

UARR: A Novel Similarity Measure for Collaborative Filtering Recommendation

Yue Huang, Xuedong Gao, Shujuan Gu

*Dongling School of Economics and Management, University of Science and Technology Beijing, Beijing, 100083, P. R. China
Emails: mylinnia@126.com gaoxuedong@manage.ustb.edu.cn gushujuan@ustb.edu.cn*

Abstract: *User similarity measurement plays a key role in collaborative filtering recommendation which is the most widely applied technique in recommender systems. Traditional user-based collaborative filtering recommendation methods focus on absolute rating difference of common rated items while neglecting the relative rating level difference to the same items. In order to overcome this drawback, we propose a novel user similarity measure which takes into account the degree of rating the level gap that users could accept. The results of collaborative filtering recommendation based on User Acceptable Rating Radius (UARR) on a real movie rating data set, the MovieLens data set, prove to generate more accurate prediction results compared to the traditional similarity methods.*

Keywords: *Collaborative filtering, recommender system, user similarity measurement, User Acceptable Rating Radius (UARR).*

1. Introduction

The pervasiveness of Web 2.0 has brought the information overload problem, in which a person takes in the information faster than the brain can really process it. A recommender system [1], which can provide the users the information they need quickly according to their personal preferences, is an efficient solution to

information overload and has become an indispensable part in successful electronic commerce websites and social networks [2].

In general, there are three components in a recommender system [3]: recommended candidates, users, and a recommendation algorithm. With user preference information as input, the role of a recommender system is to assess the utility value of recommended candidates to users according to its recommendation algorithm, then recommend the users items with the maximum utility.

The recommendation algorithm is the core of a recommender system, and the existing methods for recommendation are usually divided into three categories [4]: content-based recommendation, collaborative filtering recommendation, and hybrid recommendation. Content-based recommendation methods [5] recommend users items that are similar to what they have selected before. The key of content-based methods lies in feature extraction through analysis of recommended candidates [6], thus this kind of methods is not suitable for multimedia objects with a complex structure, such as music, images and videos. While in the view of collaborative filtering (CF) recommendation approaches [7], user preferences can be obtained by analysis of user groups with similar behaviour or interests. Compared with content-based methods, CF-based methods do not need to obtain feature information, thus they are able to handle multimedia objects. The third category is a hybrid of the above two types [8], combining advantages of both kinds of methods to avoid their weaknesses. Apart from the above three types of approaches, there are also other recommendation methods, such as association-based recommendation [9], knowledge-based recommendation [10], and vague-set-based recommendation [11].

Collaborative filtering, as one of the most widely applied and successful recommendation techniques in personal recommendation [12], relies highly on similarity measurement. Calculating the user similarity based on k -nearest neighbours is a widely used model. However, this kind of method determines the user similarity according to the absolute value of common-rated items of two users, while neglecting the relative rating level difference to the same items. However, the service level of a recommendation system is influenced by its items recommended to users, which is similar to the situation in traditional services [13]. In addition, though different users may give the same rating for one specific item, their subjective acceptance of this item may be different, which is of significant importance in providing good services. To overcome this drawback of traditional methods, we propose a novel recommendation algorithm based on user acceptable rating radius, shown to be efficient in distinguishing users.

2. Collaborative filtering recommendation

CF recommendation methods predict user ratings for items based on preferences of user groups with similar interests, and then recommend the target users top- N items in the item list. There are three main steps of user-based CF recommendation algorithms [14]: building the user rating matrix according to user behaviour; finding the k -nearest neighbours based on user similarity measurement; and predicting user rating for a target item with respect to neighbours' ratings to the same items.

2.1. User rating matrix

In the first step of CF recommendation, most existing methods are based on the user rating matrix (shown in Table 1), derived from user historical behaviours, and then compute similarity between users according to this user rating matrix. With explicit feedbacks provided in the data set, the value of components in the matrix indicates how strong a particular user's interest to a particular item is. Whereas, when only implicit feedbacks are provided, this matrix becomes binary, in which one represents the item visited by the user and zero indicates the unvisited.

