Use of Similarity Measure in Recommender System Based on Type of Item Preferences

Ashishkumar B. Patel[1,2(✉)] and Kiran Amin[3]

[1] LDRP Institute of Technology and Research, Gandhinagar, Gujarat, India
abp5@live.com
[2] C U Shah University, Wadhwan, Gujarat, India
[3] U V Patel College of Engineering, Kherva, Mehsana, Gujarat, India
kiran.amin@ganpatuniversity.ac.in

Abstract. During last twenty years recommender system have emerged as a research field. Recommender System is rooted in the field of Information Retrieval, Machine Learning and Decision Support System. Most of the users do not have enough knowledge to make automatic decisions. So they need recommendation of different items for better choice. Because of this many researchers tried to understand the algorithmic techniques for recommendation to the given user. It is very important factor to identify similar items related to the target user's test. To find similar items RS uses item preference of an item. In different RSs, the item preferences are available in different forms, i.e. preferences are either available, Boolean preference (yes/no) or not available. We test various User Similarity Measures for dataset with preferences, without preferences and Boolean preferences. We tested various similarity measures for User Based Collaborative Filtering techniques in Apache Mahout.

Keywords: Recommendation system · Item preference · User similarity
Collaborative filtering · Item similarity

1 Introduction

In our daily life we face different opinions and options i.e. thing we would like, don't like and even we don't care. We buy items online or from different stores. We watch movies online today. We listen songs on radio because it is of our choice or we don't notice it all. Same thing happens with hotel choice, tourism destinations, different websites, friend's updates, news etc. Although people's choice may vary, but there are some hidden patterns in their choices or in liking areas. People like the things which are liked by similar kind of other people or they like the things which they have knowledge or experienced in past. Recommender System (RS) is the area which predict about this hidden pattern, and by using these patterns to discover new things which user do not able to find it even if it is useful for them [4]. In real world the user has very large options available for choice, which are not even known to him/her. In such case recommender systems may help them to find relative items based on their choice. Such recommender systems uses preferences of items for recommendations.

A. K. Luhach et al. (Eds.): ICAICR 2018, CCIS 955, pp. 630–638, 2019.
https://doi.org/10.1007/978-981-13-3140-4_56

Figure 1 shows the concepts of recommendation [8]. A user who is seeking for recommendation may ask for a recommendation to the system, or a recommender system (recommender engine) may produce the list of recommended items to the user. After visiting the items recommended by recommender, the user rate the item based on his/her experience. Sometimes recommender system ask to provide the preference or rating of the item. The preferences provided by the user are stored in universe of alternatives or preference database, which will further help the recommender engine for accuracy of next recommendation in future.

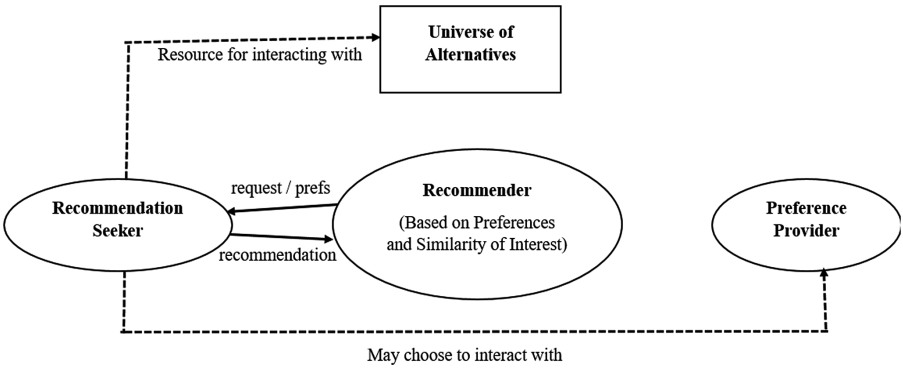


Fig. 1. Model of the recommendation process

2 Mahout – Collaborative Filtering Library

Apache Mahout [1] is a collaborative filtering library from Apache Software Foundation. It is also an open source library which includes machine learning and collaborative filtering algorithms in a single package. Mahout can be used for recommender engine with classification and clustering algorithms. Mahout is used to process scalable data. It can be used to process very large collection of data in a single machine. It has also support of Hadoop for distributed computing and Big Data. We implement and test various methods which uses inbuilt similarity algorithms in Mahout. Our methods takes preference of items from users and based on that it estimates preferences for other items.

