

## The Principle and Improvement of the Algorithm of Matrix Factorization Model based on ALS

---

**Xiaoxuan Liang**<sup>12</sup>

*Yunnan University, Kunming, 65000, China*

*E-mail: shaw.xuan820@gmail.com*

**Chao Yi**<sup>3</sup>

*Yunnan University, Kunming, 65000, China*

*E-mail: yichao@ynu.edu.cn*

**Ruoxue Wu**

*Yunnan University, Kunming, 65000, China*

*E-mail: 691068796@qq.com*

On the base of deep research of the collaborative filtering algorithm, the discussion and verification of the principle of ALS algorithm, finding the algorithm having some problems. According to one of these problems, efforts have been devoted to improve ALS algorithm. The accuracy of ALS algorithm is increased. Verification in Spark, and the result show that the accuracy of improved algorithm better than ALS.

*ISCC 2015*

*18-19, December, 2015*

*Guangzhou, China*

---

<sup>1</sup>This work is supported by Science Research Fund of Yunnan Education Department(2014Y011)

<sup>2</sup>Speaker

<sup>3</sup>Corresponding Author

## 1.Introduction

The recommendation system, can help the user to find the information valuable to the users, and also make the information to the users' sight. It is a tool which helps people to choose the most suitable information and make the information consumers and producers to achieve a win-win situation.[1] In 1990s, the recommendation system was first proposed. It was developed very fast during the past two decades. Especially in the field of e-commerce, information retrieval, online transactions has achieved, greater development. It differs from the searching engine that the users don't have to provide the clear demand. Only based on the analyses, it makes the model of the historical behavior of users' interests. Thus, It initiatively recommends personalized information which may satisfy the users' need to the users.[2-3]

Today, the recommendation system includes e-commerce, video sites and social networks. The application of personalized services is very broad. The basic principle of the recommendation system is collaborative filtering recommendation algorithm(abbreviation: collaborative filtering algorithm).

## 2.Collaborative Filtering Algorithm

The collaborative filtering algorithm is recommended for the current users by referring to the selection of other users who have the same or similar needs to the current users. It is mainly based on the record of the users' past behavior, and calculates the similarity between the user-users' interests preference so as, to find the neighboring-users who have interests similar to the current user's. Finally, in order to achieve the advices of similar users group among these neighboring-users', it generates the information which the current users are no aware of, but maybe interested in. Simply, the collaborative filtering algorithm is based on the community views for the current users to recommend potential projects.

Generally speaking, according to the data usage, the collaborative filtering algorithm can be divided as follows:[4]

### 2.1 Based on User(UserCF)

The principle of UserCF is based on users' similarity to infer the potential similarity between every user through each user's preferences for each product; then recommend information to the potential users via the similarity. Simply, UserCF is to recommend the items to the current users which other users show similar interests in.

### 2.2 Based on the Item(ItemCF)

The ItemCF infers the similarity between item-item through users' scores to different items. The information is recommended based on the similarity between item-item. Easy to say, it recommends items similar to the current users' favorite.

### 2.3 Based on the Model

The collaborative filtering based on the model is based on the sample of the user's preference information. Train a recommendation model predict, according to real-time user preferences, and finally, recommend the result to the current users.

There are three kinds of collaborative filtering based on the model:

- the nearest neighbor model: which is based on the distance
- Latent Factor Mode(SVD): which is based on the matrix factorization.
- Graph: which is based on the social network graph model.

### 3. Principle and Improvement Of the Algorithm of Matrix

#### 3.1 Factorization Model based On ALS

The principle of matrix factorization model is based on ALS

It is most widely used for collaborative filtering algorithm which is based on the matrix factorization model which uses the alternating least squares (ALS) method.

For a user item rating score data set, ALS will create a user \* item of m\*n of the 2D matrix(the number of users is m and the number of items is n) as shown in Table 1:

	Item1	Item2	Item3	Item4	Item5	Item6
User1				7		9
User2	1	4			8	
User3			5	2		4
User4	3	1			4	
User5			8	6	7	
User6	3			2		1

**Table 1:** Data Set of Users Score for Items

In the matrix, each row represents a user( $m_1, m_2, \dots, m_6$ ) each column represents an item( $m_1, m_2, \dots, m_6$ ). The users' score is between 1-9 (1 represents the worst, and 9 represents the best. Empty represents no score for the item)

Due to the data set, each user doesn't score each item; so the matrix is often sparse. The score could be empty that user i scores for Item j.

