Collaborative Filtering Recommendation Model Considering Integration of User Rating and Attribute Similarity

Tian Jiule*, Xu Hang

1Department of Logistics & Information Management, Zhuhai College of Jilin University, Zhuhai City
Guangzhou Province, ZipCode: 519041, China
2Department of Mechanical & Automotive Engineering, Zhuhai College of Jilin University, Zhuhai City
Guangzhou Province, ZipCode: 519041, China
*Corresponding author, e-mail: 353489177@qq.com

Abstract

Directed at the problem that the collaborative filtering algorithm tends to be subject to data sparsity and cold start, a collaborative filtering recommendation algorithm based on improvement nearest neighbor is proposed. Firstly, the current user's k nearest neighbor and reverse k nearest neighbor are obtained on the basis of the similarity algorithm, which are used to compute positive and negative credibility values respectively based on their predicted ratings and the current user ratings. Then modifications of constraint are made for the users who are both the k nearest neighbor and the reverse k nearest neighbor and the hot resources. Finally, the collaborative filtering recommendation algorithm based on weight fusion is derived and a comparative experiment of simulation is conducted on MovieLens. The result shows that the algorithm in the Thesis decreases the mean absolute error value while improving the accuracy of recommendation.

Keywords: Collaborative filtering; Similarity; User rating; Sparse representation

1. Introduction

With the development of e-commerce, service providers must clearly grasp users' needs and preferences and provide users with services to their satisfaction in the premise that product quality is guaranteed. Therefore, excellent recommendation systems are very significant in e-commerce application and also become one of the researchers' focuses of attention [1, 2]. Basic thought of collaborative filtering algorithm is to judge whether the evaluation is of value to target user through evaluation of other users having similar interest preference with target user on target item and then to decide whether to recommend target item to target user further. Nearest neighboring collaborative filtering recommendation is the most successful recommendation technology at present and the algorithm recommends items to target user based on rating data of nearest neighbor having similar rating. As nearest neighbor's rating on item is similar to that of target user, target user's rating on unrated item can be predicted through nearest neighbor's rating on the item with weighting method. As collaborative filtering algorithm is usually influenced by data sparsity and cold start, a collaborative filtering recommendation algorithm based on improvement nearest neighbor is proposed. Firstly, the current user's k nearest neighbor and reverse k nearest neighbor are obtained on the basis of the similarity algorithm, which are used to compute positive and negative credibility values respectively based on their predicted ratings and the current user ratings. Then modifications of constraint are made for the users who are both the k nearest neighbor and the reverse k nearest neighbor and the hot resources. Finally, the collaborative filtering recommendation algorithm based on weight fusion is derived and a comparative experiment of simulation is conducted on MovieLens. The result shows that the algorithm in the Thesis decreases the mean absolute error value while improving the accuracy of recommendation [3-8].

Received February 2, 2016; Revised April 28, 2016; Accepted May 15, 2016
2. Traditional Collaborative Filtering Algorithm

2.1. Establish User-Item Rating Matrix

In traditional collaborative filtering algorithm, a p×q order matrix M containing “user-item” rating data will be established according to historical behavior data of user. It contains set of p users \(U=\{U1, U2, \ldots, Up\}\) and set of q items \(I=\{I1, I2, \ldots, Iq\}\). Number value of each point in the matrix is rating value of corresponding item of user (Table 1) [9].

<table>
<thead>
<tr>
<th>(U1)</th>
<th>(I1)</th>
<th>(I2)</th>
<th>(\ldots)</th>
<th>(Iq)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S_{11})</td>
<td>(S_{12})</td>
<td>(\ldots)</td>
<td>(S_{1q})</td>
<td></td>
</tr>
<tr>
<td>(S_{21})</td>
<td>(S_{22})</td>
<td>(\ldots)</td>
<td>(S_{2q})</td>
<td></td>
</tr>
<tr>
<td>(\ldots)</td>
<td>(\ldots)</td>
<td>(\ldots)</td>
<td>(\ldots)</td>
<td>(\ldots)</td>
</tr>
<tr>
<td>(S_{p1})</td>
<td>(S_{p2})</td>
<td>(\ldots)</td>
<td>(S_{pq})</td>
<td></td>
</tr>
</tbody>
</table>