Table 1. User rating matrix for collaborative filtering recommendation

User \ Item	Item ₁	Item ₂	...	Item _n
User ₁	R_{11}	R_{12}	...	R_{1n}
User ₂	R_{21}	R_{22}	...	R_{2n}
...
User _m	R_{m1}	R_{m2}	...	R_{mn}

In order to measure the sparsity of a data set, the sparsity level [15], the proportion of zero entries to the entire data set, is introduced as

$$(1) \quad 1 - \frac{\text{nonzero_entries}}{\text{total_entries}}.$$

2.2. User similarity measurement

Computing similarity between users is the most important step of CF-based recommender systems, which then helps to find the most similar neighbours of each user. Considering a rating data set, given a user set $U=(U_1, U_2, \dots, U_m)$, where U_i , $i=1, 2, \dots, m$, denotes user i , and an item set $I=(I_1, I_2, \dots, I_n)$, where I_j , $j=1, 2, \dots, n$, denotes item j . Given two users a and b , firstly, let I_a denotes the item set a has rated, I_b denotes the item set b has rated, and $I_{a,b}$ denotes the common rated item set of a and b . Then, let $R_{a,j}$ be the rating of item j given by user a , $R_{b,j}$ be the rating of item j given by user b , R_{a_mean} be the average value of ratings of all items with user a rates, and R_{b_mean} be the average value of ratings of all items with user b rates. Lastly, let $\text{Sim}(a, b)$ be the similarity between user a and user b . Vector similarity measures, such as Cosine, modified Cosine, and Jaccard, are often used to compute the similarity between users.

- Cosine similarity

$$(2) \quad \text{Sim}_{\text{Cosine}}(a, b) = \cos(a, b) = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \times \|\vec{b}\|} = \frac{\sum_{j \in I_{a,b}} R_{a,j} R_{b,j}}{\sqrt{\sum_{j \in I_a} R_{a,j}^2} \sqrt{\sum_{j \in I_b} R_{b,j}^2}}.$$

- Modified Cosine similarity

$$(3) \quad \text{Sim}_{\text{ModifiedCosine}}(a, b) = \frac{\sum_{j \in I_{a,b}} (R_{a,j} - R_{a_mean})(R_{b,j} - R_{b_mean})}{\sqrt{\sum_{j \in I_a} (R_{a,j} - R_{a_mean})^2} \sqrt{\sum_{j \in I_b} (R_{b,j} - R_{b_mean})^2}}.$$

- Jaccard similarity

$$(4) \quad \text{Sim}_{\text{Jaccard}}(a, b) = \frac{\sum_{j \in I_{a,b}} R_{a,j} R_{b,j}}{\sum_{j \in I_a} R_{a,j}^2 + \sum_{j \in I_b} R_{b,j}^2 - \sum_{j \in I_{a,b}} R_{a,j} R_{b,j}}.$$

2.3. Rating prediction

Let NN be the k -nearest neighbors of a target user t , and $P_{t,j}$ denotes the rating prediction of t for an unrated item j . One of the most commonly used prediction methods is defined as in (5).

$$(5) \quad P_{t,j} = \frac{\sum_{u \in \text{NN}} \text{Sim}(t, u) R_{u,j}}{\sum_{u \in \text{NN}} \text{Sim}(t, u)}.$$

3. User similarity based on user acceptable rating radius

3.1. Basic concepts

Different users hold various preferences for items in a recommendation system. User preferences can be obtained by analyzing whether a user has rated an item and his evaluation. Most existing methods compute user similarity through comparison of absolute rating values given to the same items. However, in reality, although two different users have rated the same item, they may hold different opinions on it. The larger the evaluation gap given to the same item is, the larger dissimilarity of two users about this item is. For example, suppose two persons have seen a movie called “Avatar”. One rating is five, indicating it is strongly recommended, while the other rating is only two, indicating it is not worth seeing. Obviously, although these two have seen the same movie (meaning they are sharing something in common more or less), their preferences are rather different. In order to obtain more precise reflection of the users, we propose a novel user similarity measure which considers the user acceptable rating level difference, and give some definitions as follows.

Definition 1. Number of Rating Levels (NRL). Suppose there are several levels of ratings in a recommendation system. Define the integer number corresponding to the level starting from one as the number of rating levels, $\text{NRL} = (1, 2, \dots)$.

For example, the rating range in a recommendation system is $\{1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5\}$, thus according to Definition 1, its NRL is 9.

Definition 2. User Acceptable Rating Radius (UARR). Suppose a user could rate the items in a recommendation system. Define the user acceptable rating radius as the level gap which is acceptable to the user as in (6):

$$(6) \quad \text{UARR} = [\rho \bullet \text{NRL} + 0.5].$$

where $[]$ means getting the integer value of a variable, ρ denotes the degree of the rating level gap that users could accept. Usually, we take the value of ρ as $1/3$.