Mahout generally uses collaborative filtering for recommendation. It takes users' preferences of rich set of items and find recommended items based on estimated preferences for target item. The Fig. 2 shows different components used for user based recommendation.

Top-level packages define the Mahout interfaces to these key abstractions:

- DataModel
- UserSimilarity
- ItemSimilarity
- UserNeighborhood
- Recommender

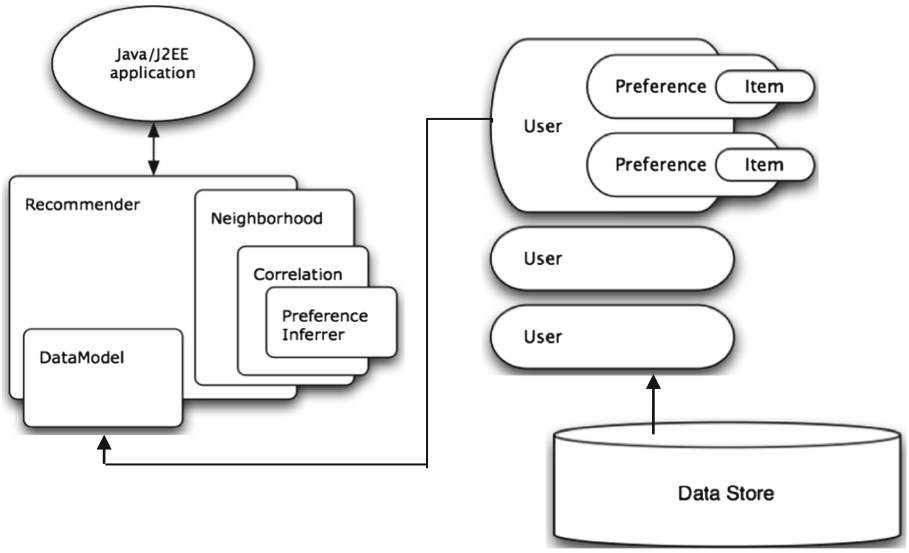


Fig. 2. Architecture of Mahout [4]

3 Dataset Used

MovieLense [10] data sets was by the GroupLens Research Project at the University of Minnesota. This dataset 943 users had rated 1682 movies. The rating of each movie in range 1 to 5. This dataset contains total 100000 ratings. In this dataset every user rated about 20 movies. This project was carried out in Computer Science Department of the University of Minnesota. In the dataset, the data is randomly ordered and tab separated.

user id | item id | rating | timestamp.

We used this data set for comparing similarity measures. By using this dataset several algorithms are tested. In this implementation the output will display the top-n items with their item id.

4 Evaluation Parameters for Recommender System

To understand or measure the accuracy of recommender systems several accuracy measures are used. The popular accuracy measures are Precision, Recall and F-Measures [2]. To derive the accuracy parameters the confusion matrix is often used [12]. The confusion matrix is as shown in Table 1.

Table 1. Confusion matrix

Actual/Predicted	Negative	Positive
Negative	A	C
Positive	C	D

Precision, Recall and F-Measure

Precision and Recall are the well-known metrics for measuring the accuracy of recommends in classical information retrieval (Table 2).

Table 2. Categorization of all possible recommendations

	Recommended	Not Recommended	Total
Used	True – Positive (TP)	False – Negative (FN)	Total Used
Not Used	False – Positive (FP)	True – Negative (TN)	Total not Used
Total	Total Recommended	Total Not Recommended	Total (T)

$$\text{Precision} = \text{count}\ (\text{N})\ /\ \text{N} \qquad \text{Precision} = \text{TP}\ /\ (\text{TP}\ +\ \text{FP}) \tag{1}$$

$$\text{Recall} = \text{count}\ (\text{N})\ /\ \text{R} \qquad \text{Recall} = \text{TP}\ /\ (\text{TP}\ +\ \text{FN}) \tag{2}$$

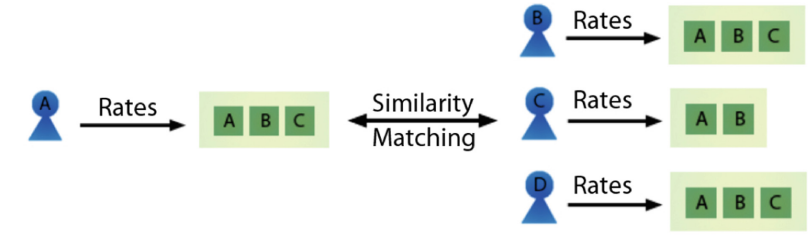
Recall is the ratio of number of items system correctly recall (TP) to the total number of all correct items (R). However precision is the ratio of no. of relevant records (TP) retrieved to the total no. of irrelevant (FP) and relevant (TP) records retired which is expressed in percentages [11]. It is the ration of number of items correctly recall to the number of all items called.