2.2. Calculate Similarity

After gaining “user-item” rating matrix, we need to use similarity measurement algorithm to get k nearest neighbor set \(U=\{U1, U2, \ldots, Up\}\) of current user \(Ua\), and similarity between user \(Ua\) and users in the set shall be sorted from big to small. Specific value of nearest neighbor k is determined by specific application. \(I_{ab}\) is used to express item set of rating records shared by user \(Ua\) and \(Ub\). Specific description of common-used similarity measurement method at present is as follows:

1. Standard Cosine Similarity: user’s rating is abstracted as vector in n dimensional item space and cosine values of vectors are used to measure similarity between corresponding users. Cosine similarity between user \(a\) and user \(b\) is:

   \[
   Sim(a, b) = \frac{\sum_{i \in I_{ab}} s_{ai} s_{bi}}{\sqrt{\sum_{i \in I_{ab}} (s_{ai})^2 \sum_{i \in I_{ab}} (s_{bi})^2}}
   \]

2. Constrained Cosine Similarity: in order to amend rating dimension differences of different users and reduce influence of such difference on acquisition of user preference, user’s average rating value on all other items is deducted in Constrained Cosine Similarity algorithm to relieve rating difference. Constrained Cosine Similarity is:

   \[
   Sim(a, b) = \frac{\sum_{i \in I_{ab}} (s_{ai} - \bar{s}_a)(s_{bi} - \bar{s}_b)}{\sqrt{\sum_{i \in I_{ab}} (s_{ai} - \bar{s}_a)^2 \sum_{i \in I_{ab}} (s_{bi} - \bar{s}_b)^2}}
   \]

   Where, \(\bar{s}_a\) and \(\bar{s}_b\) indicate average rating of user a and user b on all other resources respectively.

3. Pearson Correlation Coefficient: average rating value of current item shall be deducted from item set \(I_{ab}\) of sharing rating between user a and user b to amend calculation of similarity. Similarity calculation of Pearson Correlation Coefficient is as follows:

   \[
   Sim(a, b) = \frac{\sum_{i \in I_{ab}} (s_{ai} - \bar{s}_a)(s_{bi} - \bar{s}_b)}{\sqrt{\sum_{i \in I_{ab}} (s_{ai} - \bar{s}_a)^2 \sum_{i \in I_{ab}} (s_{bi} - \bar{s}_b)^2}}
   \]
2.3 Select Neighbor

For neighbor selection, k neighbor strategy is selected generally, namely the first kth users having the most similarity with current user shall be selected as neighbors. For an active user a, a neighbor set \(U = \{u_1, u_2, \ldots, u_n\}, a \in U\) shall be generated with similarity sorted from big to small.

In \(k\) nearest neighbor, positive credibility can be gained through the difference value between rating of \(k\) nearest neighbors on certain item and current user’s predicted rating on the item. Set \(I_{ab}\) as sharing rating item set between user \(U_b\) and user \(U_a\) is one of the \(k\) nearest neighbors of user \(U_b\), \(U_a \in kNN(b)\). Predicted rating of user \(U_b\) on item \(I_i\) \(\in I_{ab}\) shall be gained as follows:

\[
P_{bi} = \frac{\sum_{u \in kNN(b)} (S_{iu} - \bar{S}_a) \cdot \text{sim}(b, a)}{\sum_{u \in kNN(a)} |\text{sim}(b, a)|}
\]

Where, \(\bar{S}_a\) and \(\bar{S}_b\) represent average rating of user \(U_a\) and user \(U_b\) on all other items respectively; \(S_{iu}\) represents rating value of user \(U_a\) on item \(I_i\) and \(\text{sim}(b, a)\) represents the similarity between user \(U_b\) and user \(U_a\) [10].

