Personalized Recommendation Based on Behavior Sequence Similarity Measures

Yuqi Zhang and Jian Cao*

Shanghai Jiaotong University, China
Zyuqi7@gmail.com, cao-jian@cs.sjtu.edu.cn

Abstract. Personalized recommendation is attracting more and more attentions nowadays. There are many kinds of algorithms for making predictions for the target users, and among them Collaborative Filtering (CF) is widely adopted. In some domains, a user's behavior sequences reflect his/her preferences over items so that users who have similar behavior sequences may indicate they have similar preference models. Based on this fact, we discuss how to improve the collaborative filtering algorithm by using user behavior sequence similarity. We proposed a new Behavior Sequence Similarity Measurement (BSSM) approach. Then, different ways to combine BSSM with CF algorithm are presented. Experiments on two real test data sets prove that more precise and stable recommendation performances can be achieved.

1 Introduction

With the development of technology, the Internet has penetrated into people's lives in all areas of study and work to develop the largest information database in today's world. Faced with such a large amount of information, how to make use of these data is becoming the focus of current research [Han et al 2011].

Personalized recommendation is a research field emerged with the increasingly sophisticated use of data mining techniques in recent years. It analyzes the preferences of users according to the user's access records to provide a personalized recommendation service to the user when they access the web site so that customers' satisfaction and loyalty can be improved [George et al 2007]. These systems have been successfully applied to various domains such as movies [Alspector et al 1997; Good et al 1999], news[Resnick et al 1994] , and online e-commerce, such as Amazon.com and eBay[Schafer et al 2001], and it brings a lot of benefits to the users and service providers.

There are various kinds of algorithms that have been applied to personalized recommendation problems, and Collaborative Filtering (CF) is a main one among these methods [Su et al 2009]. In general, CF algorithm uses a database of users' preferences over items to predict additional topics or products that the target user might like. There are mainly three categories of CF algorithms, which are memory-based, model-based and hybrid based ones [Su et al 2009]. Although CF algorithms show their advantages in many applications, they still have some drawbacks [Badrul et al 2001], and thus need to be improved.

---

* Corresponding author.

L. Cao et al. (Eds.): BSIC/BSI 2013, LNAI 8178, pp. 165–177, 2013.
© Springer International Publishing Switzerland 2013

In some domains, the sequence of a user's behaviors can reflect his/her preferences over items [Cao 2010]. Taking the following case as an example, there are some movies shown in recent three weeks (See Fig. 1). Movies of same letter belong to the same type. User u_1 watched movies in the sequence of $<a_1, b_1, c_1, a_2, b_2>$, whereas user u_2 watched movies in the sequence of $<b_1, a_1, b_2, a_2>$, and user u_3 watched movies in the sequence of $<c_1, a_1, b_1, a_2, b_2>$. The reasons leading to the various behavior sequences may differs between different people. However, the order of movies watched will undoubtedly be affected by user's personal interests. Therefore, it is possible to find users with similar interests in terms of their Behavior Sequence Similarity Measures (BSSM).

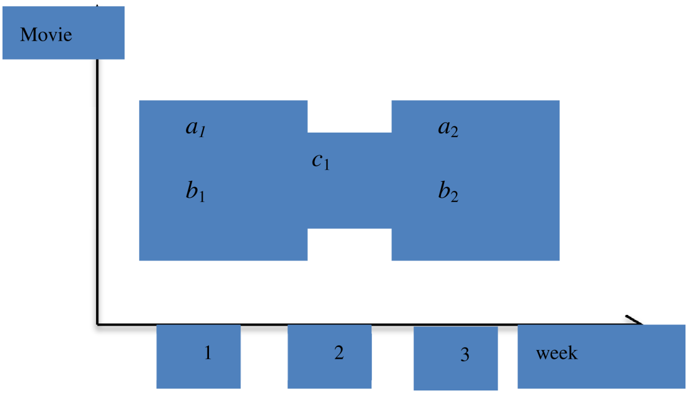

Fig. 1. Movie and Release Time

According to this fact, in this paper, we propose a model that combines BSSM with CF algorithm, and we present several ways to enhance the CF algorithm by applying BSSM.