For example, given NRL is 9, according to Definition 2, its UARR is 3.

Definition 3. User Acceptable Rating Level Difference (UARLD). Given the UARR of a recommendation system, define the user acceptable rating level difference as the integer number starting from zero, that is $UARLD = [0, UARR]$ ($UARLD \in N$).

For example, the NRL and UARR of a recommendation is 7 and 2, respectively. Then, according to Definition 3, the range of its UARLD is $\{0, 1, 2\}$, which indicates a user gives a rating level 5 to an item, then who give a rating of 3, 4, or 5 are considered with similar preferences. However, the degree of user's interest in this item is not exactly the same, which is further described as follows.

Definition 4. Similarity Weight of User Acceptable Rating Level Difference (SW_UARLD). Given the UARR of a recommendation system, define the weight of different UARLD as follows:

$$(7) \quad SW_UARLD = 1 - \frac{UARLD}{UARLD + 1}.$$

For example, given $UARLD = \{0, 1, 2\}$, according to Definition 4, the SW_UARLD corresponding to 0, 1, 2 is 1, 2/3, 1/3, respectively.

Definition 5. User Similarity. Given two users U_i and U_j , let $U_i.I$ denote the items U_i rated, $U_j.I$ denote the items U_j rated. Define similarity of U_i and U_j as follows:

$$(8) \quad Sim_{UARLD}(U_i, U_j) = \frac{\sum_{k=0}^{UARLD} Count(k) \cdot SW(k)}{|U_i.I \cap U_j.I|},$$

where $Count(k)$ denotes the number of items that meets the condition of $UARLD = k$, $SW(k)$ denotes the similarity weight of $UARLD = k$, and $|U_i \cap U_j|$ denotes the size of intersection of $U_i.I$ and $U_j.I$.

3.2. Numerical example

In this section we will illustrate our algorithm by giving a running example of UARR on a toy problem. Suppose there are five items in a toy recommendation system, ratings are integers from one to five, and zero stands for unrated. Given the user rating matrix of two users shown in Table 2, user similarity calculation of those two is depicted as follows.

Table 2. Rating data of two users in a recommendation system

User \ Item	Item ₁	Item ₂	Item ₃	Item ₄	Item ₅
User ₁	4	5	4	5	4
User ₂	0	4	4	2	0

Step 1. According to Definition 1, $NRL=5$.

Step 2. According to Definition 2, $UARR=[(1/3) \times 5 + 0.5]=2$.

Step 3. According to Definition 3, UARLD is $\{0, 1, 2\}$.

Step 4. According to Definition 4, UARLD are $SW(0)=1-(0/(2+1))=1$, $SW(1)=1-(1/(2+1))=2/3$, and $SW(2)=1-(2/(2+1))=1/3$, respectively.

Step 5. Count the number of items within each UARLD that $Count(0)=1$, $Count(1)=1$, $Count(2)=0$. Get the rated items of U_1 as $U_1.I=\{1, 2, 3, 4, 5\}$, the rated items of U_2 as $U_2.I=\{2, 3, 4\}$, then $|U_i \cap U_j| = |\{2, 3, 4\}| = 3$. According to equation (8), user similarity of U_1 and U_2 is $Sim(U_1, U_2) = [(1 \times 1 + 1 \times (2/3) + 0 \times (1/3))] / 3 = 5/9 = 0.556$.

It can be seen that the similarity obtained by our method is consistent with the real situation.

4. Experimental evaluation

4.1. Data sets and evaluation method

To evaluate the proposed method UARR, we test CF methods based on user similarity using Cosine, modified Cosine, Jaccard, and UARR respectively on a benchmark data set – MovieLens-100K [16] which was collected by the GroupLens Research Project at the University of Minnesota. The data set consists of 100 000 ratings (1)-(5) from 943 users on 1682 movies. Each user has rated at least 20 movies. In order to compare the recommendation accuracy of four methods on data sets with different scales and sparsity levels, we extract four data sets with 250, 500, 750 and 943 users from MovieLens-100K (shown in Table 3).