3. Collaborative Filtering Algorithm of the Thesis

3.1. Problem Analysis

Above recommended algorithms use influences of \(k\) nearest neighbor on current users to gain interested items of current users without consideration of influence relation between reverse \(k\) nearest neighbor and current users and the influence of user’s credibility on preference acquisition expressed by user’s behavior. Therefore, for cold start and data sparsity problems, it is difficult for above algorithms to get high quality recommendation results.

3.2. Acquisition of Reverse nearest Neighbor

Set \(k\) nearest neighbor of user \(U_u\) is \(\{U_1, U_2, \ldots, U_p\}\) and definition for RkNN (Reverse k Nearest Neighbor) of user \(U_u\) is: set of all users taking user \(U_u\) as \(k\) nearest neighbor in related user groups. If \(kNN(m)\) is used to represent \(k\) nearest neighbor set of user \(U_m\), formalized definition for reverse \(k\) nearest neighbor of user \(U_u\) is as follows:

\[
\text{RkNN}(u) = \{U_u \in U \mid U_u \in kNN(m)\}
\]

Its graphic description is shown in Figure 1.

![Figure 1. Graphic description of reverse nearest neighbor and nearest neighbor](image)
3.3. Constraint Model for Credibility between Users

Credibility shall be introduced into recommendation system and its specific definition is: a kind of subjective recognition degree of users to be recommended on authenticity and reliability of \( k \) nearest neighbor and reverse \( k \) nearest neighbor selected by system. It is a kind of “user-user” and “user-group” subjective credibility relation. Introduction of credibility is a kind of punitive correction on similar users actually and it aims to weaken influence of unrelated users on recommend results in \( k \) nearest neighbor under data sparsity condition. In the Thesis, position credibility and negative credibility are introduced to correct \( k \) nearest neighbor and reverse \( k \) nearest neighbor respectively.

After predicted rating of current user \( U_b \) on item \( i \) is gained according to equation (4), computational formula of positive credibility between user \( U_b \) and user \( U_a \) is:

\[
T_{ba} = \frac{[P_{ba} - S_{ai}](k-1)}{\sum_{a \in kNN(b), a \neq i} S_{ai} - P_{ai}}
\]  

Where, \( P_{bi} \) is predicted rating of user \( U_b \) on item \( i \); \( S_{ai} \) is actual rating of user \( U_a \) on item \( i \) and \( S_{ui} \) is actual rating of other \( k \) nearest neighbor user \( U_u \) on item \( i \). The ratio between the difference value between predicted rating \( P_{bi} \) and rating of user \( a \) on item \( i \) and rating difference value of other users in \( k \) nearest neighbor of user \( U_b \) shall be deemed as positive credibility value between user \( U_b \) and user \( U_a \).

In \( k \) reverse nearest neighbor, calculation of negative credibility is contrary to that of positive credibility. The difference value between current user’s rating on certain item and predicted rating of reverse \( k \) nearest neighbor on the item shall be the basis for calculation of negative credibility. Calculation formula is:

\[
RT_{ba} = \frac{[S_{bi} - P_{ai}](k-1)}{\sum_{a \in kNN(b), a \neq i} S_{bi} - P_{ai}}
\]

Where, \( S_{bi} \) is actual rating of user \( U_b \) on item \( i \) and \( P_{ai} \) is predicted rating of user \( U_a \) on item \( i \).