"Precision is defined as the proportion of relevant items in the predicted items and recall is defined as the proportion of predicted items in the relevant items" [11]. If R is the no. of relevant items in the list, then precision and recall are defined as in Eq. 1 and 2 where N is the total no. of items. In some RSs if trying to improve precision often worsen recall. F-measure is introduced as a measure of the harmonic mean of precision and recall.

$$\text{F-Measure}\ = 2 * (\text{Precision}\ *\ \text{Recall})\ /\ (\text{Precision}\ +\ \text{Recall}) \tag{3}$$

5 User Based Recommender System

Sometime people like the same items which are liked by similar kind of people [15]. The user based recommender finds the items which are preferred by similar kind of users based on different parameters like age, location, choice, qualification etc (Fig. 3).



	i1	*i2*	*i3*	*i4*	*i5*	*i6*	*i7*	*i8*
u1	0	1	1	1	1	0	0	0
u2	1	0	0	0	1	1	0	0
u3	0	0	0	1	0	1	0	1
u4	0	0	0	1	1	1	0	1
u5	1	1	0	0	0	0	0	1
u6	0	1	0	0	1	1	0	1
ua	0	0	1	1	0	1	0	1
	0	1	1	3	2	2	0	2

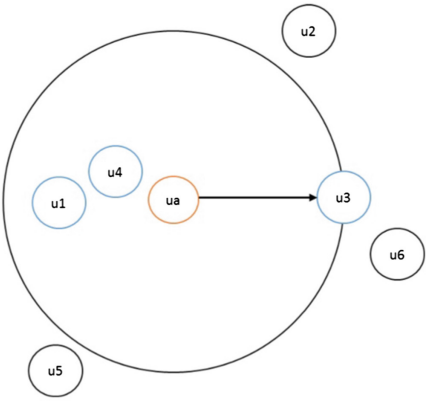


Fig. 3. User based collaborative filtering [3, 9].

As shown in figure the left hand side array is known as preference array. Each cell of preference array represents the preference of an item assigned by the user. If any cell has value 0 means he/she has not visited that item. Value 1 means user has visited and assigned the preference 1. Here we consider only Boolean preference of the items, means user has visited it or not. Sometimes the preference values may vary from 1 to 5 or 1 to 10. In user based Collaborative Filtering, system finds the neighbour users based on difference similarity modes as we experimented in this paper. In the figure the no. of neighbour n = 3 then the similar users related to ua are u1, u3 and u4. Based on the items preferred by similar users system finds items i5 and i2 are recommended items in descending order of some accuracy measures.

The UserSimilarity is one of the required components of the user based recommender method in Apache Mahout, which encapsulates some notion of similarity among users. The UserNeighborhood finds similar users.

6 Item Based Recommender System

In some cases user likes items which they know or for which they have knowledge or which they have purchased in past. The item based RS works on the information of users own preferences of items which he/she referred in past [4] (Fig 4).

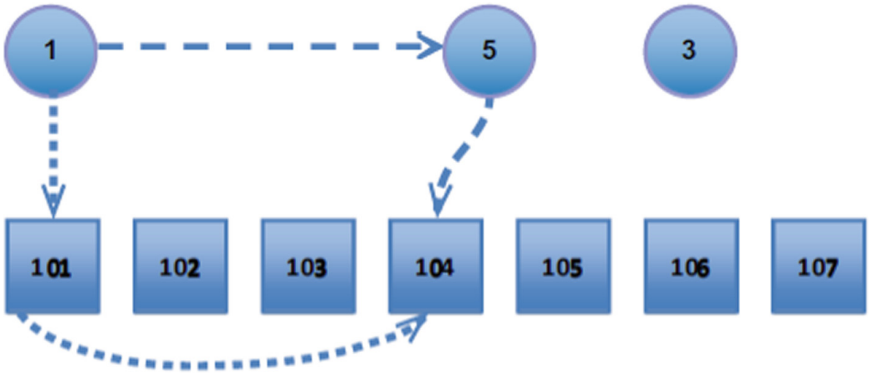


Fig. 4. Item based collaborative filtering [4].