Table 3. Four data sets extracted from MovieLens-100K

User \ Item	Users	Movies	Ratings	Sparsity
ML100K-U250	250	1467	24695	93.27
ML100K-U500	500	1616	56770	92.97
ML100K-U750	750	1660	80869	93.50
ML100K-U943	943	1682	100000	93.70

The experiments are conducted offline. Each of the four data sets are divided into two parts, of which 80 % is the training set, and the other 20 % as the test set. The k -nearest neighbors of users are computed using the training set, and then making predictions based on the test set with equation (5). There are many kinds of evaluation methods for recommendation algorithms, of which MAE (Mean Absolute Error) is one of the most widely applied metric to compare the recommendation accuracy [17]. In this paper we compare the MAE value of different methods on the above mentioned data sets. The smaller MAE value is, the better is the accuracy of the method.

4.2. Experimental results and analysis

First we test user distinguishing ability of four methods. Comparison of user similarity using Cosine, Modified Cosine (depicted as M-Cosine), Jaccard, and UARR on MovieLens-ML100K is shown in Table 4. It can be seen that 68.5 % of user similarities obtained by Cosine are within (0, 0.2], 59.4 % of user similarities obtained by M-Cosine are within (0, 0.2], 69.4 % of user similarities obtained by Jaccard are within (0, 0.1]. However, the user similarities obtained by UARR are

more evenly distributed. Thus, a conclusion can be drawn that UARR can distinguish users more efficiently than the other three.

Table 4. Distribution of similarity based on MovieLens-ML100K

User \ Item	Cosine	M-Cosine	Jaccard	UARR
sim<0	0	146237	0	0
sim=0	28657	31476	28657	46009
0<sim<=0.1	156916	233768	308256	252
0.1<sim<=0.2	147441	30181	91581	4174
0.2<sim<=0.3	77246	2190	14880	12367
0.3<sim<=0.4	28545	256	759	34330
0.4<sim<=0.5	5035	39	18	59116
0.5<sim<=0.6	303	6	2	77693
0.6<sim<=0.7	9	0	0	106347
0.7<sim<=0.8	1	0	0	55933
0.8<sim<=0.9	0	0	0	28743
0.9<sim<1.0	0	0	0	3291
sim=1.0	0	0	0	15898
Sum	444153	444153	444153	444153

Then we compare the recommendation accuracy of four methods on the above mentioned four data sets with different scales. The MAE comparisons of ML100K-U250, ML100K-U500, ML100K-U750, and ML100K-U943 based on Cosine, M-Cosine, Jaccard, and UARR under $k=10, 20, 30, 40, 50$, are depicted in Table 5. It is seen that the MAE value of UARR is smaller than the other three under all the four data sets, which shows that the prediction accuracy of CF based on UARR is better than based on Cosine, M-Cosine, and Jaccard in this case.

Table 5. Comparison of MAE for Cosine, M-Cosine, Jaccard, and UARR on four extracted MovieLens data sets

Data set \ Method	Cosine	M-Cosine	Jaccard	UARR	
ML100K-U250	10	0.9440	0.9418	0.9441	<i>0.9318</i>
	20	0.9420	0.9451	0.9417	<i>0.9304</i>
	30	0.9399	0.9478	0.9406	<i>0.9304</i>
	40	0.9390	0.9500	0.9394	<i>0.9304</i>
	50	0.9389	0.9548	0.9393	<i>0.9293</i>
ML100K-U500	10	0.9051	0.9052	0.9091	<i>0.8994</i>
	20	0.9082	0.9062	0.9118	<i>0.8969</i>
	30	0.9099	0.9086	0.9121	<i>0.8965</i>
	40	0.9106	0.9103	0.9125	<i>0.8968</i>
	50	0.9106	0.9116	0.9129	<i>0.8971</i>
ML100K-U750	10	0.9204	0.9163	0.9227	<i>0.9093</i>
	20	0.9212	0.9364	0.9229	<i>0.9085</i>
	30	0.9213	0.9754	0.9231	<i>0.9083</i>
	40	0.9219	1.0492	0.9236	<i>0.9086</i>
	50	0.9224	1.0381	0.9236	<i>0.9091</i>
ML100K-U943	10	0.9075	0.9098	0.9089	<i>0.8957</i>
	20	0.9065	0.9098	0.9079	<i>0.8964</i>
	30	0.9068	0.9097	0.9084	<i>0.8969</i>
	40	0.9072	0.9098	0.9085	<i>0.8973</i>
	50	0.9077	0.9104	0.9088	<i>0.8975</i>

5. Conclusion

Collaborative filtering recommendation is an efficient solution to the information overload problem, and its accuracy relies highly on similarity measurement. In this paper we propose a novel user similarity measure based on user acceptable rating radius (UARR) which considers not only absolute ratings for users' common rated items but also relative rating level difference to the same items. Firstly, we obtain the number of rating levels of a recommendation system and user acceptable rating radius. Then, we use similarity weight of user acceptable rating level difference to quantify the degree of user acceptance of differences in ratings on the same items. Lastly, we assess user similarity based on both the number of common rated items that are within the user acceptable rating radius and their relevant similarity weight. Experimental results on four extracted data sets with different scales and sparsity levels from MovieLens prove its competitiveness in prediction accuracy compared with Cosine, modified Cosine, and Jaccard. It is worthwhile that both effectiveness and efficiency of our method on larger data sets need to be further explored.