But by ALS, the sparse matrix will be filled by certain rule so that the score which any user scores for any item. (the score is also called the user i forecast forItem j).

There are steps of the specific rules and forecast:

Suppose m\*n of the score matrix be R, which can be approximately decomposed into  $U$   $i(V)^T$

$U$  means m\*d of users feature the vector matrix

$V$  means n\*d of items feature the vector matrix( $(V)^T$  represents the transpose of  $V$ )

$D$  means the number of features of the user or the item ( $D$  can be understood as: for example, a film, users can evaluate a movie from the director, starring, the plot, the effects of these 4 aspects; therefore,  $d$  is 4. In other words, each film has a fixed benchmark score from such 4 aspects), the  $d$  is less than the rank of  $R$  matrix.

The matrix  $I$  is composed of the number of items.  $n \times$  the number of eigenvalues is  $d$ . For matrix  $U$ , assume that for any user  $A$ . The user features the items of the comprehensive scores and the value of items have a certain linear relationship. It equals to ( $a_1 \times d_1 + a_2 \times d_2 + \dots + a_n \times d_n$ ), from the  $a_1$  to the  $a_n$  are users' eigenvalue, from  $d_1$  to  $d_n$  are items' eigenvalue (for example, if item is film,  $n$  is 4). If the matrix  $R$  hypothesis is true. A user's score for an item can be obtained by a row of the matrix  $U$  multiplying a column of the matrix  $V^T$ .

For the eigenvalue of the user and the item is determined by alternating the least square method.

POS ( ISCG2015 ) 035

$$\sum_{(i,j) \in R} (a_{ij} - U_i^T V_j)^2 \quad (3.1)$$

In the formula,  $a$  is true score that user  $i$  scores Item  $j$  in the data set.  $U_i^T V_j$  means that the transpose of the user  $i$ 's feature vector multiplying the transpose of the Item  $j$ 's feature vector. The result is the prediction score which  $i$  scores for  $j$ . Then, using the reality score to minus the prediction score, next, square, last, record the result. Repeat the above steps as to the next user and item; finally, add all results. As it is unknown for the feature vector of users and items. Formula (3.1) is changed into (3.2).

$$L(U,V) = \sum_{ij} (R_{ij} - U_i V_j^T)^2 \quad (3.2)$$

In order to prevent over fitting, the formula is coupled with the regularization parameter. And thus get Formula (3.3).

$$L(U,V) = \sum_{ij} (R_{ij} - U_i V_j^T)^2 + \lambda (\|U_i\|_{\square}^2 + \|V_j\|_{\square}^2) \quad (3.3)$$

At the same time, fix  $V$ , and then use  $\frac{\partial L(U,V)}{\partial U_i} = 0$  and compute the partial derivative of  $U_i$  get Formula (3.4).

$$U_i = R_i V_{ui} (V_{ui}^T V_{ui} + \lambda n_{ui} E)^{-1} \quad i \in [1, m] \quad (3.4)$$

In Formula(3.4),  $R_j$  is the vector of the user  $i$  who has scored for items' scores.  $V_{ui}$  represents the feature matrix of the feature vector of user  $i$  who has scored for items.  $n_{ui}$  represents the number of the user  $i$  who has scored for items.  $E$  is a unit matrix of  $d \times d$ .

In the same way, the fixed  $U$  can be obtained by the solving Formula(3.5) of  $I_j$ .

$$V_j = R_j^T U_{mj} (U_{mj}^T U_{mj} + \lambda n_{mj} E)^{-1} \quad j \in [1, n] \quad (3.5)$$

In (3.5),  $R_j$  represents that the vector of scores which users scored for item  $j$ .  $U_{mj}$  represents that the feature matrix of the feature vector of users who scored Item  $j$ .  $n_{mj}$  indicates the number of users who scored Item  $j$ .