As shown in figure if we want to find recommended items for user 1, then item based RS finds the items which user 1 have referenced in past (here item 101). Then it will find the items which are similar to item 101 (here item 104). Item based RS recommends item 104 to user 1.

7 Experiment on Dataset: Item Preferences Are Considered

For User Similarity, we have implemented and tested two similarity measures, Uncentered Cosine Similarity and Pearson Correlation Similarity.

A. **Cosine Similarity**

In this similarity [14] the result is the cosine of the angle formed between the two preference vectors. In this similarity metric, the item ratings or preferences are used as a vector to find the normalized dot product of the two users. Cosine similarity is the angular difference between two preference vectors. The expression of Cosine Similarity is:

$$Similarity = \cos(\theta) = \frac{\Sigma Ai * Bi}{\sqrt{\Sigma}(Ai)^2 * \sqrt{\Sigma}(Bi)^2} \tag{4}$$

This similarity will always between 0 and 1. Value 0 means no similarity between users or items and 1 means total similar. Based on this construct our experiment finds top-10 recommended items with Precision of 0.00522979397781299, Recall of 0.00584735743214348 and F1 of 0.010459588.

B. **Pearson Correlation Similarity**

The correlation is the association between two users or items. Correlation values are range from −1 to +1 [5]. Positive correlation states positive association and negative correlation states negative association. If we have two users X and Y, then Pearson correlation is term as:

$$Person(x,y) = \frac{\Sigma xy - \frac{\Sigma x \, \Sigma y}{N}}{\sqrt{(\Sigma x^2 - \frac{(\Sigma x)^2}{N})(\Sigma y^2 - \frac{(\Sigma y)^2}{N})}} \quad (5)$$

Our experiment generate the top-10 recommended items with Precision of 0.0164817749603803, Recall of 0.0185985963323522 and F1 of 0.03296355.

8 Experiment on Dataset: Item Preferences Are in Boolean Form or not Present

It not always possible for some recommender systems to have explicit items ratings are available [9]. As example, for news website which recommends news to user based on his/her previous news watched or read. In such recommender systems the mapping of user with news articles available but not with explicit rate or preference of the news. In such RSs it is not common to all users to rate the news.

A. Loglike Similarity

Sometimes there are such situation possible in which there are some common items preferences are possible between dissimilar users [7]. For example if you and I have rated 100 items each, and 50 overlap, we're probably similar. But if we've each rated 1000 and overlap in only 50, maybe we're not. This similarity measure is useful where two users are not similar but their overlap is due to chance (the numerator part of the expression). The denominator is the likelihood that it is not at all due to chance, i.e. that the overlap is because of our tastes are similar and the overlap is exactly what we would expect given that. When the numerator is relatively small, in such cases we are similar.

Based on this construct, our experiment gives the top-10 recommended items with Precision of 0.1256735340729, Recall 0.159425703720473 of and F1 of 0.251347068.

B. Tanimoto Coefficient Similarity

Tanimoto coefficient/ Jaccard Distance [13] is such a similarity which is used to measure surprise factor between two items [11]. This similarity focus on weather users have expressed items or not instead of the actual preference value. It is the total number of items expressed (intersection) by two users versus either user expressed (union). As illustrated in Fig. 5.

Based on this construct our experiment gives the top-10 recommended papers with Precision of 0.15949427480916, Recall of 0.156943501119412 and F1 of 0.31898855.

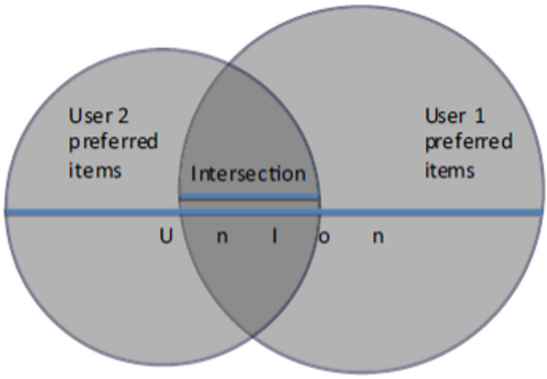


Fig. 5. The Tanimoto coefficient

9 Comparison Between Similarity Measures

According to above experiments, it is clear that the recommendations are more accurate when preferences are not considered. Both Loglike and Tanimoto Similarities are superior than similarities with item preferences are considered. Such algorithms may lead towards the serendipity of the recommender system [11] (Table 3 and Fig. 6).