Strengthening and weakening relationship between positive credibility and negative credibility shall be determined according to the corresponding rating difference. As some particular users and items shall be taken into consideration during calculation of credibility, in order to correct credibility among users, the Thesis introduces credibility constraint model. The model is divided into two parts:

1. Influence of user which is both \( k \) nearest neighbor and reverse \( k \) nearest neighbor on current user shall be greater than that of \( k \) nearest neighbor user or reverse \( k \) nearest neighbor user due to asymmetry of \( k \) reverse nearest neighbor. Therefore, credibility enhancement model is used to amplify its influence on current user. Namely for any user \( U_a \), \( U_a \in RkNN(b) \) and \( U_a \in kNN(b) \), enhancement relationship is

\[
\begin{align*}
T_{ba} &= T_{ba} \sqrt{1 + (P_{ba} - S_{ai})^2} \\
RT_{ba} &= RT_{ba} \sqrt{1 + (S_{bi} - P_{ai})^2}
\end{align*}
\]

Difference between actual rating of user \( U_a \) on item \( i \) and predicted rating of user \( U_b \) on item \( i \) is amplified to enhance credibility value.

2. Users are very likely to gain popular resources in other e-commerce platforms or through other approaches. Therefore, probability for popular resources to appear in recommendation list shall be reduced so as to reduce Matthew Effect of recommended results. Weakening relationship is as follows:
Collaborative Filtering Recommendation Model Considering Integration of User … (Tian Jiule)

\[
T_{ba} = T_{ba} \left( 1 - \frac{\sqrt{1 - (P_{ba} - \bar{S}_a)^2}}{P_{ba} - \bar{S}_a + \lambda} \right)
\]

\[
RT_{ba} = RT_{ba} \left( 1 - \frac{\sqrt{1 - (S_{bi} - P_{ai})^2}}{S_{bi} - P_{ai} + \lambda} \right)
\]  

(9)

Difference between predicted rating of user \( U_a \) on item \( I_i \) and actual rating of user \( U_b \) on item \( I_i \) shall be used to weaken the credibility value. Where, \( \lambda \) is the corrected parameter which prevents denominator to be 0. For judgment of popular resources, the Thesis adopts the following strategies. Set \( H \) is popular resources set at present, for any user \( I_i \):

\[
I_{r_i} \geq (\overline{I_r} + t) \Leftrightarrow I_i \in H
\]  

(10)

Where, \( I_{r_i} \) is quantity of users having behavior records on item \( I_i \); \( \overline{I_r} \) is average quantity of users having behavior records on other items; \( t \) is corrected parameter for different application environment.

3.4. Obtaining Final Preference of Weighted Fusion

Based on \( k \) nearest neighbor and reverse \( k \) nearest neighbor and positive and negative credibility, conduct weighted fusion for them to obtain final derivatives.

\[
P_{_{final(bi)}} = \alpha \left( 1 + \frac{\sum_{m \in \text{NN}(b)} ((s_{mi} - \bar{s}_m)T_{im})}{k} \right) \bar{x}_i + \beta \left( 1 + \frac{\sum_{m \in \text{NN}(b)} ((s_{mi} - \bar{s}_m)RT_{im})}{k} \right) \bar{x}_i
\]  

(11)

where, \( \bar{s}_i \) represents average score of the existing score values of item \( I_i \). Parameters \( \alpha \) and \( \beta \) are weight value parameters and \( \alpha + \beta = 1 \).

Two functions of introduced parameters are: (1) modify weight value proportion of nearest neighbor and reverse nearest neighbor to adapt to different demand to apply or not to apply data set; (2) prevent final evaluation score from exceeding upper limit of initial score value. Two sub-algorithms in equation (11) can be described as influence of product of the sum of score of item not yet scored by \( k \) (reverse) nearest neighbor and average score deviation of the user multiplying by positive (negative) credibility relative to the current user on average score of item not yet scored.

4. Simulation Experiment

4.1. Simulation Environment

Intel (R) Core (TM)2 Duo CPUE7400 with dominant frequency of 2.8GHz, memory of 2G and hard disk of 320G. Operating system is Windows professional sp3 realized by Visual C++ language. adopt public data set MovieLens adopting common measurement and recommended algorithm accuracy as simulation object to select public partial data sets, which contains more than 100000 true score records from 943 users for 1682 films.

