A Jaccard base similarity measure to improve performance of CF based recommender systems

Conference Paper · January 2018

DOI: 10.1109/ICOIN.2018.8343073

4 authors, including:

Raja Mubbashir Ayub
University of Engineering and Technology, Taxila
7 PUBLICATIONS **74** CITATIONS

SEE PROFILE

Mustansar ali Ghazanfar
University of East London
57 PUBLICATIONS **1,142** CITATIONS

SEE PROFILE

Muazzam Maqsood
COMSATS University Islamabad, Attock Campus
72 PUBLICATIONS **1,030** CITATIONS

SEE PROFILE

A Jaccard base similarity measure to improve performance of CF based recommender systems

Mubbashir Ayub[1], Mustansar Ali Ghazanfar[2], Muazzam Maqsood[3], Asjad Saleem[4]

Email: {mubbashir.ayub[1], mustansir.ali[2], muazzam.maqsood[3], muhammad.asjad[4]}@uettaxila.edu.pk

University of Engineering and Technology Taxila, Pakistan

Abstract

Revolution in social computing has resulted in the wonderful evolution of recommender systems. Recommender systems maintain a repository of user profiles, created by a community of users, for generating personalized recommendations aimed at individual users. One of the approaches used in recommender systems is collaborative filtering (CF) which has become one of the most famous approaches for providing personalized recommendations to users. Nearest neighbors based methods used in CF are being widely used by many online stores to enhance users shopping experience. Nearest neighbors -based CF methods use some similarity measure techniques to find similar users/items for an active user/item. Almost all similarity measurement methods use ratings of commonly rated items while calculating similarity between a pair of users/items. Our approach works in the same manner as Jaccard similarity works. But Jaccard similarity does not consider the absolute value of rating and only considers the ratio of co-rated items. We take into account the ratio of absolute rating values which are equal in value, to the total no of co-rated items. An additional argument we take into account is the average rating value of users. We compared performance of our proposed method with many state-of-the-art similarity measures. Recommendation results from a set of real data sets show that our proposed measure has some performance improvement in terms of Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

1. Introduction

The advancements in technology have revolutionized the e-commerce in the last decade. Online shopping helps people to connect with each other, get reviews about products and buy products online. People are now more interested in buying things online rather than in traditional way. The users are now considering the online shopping as more of a social activity. People connect with their friends and colleagues through social networking sites and get reviews about different products. This helps people to make better decisions. As paradigm is shifting towards e-commerce, there is a plethora of information available on the internet. Recommender systems intend to recommend users products and information based on the needs and preferences by the community. Luckily, the activities of users can be traced and logged on the social networks and e-commerce sites. This helps to analyze the choice of users and then recommender systems are used to recommend information to users that best match with user expectations. Recommender systems provide personalized services through analyzing the user behaviors, such as the recommendation of photo groups in Flickr [1], the books in Amazon [2], videos in YouTube [3], and results in the Web search [4] .

Many algorithms of recommender systems have been developed and are being used in numerous online domains such as shopping websites, digital library, electronic media, and web based advertising. Algorithms of recommender systems can be classified into two main classes, content based filtering and collaborative filtering (CF), and these two can also be combined to overcome the short comings of each other.

CF algorithms use some similarity measure techniques such as Pearson Correlation Coefficient , Cosine, Jaccard Coefficient to find similar users/items for an active user/item [5]. Almost all similarity measurement methods use ratings of commonly rated tems while calculating similarity between a pair of users/ items. Our approach works in the same manner as Jaccard similarity works. But Jaccard similarity does not consider the absolute value of rating and only considers the ratio of co-rated items. We take into account the ratio of absolute rating values which are equal in value, to the total no of co-rated items. An additional argument we take into account is the average rating value of users. We compared performance of our proposed method with many state-of-the-art similarity measures. Recommendation results from three real datasets FilmTrust, CIAODVD and Epinions show that our proposed measure has some performance improvement in terms of Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

The remaining sections of our paper are organized as follows. In section 2 we discussed the many state-of-the-art similarity measures for automated product recommendation. In section 3, we discussed the shortcomings of existing methods and presented our proposed method. In section 4, we presented a brief overview of datasets that we used for comparison. Section 5 discussed the parameters that we used for comparison. In section 6 we presented the results in comparison with seven existing approaches. Section 7 concluded our work and discussed some future prospects of our work.

Table1. Description of notations

Notation	Values	Description
U	*{a, b, c,…p, z}*	Set of users
I	$\{i_1,i_2,\ldots.i_m\}$	Set of items
I_a	$\{I_{a1}, I_{a2}, \ldots I_{ak}\}$	Set of items rated by user *a*
$R_{a,j}$	*[1-5]*	Actual Rating of user *a* on item *j*.
$\hat{R}_{a,j}$	*[1-5]*	Predicted rating of user *a* on item *j*.

