Collaborative Filtering Approach based on Item and Personalized Contextual Information

Feng Jiang1,2, and Min Gao3
1College of Civil Engineering, Chongqing University, Chongqing, China
2Construction Engineering Department, Chongqing Technology and Business Institute, Chongqing, China
Email: jiangfeng@cqu.edu.cn
3College of Computer Science, Chongqing University, Chongqing, China
Email: gaomin@cqu.edu.cn

Abstract—In order to improve the precision of rating prediction for personalized recommendation online, an approach incorporating personalized contextual information in item-based collaborative filtering is proposed. In this paper we analyze how to learn personalized contextual information and predict ratings for unknown items based on the well-known SlopeOne item-based collaborative filtering. Finally, we experimentally evaluate our results and compare them to the basic Slope One approach. Our experiments suggest that our algorithm provide better quality than Slope One algorithm.

Index Terms—Collaboration Filtering, Context, Recommendation, Personalization

I. INTRODUCTION

Due to the explosive growth of the Web, personalized recommendation systems have been widely accepted by users. Personalized recommendation is defined as automatic adjustment, re-structuring, and the presentation of tailored information content for individuals [1]. It is a process of gathering and analyzing user information for delivering the right information at the right time [2].

Collaborative filtering is the most popular technique in recommender systems. However, traditional collaborative filtering suffers scalability problem [3-5]. It has been proved that Item-based collaborative filtering can solve the problem. Item-based collaborative filtering computes item similarity for prediction. It has been shown to produce recommendation results that are comparable to traditional collaborative filtering, but the results have not been distinctly improved [4].

In this paper, we incorporate personalized contextual information into item-based filtering approach to improve the recommendation results. Contextual information has been recognized as important factor in recommender systems [6, 7]. Personalized contextual information is also significant factor. For example, in a news recommender system, time and place are usually important contextual information to determine what news should be recommended, but they are not that useful for some users working at SOHO (small office/home office). So analyze personalized contextual information for recommendation is significant.

In the paper we first review collaborative filtering approaches including Slope One (Section 2), and then propose a context-based item difference analysis, a personalized contextual information analysis, and a rating estimating method based on Slope One in Section 3, and at last experimentally evaluate our approach and compare the results to the Slope One in Section 4.

II. BACKGROUND OF ITEM-BASED COLLABORATIVE FILTERING

In a typical collaborative filtering scenario, there is a rating matrix which includes a list of m users (rows) and a list of n items (columns) and lots of ratings r_{ui}. For example, table 1 is a rating matrix RM. There are 5 users, 6 items, and several ratings in the matrix. r_{u3,i4}=Ø means the item i is not rated by the corresponding user u. A rating r_{u1,i1} means how the user u likes the item i. The key step of it is to extrapolate unknown ratings.

<table>
<thead>
<tr>
<th></th>
<th>i1</th>
<th>i2</th>
<th>i3</th>
<th>i4</th>
<th>i5</th>
<th>i6</th>
</tr>
</thead>
<tbody>
<tr>
<td>u1</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>Ø</td>
</tr>
<tr>
<td>u2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>Ø</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>u3</td>
<td>Ø</td>
<td>4</td>
<td>Ø</td>
<td>2</td>
<td>3</td>
<td>Ø</td>
</tr>
<tr>
<td>u4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>u5</td>
<td>Ø</td>
<td>Ø</td>
<td>3</td>
<td>Ø</td>
<td>2</td>
<td>Ø</td>
</tr>
</tbody>
</table>

The traditional (also user-based) collaborative filtering is to predict ratings for a user based on the opinions of similar users. It was very successful in past, but some challenges have been revealed [8] such as scalability. Scalability means its computation grows rapidly with users and items increasing. It has been proved that item-based collaborative filtering can solve the problems. Item-based collaborative filtering is to compute the similarity between items and then to select the most similar items for prediction. There are several ways to compute the similarity between items and to estimate prediction.
The Slope One is a typical item-based collaborative filtering approach. It was proposed at February, 2005 by Lemire and Maclanchian. It works on comparing the intuitive principle of a popular differential between items [9]. It computes the deviation between items rather than similarity. The deviation of item \( i \) and \( j \) \( d_{i,j} \) is computed by the average difference between item arrays of \( i \) and \( j \) (See formula 7). In turn, the deviation of items will be used to predict an unknown item, given their ratings of the other. The prediction is based on a linear regression model of the known ratings and deviations. It has won the wide attention because it is simple and efficient, but the results have not been distinctly improved [4].