The rest of the paper is organized as follows. In Section 2, we introduce the related work. In Section 3, we define the user behavior sequence and discuss how to select users who have similar preference with the target user based on BSSM. Then in Section 4 we propose several ways to combine the BSSM with the traditional CF algorithm. Experiments to evaluate the performance of our algorithm is demonstrated and discussed in Section 5. Finally, we conclude the paper and outline future research directions in Section 6.

2 Related Work

Recommendation systems have attracted many attentions nowadays in the field of web application system and on-line information retrieval systems. Traditionally, recommendation algorithms are partitioned into two main families: content-based filtering recommenders and collaborative filtering ones [Resnick et al 1997]. Content-based filtering recommenders make recommendation based on an evaluation of the user own past actions, as WebWatcher [Joachims et al 1997] and client-side agent Letizia [Lieberman et al 1995], while collaborative filtering is based on other similar users' preference. Our approach is an improvement to the traditional CF algorithm.

User behavior has become a research topic that tries to catch the sequence of user interactions at a higher level. These models will in general be hierarchical, as the workload requests at a lower level [Helmult et al 1999], such as WUM, which discover usage pattern from web log file and identify underlying user visit interest exhibited from user's navigational activity to satisfy the expert's criteria [Myra et al 1998], and LDA Model, which incorporating Web user access pattern based on Latent Dirichlet Allocation model to discover the associations between user sessions and multiple topics via probability inference so that to predict more preferable web pages for users via collaborative recommending technique [Guandong Xu et al 2008]. However, the research on user behavior mainly tries to predict future behavior for the original user. In this paper, we combine the user behavior analysis with the traditional CF algorithm to propose a new recommendation model, which can discover the user group that has similar preference with the target user and also take the advantages of a CF algorithm, so that the recommendation performance can be improved.

3 User Behavior Sequence Similarity Measurement

3.1 User Behavior Sequence

Based on social behavioristic theory, users' behaviors are influenced by users' objective preferences and subjective attributes such as age, occupation, area, etc, which has regularities that can reflect the characteristic and personal preferences of users. Based on this fact, users who have similar behaviors should have similar personal preferences.

User behavior sequence is the sequence of user's access behavior in the order of temporal precedence. Here we give the definition of user behavior sequence as follows.

Definition 1(User Behavior Sequence)**.**
Given a user u, whose access behaviors within a time window can be represented as a temporal sequence bs=<a_1, a_2, a_3,..., a_n>, where a_i is the action of user u, then we call the sequence bs the user behavior sequence for user u.

Based on the definition of user behavior sequence, we can discover the user behavior pattern, which can be defined as:

Definition 2(User Behavior Pattern)**.**
Given a user u whose user behavior sequence is bs=<a_1, a_2, a_3,..., a_n>, and a number $m(0 \leq m \leq n)$, then all distinct sub-sequences whose length is m within a user behavior sequence form the user behavior pattern set for user u.

Here, the similarity between two users is measured by the similarity between their corresponding behavior sequences. Given two user behavior sequences, and given a length for constructing user behavior pattern set, the Behavior Sequence Similarity Measurement (BSSM) is the intersection of the user behavior pattern sets of two users. If we denotes W_{ij} as the similarity of behavior sequences between user u_i and user u_j, then W_{ij} is calculated by

$$W_{ij} = max(\frac{simiCount}{P^m_{length\_i}}, \frac{simiCount}{P^m_{length\_j}}) \quad (1)$$

Here, *simiCount* is the number of common user behavior patterns in both sequences. *length_i* and *length_j* is the number of actions in user behavior sequences, and *m* is the length of user behavior pattern.