Acknowledgments: This work was supported by the National Natural Science Foundation of China under Grant No 71272161, and the Research Funds for University of Science and Technology Beijing Ph. D. Students under Grant No 06106097.

References

1. Resnick, P., H. R. Varian. Recommender Systems. – Communications of the ACM, Vol. **40**, 1997, No 3, 56-58.
2. Zhou, X., Y. Yu, Y. Li, A. Josang, C. Cox. The State-of-the-Art in Personalized Recommender Systems for Social Networking. – Artificial Intelligence Review, Vol. **37**, 2012, No 2, 119-132.
3. Xu, H., X. Wu, X. Li, B. Yan. Comparison Study of Internet Recommendation System. – Journal of Software, Vol. **20**, 2009, No 2, 350-362.
4. Adomavicius, G., A. Tuzhilin. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. – IEEE Transactions on Knowledge and Data Engineering, Vol. **17**, 2005, No 6, 734-749.
5. Bezerra, B. L. D., F. A. T. Carvalho. A Symbolic Approach for Content-Based Information Filtering. – Information Processing Letters, Vol. **92**, 2004, No 1, 45-52.
6. Balabanović, M., Y. Shoham. Fab: Content-Based, Collaborative Recommendation. – Communications of the ACM, Vol. **40**, 1999, No 3, 66-72.
7. Resnick, P., N. Iakovou, M. Suchak, P. Bergstrom, J. Riedl. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. – In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, ACM New York, NY, USA, 1994, 175-186.
8. Burke, R. Hybrid Recommender Systems: Survey and Experiments. – User Modeling and User-Adapted Interaction, Vol. **12**, 2002, No 4, 331-370.
9. Lin, W., S. A. Alvarez, C. Ruiz. Efficient Adaptive-Support Association Rule Mining for Recommender Systems. – Data Mining and Knowledge Discovery, Vol. **6**, 2002, No 1, 83-105.
10. Mandl, M., A. Felfernig, E. Teppan, M. Schubert. Consumer Decision Making in Knowledge-Based Recommendation. – Journal of Intelligent Information Systems, Vol. **37**, 2011, No 1, 1-22.

11. Cui, C., Z. Zang, F. Liu, Y. Qu. Application of Vague Set in Recommender Systems. – Journal of System and Management Sciences, Vol. **2**, 2012, No 4, 50-56.
12. Lee, S. K., Y. H. Cho, H. K. Kim. Collaborative Filtering with Ordinal Scale-Based Implicit Ratings for Mobile Music Recommendations. – Information Sciences, Vol. **180**, 2010, No 11, 2142-2155.
13. Phan, K., T. Daim. Exploring Technology Acceptance for Mobile Services. – Journal of Industrial Engineering and Management, Vol. **4**, 2011, No 2, 339-360.
14. Claypool, M., P. Le, M. Waseda, D. Brown. Implicit Interest Indicators. – In: Proceedings of the 6th ACM International Conference on Intelligent User Interfaces, ACM New York, NY, USA, 2001, 33-40.
15. Sarwar, B., G. Karypis, J. Konstan, J. Riedl. Analysis of Recommendation Algorithms for e-Commerce. – In: Proceedings of the 2nd ACM Conference on Electronic Commerce, ACM New York, NY, USA, 2000, 158-167.
16. Miller, B. N., I. Albert, S. K. Lam, J. A. Konstan, J. Riedl. MovieLens Unplugged: Experiences with an Occasionally Connected Recommender System. – In: Proceedings of the 8th ACM International Conference on Intelligent User Interfaces, ACM New York, NY, USA, 2003, 263-266.
17. Zhu, Y., L. Lv. Evaluation Metrics for Recommender Systems. – Journal of University of Electronic and Science and Technology of China, Vol. **41**, 2012, No 2, 163-175.