The Slope One does not take additional personalized contextual information into consideration. However, personalized contextual information is important for recommendation. Because a user usually has different preferences in different context and some context is important for him/her, but others are not. Personalized contextual information includes the contextual parameter that the user is most sensitive to. For instance, in the case of news delivery system, personalized contextual information can be \{time\}, \{place\}, or \{time and place\}.

In order to provide more accurate recommendations, we propose a personalized contextual information-based collaborative filtering.

III. PERSONALIZED CONTEXTUAL INFORMATION-BASED COLLABORATIVE FILTERING APPROACH

Contextual information in recommendation systems is additional information, besides information on Users and Items. It is relevant in identifying pertinent subsets of data, building richer rating estimation models, and providing various types of constraints on recommendation outcomes [7]. Personalized contextual information for a user is the contextual parameter that the user is sensitive to.

In this section we analyze context-based item differences and personalized contextual information for rating estimation [10].

Phase 1. Context-based Item Difference Matrix (Offline)

Context-based item difference matrix includes all deviations between items in a certain context.

Step1. Given a training dataset \( D \), we extract subset \( \{D_1, D_2, ..., D_n\} \) in different context \( \{C_1, C_2, ..., C_n\} \) from \( D \). Every \( D_k \) includes the data belongs to the context \( C_k \).

For example, suppose \( RM \) is a dataset \( D \), Table 2 is a subset \( DC_j \) of its.

<table>
<thead>
<tr>
<th>( i_1 )</th>
<th>( i_2 )</th>
<th>( i_3 )</th>
<th>( i_4 )</th>
<th>( i_5 )</th>
<th>( i_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_1 )</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>( u_2 )</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>( \emptyset )</td>
<td>3</td>
</tr>
<tr>
<td>( u_3 )</td>
<td>( \emptyset )</td>
<td>4</td>
<td>( \emptyset )</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 2. An example subset \( D1 \) of \( RM \)

Step2. For every two items \( i \) and \( j \) of \( D_k \), compute their difference (See formula 1). Here \( U(i) \) includes users who have rated on \( i \). The entire \( d_{i,j} \) forms the IDM for \( C_k \).

\[
d_{i,j} = \sum_{u \in U(i) \cap U(j)} \frac{(r_{u,i} - r_{u,j})}{|U(i) \cap U(j)|}.
\]

Phase 2. Personalized Contextual Information Analysis (Offline)

The purpose of this procedure is to get personalized contextual information for every user.

Step1. On each subset \( D_k \) to run a collaborative filtering algorithm \( A \) and calculate its performance \( PA(u,C_k) \) for every user and context. All these performances comprise a performance matrix. It includes users as rows and contexts as columns.

Given \( P \) is a evaluation metric, we calculate \( A \)'s performance \( P_{A}(u,C_k) \) for every user in every context. All these performances for users and contexts comprise a performance matrix (PM). PM includes users as rows and contexts as columns.

\[
PM = \begin{pmatrix}
P_{A(1,d_{1})} & P_{A(1,d_{2})} & ... & P_{A(1,d_{6})} \\
P_{A(2,d_{1})} & P_{A(2,d_{2})} & ... & P_{A(2,d_{6})} \\
... & ... & ... & ... \\
P_{A(n,d_{1})} & P_{A(n,d_{2})} & ... & P_{A(n,d_{6})}
\end{pmatrix}
\]

For example, Table 3 is the \( PM \) of \( RM \) and its subset.

Table 3. An example of \( PM \)

<table>
<thead>
<tr>
<th>( C_1 )</th>
<th>( C_2 )</th>
<th>( C_3 )</th>
<th>( C_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_1 )</td>
<td>0.85</td>
<td>0.72</td>
<td>0.64</td>
</tr>
<tr>
<td>( u_2 )</td>
<td>0.73</td>
<td>0.83</td>
<td>0.76</td>
</tr>
<tr>
<td>( u_3 )</td>
<td>0.68</td>
<td>0.74</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Step2. To find the best performance value in each row in the PM.