For example, if user u_i has the user behavior sequence $<a_1, a_2, a_3, a_4, a_5>$, user u_j has the user behavior sequence $<b_1, a_2, b_3, a_4, a_5, b_6>$, and given the length of user behavior pattern is 2, then the similarity between the user sequences of u_i and u_j is

$$W_{ij} = max\left(\frac{3}{P_5^2}, \frac{3}{P_6^2}\right) = 0.3 \tag{2}$$

There are many ways to measure the similarity between user behavior sequences, such as comparing the frequency of items appearing in the sequences or comparing the common single items appearing in the sequences. Here, we choose to measure the similarity between user behavior pattern sets to compare the similarity between different users, because we believe that this can reflect the similarity of preference structures of different users over items.

In order to verify this intuition, we did some experiments on the real data set. For example, we did experiments on the dataset of MovieLens. Fig. 2 shows the normalized watching frequency distribution for an identified user group over the top ten most-watched movies. It can be observed that there is a common preference among users in the same group. This fact tells us it's possible to make use of BSSM to find users with similar preferences.

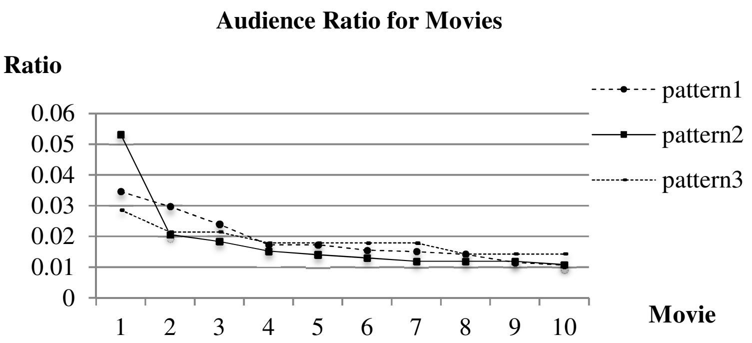

Fig. 2. Audience Ratio Distribution over Top Ten Movies

3.2 Analysis on the Length of User Behavior Pattern

BSSM relies on a parameter, i.e., the length of sub-sequence representing the user behavior pattern. Actually, when the length is 2, it can reveal the preference structure over any two items. When the length is 3, it reveals linear preference structure over any three items. In addition, this number also determines how many similar users could be found. When it is relatively short, the number of similar users will be relatively large. On the contrast, when this number is relatively large, the number of similar users will be relatively small. Obviously, the number of users in the group will influence the recommendation performance. Thus, we should find the best length for describing user behavior patterns.

In order to get reasonable result, an experiment dataset should have sufficient user numbers, so we first find out the movie that has been watched by most users, whose ID is 4196, recorded as M_1. In the first step (n=1), we select users who have watched movie 4196, recorded as $UserSet_1$. Then we treat $UserSet_1$ as dataset for the next experiment, and find out the movie that has been seen most times, which is movie 424, recorded as M_2. Next we select users who have seen M_3 in $UserSet_2$, recorded as $UserSet_2$, and record this step as n=2. Similarly, we can get M_3, which is movie 1088 and corresponding user set $UserSet_3$ for n=3. When performing on $UserSet_3$ and obtain $UserSet_4$, we notice that there is an obvious decrease in the number of users in this group, which makes it contributes little to the experiment, so we stop at n=3. At last we get three user behavior patterns, which are (4196) noted as *pattern1*, (4196, 424) noted as *pattern2*, and (4196, 424, 1088) noted as *pattern3*.

For every user behavior pattern, we first find out the next movie which has been seen just after seeing the movies in the user behavior pattern, noted as after-movie, and the movie set we get is noted as after-movie-set. Then we make statistic on the frequency of movies in after-movie set, and rank them from large to small, noted as movie-count-list. Since the numbers of each user set are not same, we transform the result into movie-ratio-list, which avoids the influence caused by number difference. We calculate movie-ratio by the formula below,

$$\text{ratio} = \frac{\text{Count(MovieN)}}{\text{Count(UserSet)}} \tag{3}$$

where count(*MoiveN*) indicates the frequency distribution of movies, and count(*UserSet*) indicates the number of user in the user set.