First, the  $V$  is initialized with a random number which is less than 1 (usually set to 0.01). Then, get  $U$  by Formula(3.4). Now, the initial  $UV$  matrix is obtained. Compute the previous sum of the squares of the differences. Compute and recover  $V$ , according to  $U$ ; through computation, Formula(3.5) applies. Last, compute the sum of squares of the differences again.

Repeat the above steps until the sum of the squares of the differences is less than a preset number, or the number of iterations (usually set to 10~20) to meet the requirements, then stop. To get the latest UV matrix, finally, sparse matrix R can be re replaced by  $R=U \times (V)^T$ . [5]

The algorithm and the idea is the product of the sparse matrix factorization into users feature vector matrix and items feature vector matrix. Then, by using the least square method, the feature vector of the users / items is gradually calculated, which makes the sum of the squares of the differences minimum. Finally, the matrix of users / items feature vector is used to predict the score of an item.

### 3.2 Improvement Of ALS Algorithm

As there are some information will be lost when computing the matrix U and V, such as the similarity between users and items, the traditional ALS algorithm and the similarity between the user ratings matrix has a certain error. Thus this paper proposes an improved algorithm based on the similarity of the ALS algorithm(ALS-S), which can eliminate the error and keep the similarity of the two.

Steps are shown as follows:

First of all, matrixing of the users rating data.

Second, use the cosine similarity measure method, use Formula (4.1) and (4.2) to calculate the similarity between user-user and item-item. [6]

$$sim(U_i, U_j) = \frac{\sum_{r \in V} (v_{i,r} - \bar{v}_i)(v_{j,r} - \bar{v}_j)}{\sqrt{\sum_{r \in V} (v_{i,r} - \bar{v}_i)^2} \sqrt{\sum_{r \in V} (v_{j,r} - \bar{v}_j)^2}} \quad (4.1)$$

$$sim(V_i, V_j) = \frac{\sum_{u \in K} (v_{ui} - \bar{v}_u)(v_{uj} - \bar{v}_u)}{\sqrt{\sum_{u \in K} (v_{ui} - \bar{v}_u)^2} \sqrt{\sum_{u \in K} (v_{uj} - \bar{v}_u)^2}} \quad (4.2)$$

In Formula (4.1),  $\bar{v}_i$  and  $\bar{v}_j$  represent the average score which user i and j scored as to items that users i and j have scored.  $v_{i(j),r}$  represents the score which user i(j) scored as to Item r. In Formula (4.2), K represents the set all of users who scored for both item i and j.  $\bar{v}_u$  represents the average score of item u.  $v_{ui(j)}$  is the score which user i and j scored as to Item u.

Add the similarity data to Formula (3.3) and get Formula (4.3).

$$L(U,V) = \sum_{ij} (R_{ij} - U_i V_j^T)^2 + \lambda (\|U_i\|_{\square}^2 + \|V_j\|_{\square}^2) + \sum_i (U_i - \frac{\sum_{p \in NN(U_i)} S_{U_i p} U_p}{\sum_{p \in NN(U_i)} S_{U_i p}})^2 + \sum_j (V_j - \frac{\sum_{q \in NN(V_j)} S_{V_j q} V_q}{\sum_{q \in NN(V_j)} S_{V_j q}})^2 \quad (4.3)$$

In (4.3), NN (Neighbor-N) ( $U_i$ ) indicates that user  $U_i$ 's neighbors which quantity is n. NN( $I_j$ ) represents that item  $V_j$ 's neighbors which quantity is n.  $S_{i,p}$  represents the similarity between user-user.  $S_{j,q}$  represents the similarity between item-item.

Then, continue to fix V, and compute the partial derivative of U. Next, fix U, and compute the partial derivative of V, get to (4.4) and (4.5):

$$U_i = R_i V_{ui} (V_{ui}^T V_{ui} + \lambda n_{ui} E)^{-1} + \frac{\sum_{p \in NN(U_i)} S_{U_i p} U_p}{\sum_{p \in NN(U_i)} S_{U_i p}} \quad i \in [1, m] \quad (4.4)$$

$$V_j = R_j^T U_{mj} (U_{mj}^T U_{mj} + \lambda n_{mj} E)^{-1} + \frac{\sum_{q \in NN(V_j)} S_{V_j q} V_q}{\sum_{q \in NN(V_j)} S_{V_j q}} \quad j \in [1, n] \quad (4.5)$$

Repeat these steps until the sum of the squares of the differences is less than a preset number, or the number of iterations to meet the requirements, then stop. Finally, get the latest UV matrix and also use  $R=U \times (V)^T$  to replace the original matrix.