Table 3. Comparison between all similarity measures on precision, recall and F1

Similarity	Precision	Recall	F1
Cosine	0.00523	0.005847	0.01046
Pearson	0.016482	0.018599	0.032964
Loglike	0.125674	0.159426	0.251347
Tanimoto	0.159494	0.156944	0.318989

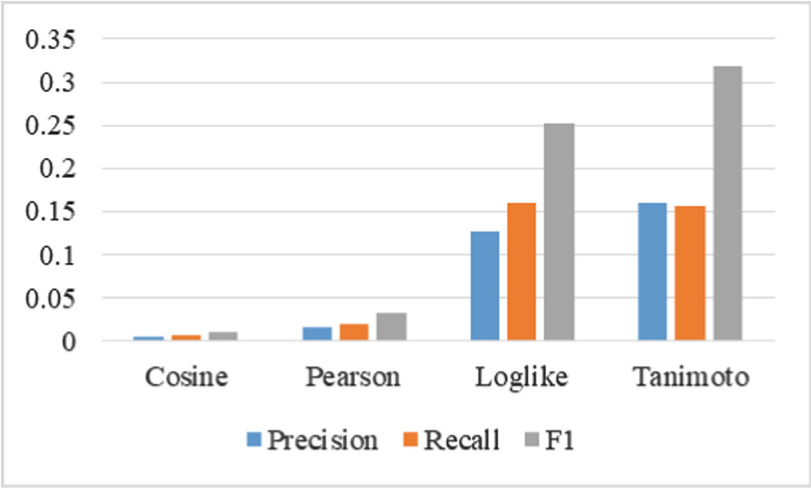


Fig. 6. Comparison between similarity measures for movielense dataset

10 Conclusion

We tested user based collaborative filtering approach with different similarity algorithms. Our methods tests the dataset with different cases of items preferences, i.e. items preferences explicitly available, Boolean (yes/no) preferences and in the case of preferences not available. Based on our experiments we conclude that when item preferences not considered then recommender system's accuracy in terms of precision and recall is higher. Tanimoto and Loglike similarity measures are promising algorithms which may be useful for recommender systems which focus on uncertainty or surprise which may trigger the Serendipity .

References

1. Apache Mahout, https://mahout.apache.org/
2. Ziegler, C.-N., McNee, S.M., Konstan, J.A., Lausen G.: Proceedings of the 14th International World Wide Web Conference (WWW '05), May 10–14, 2005, Chiba, Japan. To appear
3. Asanov, D.: Algorithms and methods in recommender systems. Technology (2011)
4. E. Friedman et.al.: Mahout in action. In: Manning Shelter Island (2012)
5. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B.: Recommender systems handbook. In: Media (2011)
6. Bobadilla, J., et al.: Knowledge based systems. Knowl. Based Syst. **46**, 109–132 (2013)
7. Keshav, R., et al.: Int. J. Comput. Sci. Inf. Technol. (IJCSIT) **5**(3), 4782–4787 (2014)
8. Terveen, L., Hill, W.: Beyond recommender systems: helping people help each other recommendation: examples and concepts (1), 1–21 (2001)
9. Hahsler, M.: Developing and testing top- N recommendation algorithms for 0–1 data using recommenderlab, 1–21 (2011)
10. MovieLense, http://www.movielens.org/
11. Patel, A., Amin, K.: Serendipity in recommender systems. Int. J. Eng. Technol. (IJET) **10**(1), pp (2018). https://doi.org/10.21817/ijet/2018/v10i1/181001067
12. Stehman, Stephen V.: Selecting and interpreting measures of thematic classification accuracy. Remote Sens. Environ. **62**(1), 77–89 (1997). https://doi.org/10.1016/S0034-4257(97)00083-7
13. Tanimoto, T.T.: IBM Internal Report 17th Nov 1957. http://en.wikipedia.org/wiki/Jaccard_index
14. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. Adv. Artif. Intell. **2009**(Section 3), 1–20 (2009)
15. Stephen, S.C., Xie, H., Rai, S.: Measures of similarity in memory-based collaborative filtering recommender system: a comparison. In: Proceedings of the 4th Multidisciplinary International Social Networks Conference (MISNC 2017). ACM, New York, Article 32, 8 pp (2017). https://doi.org/10.1145/3092090.3092105