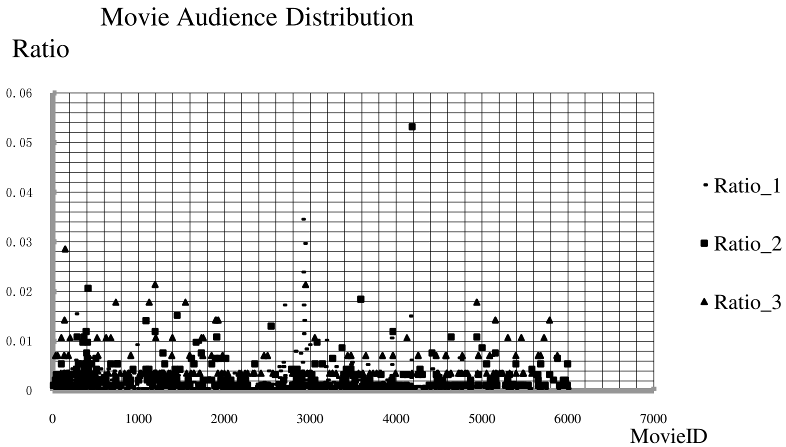

Fig. 3. Audience Ratio Distribution over all Movies

Fig. 2 and Fig. 3 show audience ratio distribution over movies. We set the length equals to 2, 3 and 4 respectively. It can be observed that the preference of users selected is most significant when the length of user behavior pattern is 2, which means the number of users that have similar preference in one user group set is the largest.

From the statistic chart, we can get this conclusion by noticing that the highest point belongs to the result of the user behavior pattern when the length is 2.

From the above observations we can get following result:

1. The preference of users in the same group does not become more obvious while the length of the behavior pattern increases. From the statistic chart above, the preference becomes most obvious when the length is 2. The reason for this result might due to the decrease of number in user group set while the length of the user behavior pattern increases.
2. The length of user behavior pattern also influences the dispersion of the result. While the length increases, the dispersion becomes small.

For different length user behavior patterns, the preference of every user group also different, which is reflected in the statistic chart by the different highest point in the normal distribution of the every result.

Based on above conclusions, we apply behavior patterns of length 2 to discover similar users and then combine BSSM with CF algorithm to support recommendation, which will be discussed in the next section.

4 Recommendation Based on BSSM

We tried several ways to combine BSSM with CF algorithm, and experimenting on the dataset to see whether the recommendation performance has been improved. Here we mainly introduce three methods, which are Linear combination of BSSM and CF algorithm (LCBC), multiplying the Results of BSSM with CF algorithm (MRBC), and Cascading Combination of BSSM with CF algorithm (CCBC).

4.1 LCBC

In traditional CF algorithm, we first calculate the similarity between target user and other users, and find out target user's nearest neighbor set by comparing the similarity, then make prediction on the target user's rating for a given item based on the user's nearest neighbors ratings on that item. We modify the algorithm by linearly combine the similarity of user behavior pattern between the target user and users in the nearest neighbors with the rating prediction calculating formula, which is

$$\text{ratingPredict}=rating_{NN} * ratio + rating_{NS} * (1 - ratio) \tag{4}$$

where $rating_{NN}$ indicates predicted rating for a given item that calculated from a CF algorithm, and $rating_{NS}$ indicates predicted rating for the given item that comes from BSSM algorithm, which is calculated by

$$rating_{NS} = \sum \frac{W_n * rating(n,j)}{\sum W_n} \tag{5}$$

where n is the number of nearest neighbor for the target user, and W_n is the similarity of behavior pattern between target user and user n in the nearest neighbor set.

By changing the value of ratio, we change the relative influences of these two algorithms on the final result, and get the predicted rating for each ratio. The experiment result is shown in Section 5.

4.2 MRBC

We consider another way to combine the two algorithms. After we get the nearest neighbor set for the target user by using CF algorithm, we calculate the BSSM between the target user and every user n in the nearest neighbor set, which is W_n .Then we calculate the predicting rating using the following formula

$$\text{ratingPredict} = \sum \frac{\frac{1}{distance[k]+1} * \omega_n * rating(k)}{\sum \frac{1}{distance[k]+1}} \tag{6}$$

Here, the first item in the multiplication indicates the distance between the target user and the neighbor using CF algorithm, and n is the number of users in the nearest neighbor set. After getting the predicting ratings for the target user in this way, we compare the result with traditional CF algorithm, which is also shown in Section 5.