Figure 1. Performance results on the item size.
The column with the best value is the personalized contextual parameter for the user. Personalized contextual information will determine which difference matrix will be used in rating prediction.

Given that the lower the PM value, the better the recommendation results are, then the personalized contextual parameters for all users are as following table.

<table>
<thead>
<tr>
<th>User</th>
<th>Personalized Context Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>u1</td>
<td>C3</td>
</tr>
<tr>
<td>u2</td>
<td>C1</td>
</tr>
<tr>
<td>u3</td>
<td>C1</td>
</tr>
</tbody>
</table>

Phase3. Personalized Contextual Information-based Prediction (Runtime)

Step1. Given the target user \( u \), we get his/her personalized contextual information \( C_u \), and then get the DMeu.

Step2. According to the Slope One, \( d_{ui} \) is a prediction for \( r_{ui} \) according to \( r_{ui} \). So we calculate prediction for \( r_{ui} \) by formula 2. Here \( R_u \) is all known ratings of user \( u \). \( |R_u| \) is the number of ratings in \( R_u \). \( d_{ij} \) is the deviation in \( D_c \).

\[
p_{ui} = \frac{\sum_{i \in R_u} (r_{ui} - d_{ij})}{|R_u|} = r_i + \frac{\sum_{i \in R_u} d_{ij}}{|R_u|}. \tag{2}
\]

IV. EXPERIMENTAL EVALUATION

A. Data Set and Evaluation Metric

We used dataset from the well-known Movielens project (http://movielens.umn.edu). Movielens is a free service provided by GroupLens Research at the University of Minnesota. There are two datasets in the Movielens project. One includes 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 users who joined Movielens in 2000. Another is data sets consists of 100,000 ratings (1-5) from 943 users on 1682 movies. Each user has rated at least 20 movies.

We selected the second one as our dataset. The data set was divided into a training set (80% of the data) and a test set (20% of the data) five times. These training and test sets are named U1base, U2base, U3base, U4base, and U5base, and U1test, U2 test, U3test, U4test, and U5test respectively.

Without loss of generality, we used U2 to analyze personalized contextual information. And then we used others to evaluate our algorithm and Slope One. The contextual parameters here include age, gender, occupation, zip, and time (work time and rest time).

We used widely used metric MAE (Mean Absolute Error) to measure the deviation between predictions and ratings. For predictions \( \{p_1, p_2, \ldots, p_n\} \) and their true ratings \( \{r_1, r_2, \ldots, r_n\} \), MAE is the average absolute error of them (See formula 3). The lower the MAE, the better the recommendation approach is.

\[
MAE = \frac{\sum_{n} |p_i - r_i|}{N}. \tag{3}
\]

B. Experimental Procedure and Results

To compare our approach with Slope One and determine the sensitivity of the size of items, we performed the experiment where we computed MAE for different number of items. Our results are shown in Fig.1. In the figure, each left column is the experimental result of Slope One, and each right column is the experimental result of our algorithm. It can be observed that our algorithm out performs Slope One at all values of item size. Note that at item size of above 250, both algorithms perform worse. We believe this happens as the regression model suffers from data over fitting at high density levels.

V. CONCLUSION AND FUTURE WORK

Recommender systems help users find items they would be interested in. Currently, item-based collaborative filtering is most popular in recommender systems. Slope One is a well-known approach of them. In this paper we analyze how to learn personalized contextual information and predict ratings based on Slope One. Experimental results show that personalized contextual information is helpful to improve the prediction results of Slope One. One of the drawbacks of our approach is that we only choose one contextual parameter for a user and weight of it is not taken into consideration. In the future, more parameters and their weights [parameter, weight] pairs will be considered to achieve weighted personalized contextual information-based prediction.

ACKNOWLEDGMENT

We gratefully acknowledge valuable input from Tao Zhou, MSc student, from the School of Computer Science, Chongqing University, and Guoxi Cui, PhD candidate in Information Systems, from Informatics Research Center, the University of Reading.

REFERENCES


