University of technology, **(2008)**.
[13] J. Xiaosheng, L. Yanbing and L. Laiming, "Method of similarity measurement based on user interest in collaborative filtering. Computer application", vol. 30, no. 10, **(2010)**, pp. 2618-2610.

[14] Y. Fang and L. Jie, "An improved collaborative filtering recommendation algorithm", Journal of Hebei University of Technology, vol. 39, no. 3, **(2010)**, pp. 82-87.

Author



Shulin Liu, he received his M.S degree from Northeast Normal University. He is a lecturer in Hainan College of Economics and Business. His research interests include the sensor network.