4.3 CCBC

Since traditional CF algorithm might have low efficiency when the number of user group is large, we consider first using user behavior pattern to select users who have similar preference with the target user, then using CF algorithm to select the nearest neighbor set for the target user, and making prediction for the target user.

First we need to find out users who have similar preferences with the target user using BSSM algorithm. We use the last n items of the target user to form the target user behavior sequence ($M_1, M_2, M_3, \ldots, M_n$), and set the user behavior pattern length to be 2 for filtering. We consider setting a threshold value while selecting similar preference user group by comparing the similarity between target user and other users. When the similarity of the behavior pattern between the target user and a candidate user exceeds the threshold value, we add the candidate user into the similar preference user group for the target user. After comparing all users with the target user, we get the similar preference user group for the target user, and run CF algorithm on this user set to get the predicting ratings. We set several different threshold values and compare the recommendation performance for each of them to select the best threshold value. We choose 0, 0.05, 0.1, 0.15 as the tested threshold value, and compare the recommendation performance between each other. The result of the experiment is also shown in Section 5.

5 Experiments and Results

5.1 Experiment Datasets

We use two datasets for our experiments. The first one is MovieLens dataset. MovieLens is a movie recommendation website (http://www.grouplens.org/node/73). It uses user's ratings to generate personalized recommendation for other movies user would like or not. In the experiment we use MovieLens 1M data set, which consists of 1 million ratings from 6000 users on 4000 movies. MovieLens gets users' preference information by letting users rate for the movie. Before using their service, the user need to rate for at least 15 movies. MovieLens data set is a widely applied dataset among experiments on recommendation algorithm, which has already become the basic data set for evaluating recommendation algorithm. The second dataset is book reading behavior information of users from website www.qidian.com. www.qidian.com is one of the largest online reading and writing website in China. We select one-month reading behavior data of users and tested our improved algorithm on it. We first use MovieLens dataset to compare the three ways of integrating BSSM with CF method with the traditional CF method. Then we select the best one based on the recommendation performance and use dataset from qidian website to do the test experiment.

5.2 Experiment on LCBC

We measure the difference between predicted rating with the actual result by calculating the deviation between the two, which is calculated by formula

$$\text{MAE} = \sum |ratingPre(i) - rating(i)|/n \tag{7}$$

where *ratingPre(i)* is the predicted rating calculating from our algorithm, and *rating(i)* is the actual rating of the target user. n is the number of rating items. MAE reflects the recommendation precision of the algorithm.

The MAE of different ratio value is shown in Fig. 4.

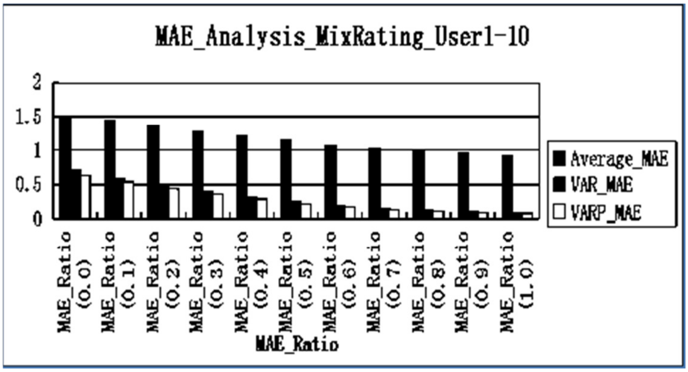

Fig. 4. MAE for LCBC

From the chart we can find that the recommendation precision is increased with the ratio increases, which means the modified algorithm performs worse than the collaborative filtering algorithm.

5.3 Experiment on MRBC

The MAE of different ratio value is shown in Fig. 5 and Fig. 6.