With the improvement of ALS algorithm, the error is eliminated, and the similarity is maintained. The accuracy of the recommendation algorithm is higher.

#### 4. Experiment Results And Analysis

In order to guarantee the objectivity and comparability of the experiment, use MovieLen (1M) as the experiment data source(<http://grouplens.org/datasets/movielens/>). As to the experiment environment, Spark1.4.0 pseudo distributed environment is built in Macbook Pro Retina. The operating system is OS MAC 10.10 system. The programming environment is IntelliJ IDEA 14.14.

Using the RMSE(Root Mean Square Error) to judge the accuracy of the algorithm.

$$RMSE = \sqrt{\frac{1}{T} \sum (r - \hat{r})^2} \quad (5.1)$$

In the formula, T is the number of test score, and  $\hat{r} = U^T V$  is the value of predictive score. If the result is smaller, the accuracy rate is higher.

Name	Ranks 4		8		10	
	10	20	10	20	10	20
ALS	0.91145	0.90012	0.90232	0.89732	0.89874	0.89297
ALS-S	0.90056	0.89322	0.89763	0.89215	0.89219	0.88421

**Table 2:** Value of RMSE of ALS and ALS-S

In the table, 10 and 20 are iteration numbers. 4, 8 and 10 are items' ranks. In the table, the optimized ALS algorithm can reduce the value of RMSE when the item's rank is the same. When the iteration number is increased, the value is decreased. Thus, considering the similarity factor before training can reduce the effect of error; in this sense, after training the matrix U or V is more close to the similarity of between user-user or item-item. Making the value of RMSE is decreased and the algorithm accuracy is more accurate.

#### 5. Problem Of Collaborative Filtering Algorithm based On ALS

Now, the collaborative filtering algorithm still has lots of problems.

- the number of items is numerous, but as a user could not buy or score all the items. They only buy items of small quantities resulting from the data sparse effect accuracy. At the same time, in practice it is likely that the lack of same items between user-user resulted in two very similar users cannot be recommended.
- Currently, the collaborative filtering algorithm efficiency is very low and it is bad for large scale computing.
- In reality, there are so many factors which affect the users. How to consider more factors in practice, and use more information to fully close to the user's real needs, they are problems to be solved at present.

## 6. Conclusion

This paper introduced the principle of collaborative filtering algorithm based on the user, item and ALS, and pointed out some shortcomings of the collaborative filtering algorithm based on ALS. Through experiment, use the value of RMSE to compare ALS and ALS-S. The results indicated that ALS-S has a higher accuracy rate than ALS. With the advent of the information age, how to improve the accuracy of the collaborative recommendation algorithm is still worth considering for us.

## References

- [1] L Xiang. *Recommender Systems in practice* [M]. Posts & Telecom Press, Beijing, 2012:1-4.( In Chinese)
- [2] Jannach D. *Recommender Systems*[M].Posts & Telecom Press, Beijing, 2013:23-25
- [3] Ge Y, Xiong H, Tuzhilin A, et al. *Cost-aware collaborative filtering for travel tour recommendations*[J]. ACM Transactions on Information Systems, USA, 2014, 32(1):4-4.
- [4] G Li, L Li. *Collaborative filtering algorithm based on matrix decomposition* [J].Computer Engineering and Applications, 2011, 47(30):4-7. (In Chinese)
- [5] H Wei. *Collaborative Filtering Algorithm in Recommendation Systems*[J]. Computer Era, 2009,11:16-17( In Chinese)
- [6] Sarwar B, Karypis G, Konstan J, et al. *Item-based collaborative filtering recommendation algorithms*[c]//Proceedings of the 10th international conference on World Wide Web. ACM, USA,2001:285-295.