4.2. Result and Analysis

(1) Influence of Selection of \( \alpha \) and \( \beta \) Parameters on Experiment Result: See values of different \( \alpha \) and \( \beta \) parameters in Table 2. Conduct 5 calculations for every node for values of \( \alpha \) and \( \beta \) and then use its average as currently predicted accuracy value. Finally, adopt three kinds of similarity measurement methods of \( \text{Pearson} \) relevancy and similarity, cosine similarity, and
modified cosine similarity to conduct analysis experiment, the result of which is shown in Figure 2-4.

Table 2. Parameter values matrix

<table>
<thead>
<tr>
<th>Parameter</th>
<th>(\alpha)</th>
<th>(\beta)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence 1</td>
<td>0.1</td>
<td>0.9</td>
</tr>
<tr>
<td>Sequence 2</td>
<td>0.3</td>
<td>0.7</td>
</tr>
<tr>
<td>Sequence 3</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Sequence 4</td>
<td>0.7</td>
<td>0.3</td>
</tr>
<tr>
<td>Sequence 5</td>
<td>0.9</td>
<td>0.1</td>
</tr>
</tbody>
</table>

From Figure 2-4, at the time of adopting cosine similarity calculation method, recommended accuracy and stability of algorithm are optimal. With increasing of the nearest neighbor points, values of MAE indicates the trend of increase followed by decrease and the nearest neighbor obtains overall optimal value at 30-35. With increasing value of parameter \(\beta\), namely the reverse \(k\) nearest neighbor ratio, in selection of different \(k\) values, the value of MAE indicates overall decline trend in first 4 sequences and begins to increase in the fifth group. It can be found after comprehensive analysis, overall MAE value is the smallest and recommended accuracy is the highest at the time of sequence 4 under cosine similarity. Therefore, in the experiment below, measurement method of cosine similarity is adopted with \(\alpha=0.7\) and \(\beta=0.3\).

Figure 2. Experiment result of parameter selection (cosine similarity)

Figure 3. Experiment result of parameter selection (modified cosine similarity)
Figure 4. Experiment result of parameter selection (pearson relevancy and similarity)

(2) Accuracy Comparison with Other Algorithms: In order to make algorithm results of the Thesis to be more comparable, select algorithms for comparison experiment. There are many evaluation indexes in currently recommended system, of which the most commonly used ones are mean absolute error (MAE) and root mean square error (RMSE). MAE (u) equals to the average of the sum of absolute values of true score values in predicted value of target user u and test set. MAE of the whole recommended algorithm is the average of MAE of all users. Details are as follows:

\[
MAE(u) = \frac{1}{n} \sum_{i=1}^{n} |r_{ui} - \hat{r}_{ui}| / n \tag{12}
\]

\[
MAE = \frac{\sum_{u=1}^{N} MAE(u)}{N} \tag{13}
\]

Where, n is the number of the existing score numbers of target user in test set.

RMSE is more sensitive to large errors. RMAE (u) equals to the root square error of the sum of squares of deviation of true score values in predicted value of target user u and test set. RMAE of the whole recommended algorithm is the average of RMAE of all users. Details are as follows:

\[
RMSE(u) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (r_{ui} - \hat{r}_{ui})^2} / n \tag{14}
\]

\[
RMSE = \frac{\sum_{u=1}^{N} RMSE(u)}{N} \tag{15}
\]

Different algorithm experiment results are shown in Figure 5. From Comparison Figure 5, compared to reference algorithm, the algorithm recommended in the Thesis is higher in accuracy so as to obtain better recommended quality. Comparison results show algorithm of the Thesis has fused influence of credibility between reverse k nearest neighbor and users and modify the credibility to obtain more ideal recommended result.
5. Conclusion

As for the problem the current collaborative filtering algorithm is largely affected by data sparseness degree, a recommended collaborative filtering algorithm with improved nearest neighbor is proposed. Simulation experiment result shows algorithm in the Thesis has improved recommended accuracy and recommended quality to better solve defects in traditional collaborative filtering algorithm with nearest neighbor, thus significantly improving efficiency of algorithm.

References