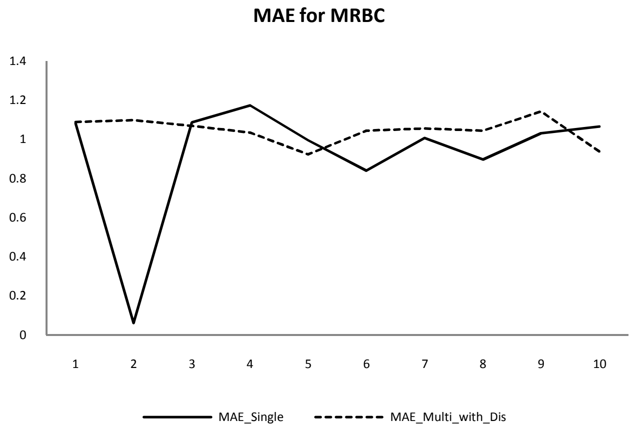

Fig. 5. MAE for MRBC

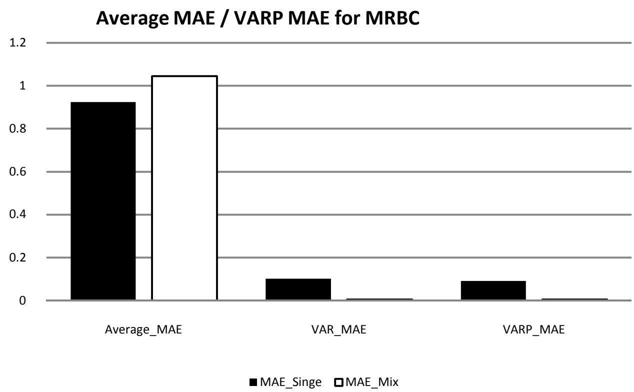

Fig. 6. Average MAE and varp MAE for MRBC

From the result, we can find the performance of modified algorithm has less recommendation precision, but is more stable than the traditional CF algorithm, which has more stable MAE value.

5.4 Experiment on CCBC

The MAE of different ratio value is shown in Fig. 7 and Fig.8.

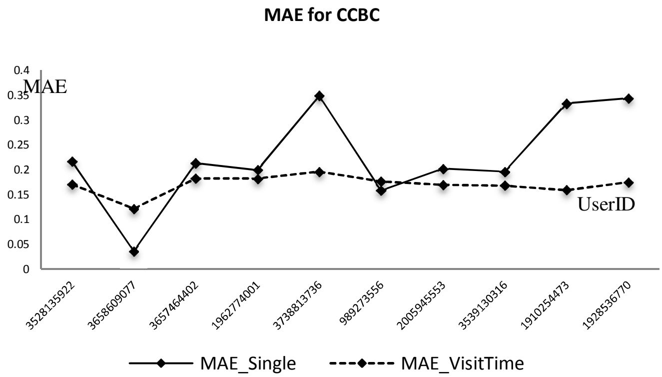

Fig. 7. MAE for CCBC

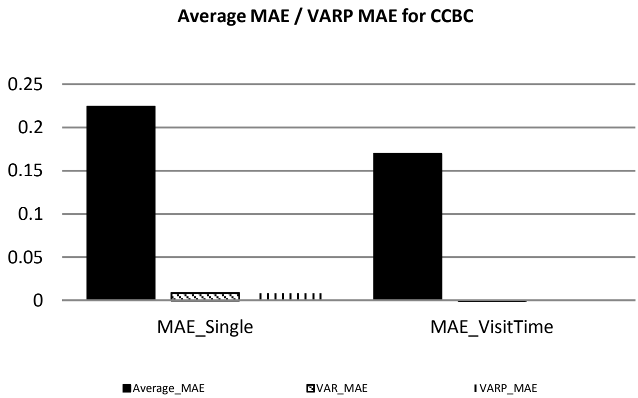

Fig. 8. Average MAE and varp MAE for CCBC

By comparing the result of each threshold, we can find that when the threshold value is 0.1, corresponding with the result of *MAE_Mix_simiRatio*>=0.1 in Fig. 7 and Fig. 8, the recommendation performance is the best, which has more stable MAE value.

5.5 Testing Experiment on CCBC

From the previous experiment, we find that the best way to integrate BSSM with the CF method is the third way, which uses user behavior pattern first before collaborative filtering algorithm. We use dataset from www.qidian.com as our test dataset to evaluate the performance of the algorithm. The experiment result is shown in Fig. 9 and Fig. 10.

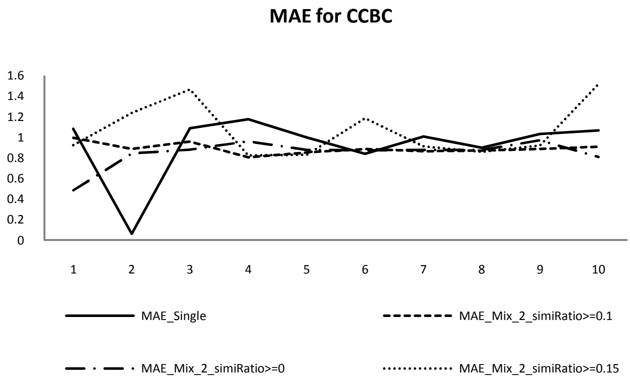

Fig. 9. MAE for CCBC

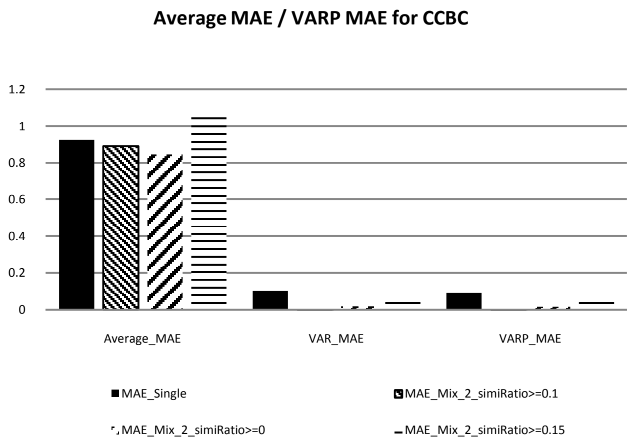

Fig. 10. Average MAE and varp MAE for CCBC

In Fig. 9, we used the CCBC method operating on the dataset from www.qidian.com and get the MAE value compared with single collaborative filtering method. From the result we can find that the user behavior pattern enhanced collaborative filtering method performs better than the traditional CF method, which have better precision and stability.

5.6 Experiment Results Analysis

We tried several ways to integrate the user behavior pattern algorithm into CF algorithm. We first add the two algorithm with different ratio assigned for each algorithm, and it turns out that the integration of the two algorithm does not improve the recommendation performance. This result might due to the fact that CF algorithm gets more precise result, while algorithm based on BSSM gets less precise result, and thus simply adding the results of two algorithms cannot improve the recommendation performance. Then we use BSSM algorithm before CF algorithm to select users who have similar preference with the target user, and then find nearest neighbors among these users using CF algorithm to get the predicting ratings. The result shows it produces less precise recommendation results, but it is more stable compared with traditional CF algorithm. The reason might be that target user and users in similar behavior user group are more similar compared with users in nearest neighbor set selected from the whole users. However, since the selection of the first step is relatively rough, so the whole performance is worse. Next we try to multiply the two algorithms, and set a threshold value for selecting users into similar behavior user group, and using CF algorithm running on this user set. The result turns out to be better than the traditional CF algorithm in both recommendation precision and stabilization. Thus, this is the best algorithm to combine BSSM and CF algorithm.

6 Conclusions

In this paper, we proposed a new model for personalized recommendation, which is based on user behavior sequence similarity measurement and integrated with CF algorithm. We defined a new approach to measure the user behavior sequence similarities, which can help to find users with similar preferences. Then we introduced three ways to combine BSSM with CF algorithm. We first combined the two methods parallel by assigning different ratio to them, and the result shows that it does not improve the recommendation performance. Then we multiplied the two methods to get the union recommendation, and the result shows that it improves the stability of recommendation performance, but does not improve the recommendation precision. After that we combined the two methods by first using BSSM to select the similar users for the target user, and then using CF method running on this group set, which turns out to have better recommendation performance in both precision and stability. Thus, this is the best way to integrate the two methods.

There are still some problems to be solved in our model, such as cool start problem and sparse data problem, which occurs in most CF methods. Also, we can improve our model by making combination between user behavior pattern methods with other recommendation algorithms. These issues will be the main focus in our future work.

Acknowledgments. This work is partially supported by China National Science Foundation (Granted Number 61073021, 61272438), Research Funds of Science and Technology Commission of Shanghai Municipality (Granted Number 11511500102, 12511502704), Cross Research Fund of Biomedical Engineering of Shanghai Jiaotong University (YG2011MS38).

References

1. [Alspector et al 1997] AL Spector, J., Koicz, A., Karunanithi, N.: Feature-based and Clique-based User Models for Movie Selection: A Comparative Study. User Modeling and User-Adapted Interaction 7(4), 279–304 (1997), doi:10.1023/A:1008286413827

2. [Badrul et al 2001] Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web (WWW 2001), pp. 285–295. ACM, New York (2001), doi:10.1145/371920.372071
3. [Cao 2010] Cao, L.: In-depth Behavior Understanding and Use: the Behavior Informatics Approach. Information Science 180(17), 3067–3085 (2010)
4. [George et al 2007] Lekakos, G., Giaglis, G.M.: A hybrid approach for improving predictive accuracy of collaborative filtering algorithms. User Modeling and User-Adapted Interaction 17(1-2), 5–40 (2007), doi:10.1007/s11257-006-9019-0
5. [Good et al 1999] Good, N., Ben Schafer, J., Konstan, J.A., Borchers, A., Sarwar, B., Herlocker, J., Riedl, J.: Combining collaborative filtering with personal agents for better recommendations. In: Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence (AAAI 1999/IAAI 1999), pp. 439–446. American Association for Artificial Intelligence, Menlo Park (1999)
6. [Guandong Xu et al 2008] Xu, G., Zhang, Y., Yi, X.: Modelling User Behaviour for Web Recommendation Using LDA Model. In: 2008 IEEE/WIC/ACM International Confernce on Web Intelligence and Intelligent Agent Technology (2008)
7. [Han et al 2011] Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques, 3rd edn. Morgan Kaufmann (2011) ISBN-13: 978-0123814791
8. [Helmult et al 1999] Hlavacs, H., Kotsis, G.: Modeling User Behavior: A Layered Approach. In: Proceedings of the 7th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 1999), p. 218. IEEE Computer Society, Washington, DC (1999)
9. [Resnick et al 1994] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work (CSCW 1994), pp. 175–186. ACM, New York (1994), doi:10.1145/192844.192905
10. [Resnick et al 1997] Resnick, P., Varian, H.R.: Recommender systems. Communications of the ACM 40(3) (1997)
11. [Joachims et al 1997] Joachims, T., Freitag, D., Mitchell, T.: Webwatcher: A Tour Guide for the World Wide Web. In: The 15th International Joint Conference on Artificial Intelligence (IJCAI 1997), Nagoya, Japan, pp. 770–777 (1997)
12. [Lieberman et al 1995] Lieberman, H.: Letizia: An Agent that Assists Web Browsing. In: Proc. of the 1995 International Joint Conference on Artificial Intelligence, pp. 924–929. Morgan Kaufmann, Montreal (1995)
13. [Myra et al 1998] Spiliopoulou, M., Faulstich, L.C.: WUM: A Web Utilization Miner. In: Proceedings of EDBT Workshop WebDB9 (1998)
14. [Schafer et al 2001] Schafer, J.B., Konstan, J.A., Riedl, J.: Electronic-commerce recommender systems. J. Data Mining Knowl. Discov. 5(1), 115–152 (2001)
15. [Su et al 2009] Su, X., Khoshgoftaar, T.M.: A Survey of Collaberative Filtering Techniques. Advances in Atificial Intelligence 2009, Article ID 421425, 19 Pages (2009), doi:10.1155/2009/421425