2. Related work

Due to easiness and effectiveness of Collaborative Filtering (CF) algorithms, they are generally used for recommendations by many websites like Netflix[6] and Amazon [9]. CF algorithms can be classified into two classes; model-based CF and memory-based CF [7]. Model-based approaches train a model from database of ratings and use this model for making prediction; example of which is Singular Value Decomposition (SVD) based models [8]. Memory based techniques construct a user-item rating matrix, such as the one shown in Table2. To generate recommendations for users, similarity functions are used on user and item ratings. The main part of recommender systems is to estimate similarity between users known as neighborhood identification. The similarity measure is used to find the k nearest neighbor according to their similarity. The value of k can range from 2 to 5 or to 20 or so on up to 100s. But using greater value of k does not guarantees that accuracy of the systems will be improved [5].

Table2. An example user-item rating matrix

	User 1	User 2	User 3	User 4	User 5
Item 1	2	-	2	2	2
Item 2	-	2	3	-	2
Item 3	3	2	3	5	2
Item 4	4	5	3	-	2

To find the similarity between users, a similarity function is required like Pearson Correlation Coefficient (PCC) [9]. In PCC, the similarity between two users/items *a* and *b* is calculated using Eq. (1). The similarity value lies in the range −1 to +1 and higher value makes a greater desire.

$$Sim\_PCC(a,b) = \frac{\Sigma_{j\in Ia\cap Ib}(Ra,j-\bar{R}a)(Rb,j-\bar{R}b)}{\sqrt{\Sigma_{j\in Ia}(Ra,j-\bar{R}a)^2}\sqrt{\Sigma_{j\in Ib}(Rb,j-\bar{R}b)^2}} \quad (1)$$

Where $R_{a,j}$ is the rating of user *a* for item *j*, $R_{b,j}$ is the rating of user *b* for item *j* and *j* is the set of common rated items between user *a* and *b*. $\bar{R}a$ is the average rating of user a and is calculated as:

$$\bar{R}a = \frac{1}{Ia}\sum_{j\in Ia}(Ra,j) \quad (2)$$

After computation of similarity according to the equation (1) the k-nearest neighbors' decision takes place, after which the prediction procedure is initiated. In prediction procedure ratings are being predicted for items/users. Those items/users whose predicted rating value is high are recommended to the target/active user. A target/active user is the one who made the request for recommendation.

Generally, the scale of ratings is finite in recommender systems and ratings can be categorized as positive or negative. A rating is considered to be positive if it is above the median value of rating scale and negative if it is below the median value. The constrained Pearson correlation coefficient (CPCC) [10] has been developed to measure the effect of these positive and negative ratings,. The CPCC is defined as follows:

$$Sim\_CPCC(a,b) = \frac{\Sigma_{j\in Ia\cap Ib}(Ra,j-R_{med})(Rb,j-R_{med})}{\sqrt{\Sigma_{j\in Ia}(Ra,j-R_{med})^2}\sqrt{\Sigma_{j\in Ib}(Rb,j-R_{med})^2}} \quad (3)$$

In Eq. (3) R_{med} is median value of rating in rating scale of dataset. Value of R_{med} is 3.0 in a scale of [1-5]. Drawback of CPCC is that it cannot be applied to datasets whose rating scale is even. For example rating scale of Filmtrust[11, 12] dataset is even.

There is another method known as Cosine Measure that considers angles between two vectors of ratings. The smaller the angle, more the similarity [13].

$$Sim\_COS(a,b) = \frac{\boldsymbol{\Sigma_{j\in Ia\cap Ib}(Ra,j)(Rb,j)}}{\sqrt{\boldsymbol{\Sigma_{j\in Ia}(Ra,j)^2}}\sqrt{\boldsymbol{\Sigma_{j\in Ib}(Rb,j)^2}}} \quad (4)$$

Rating items is a different experience for different people. Some people tends to rate high even for the items they don't like much and some tend to rate low even for those items which they like. Traditional cosine similarity does not take into account the preference of the user's ratings. For this, Adjusted Cosine Measure (ACOS) [14] has been introduced. It can be defined as:

$$Sim\_ACOS(a,b) = \frac{\boldsymbol{\Sigma_{p\in I}(Ra,p-\bar{R}a)(Rb,p-\bar{R}b)}}{\sqrt{\boldsymbol{\Sigma_{p\in I}(Ra,p-\bar{R}a)^2}}\sqrt{\boldsymbol{\Sigma_{p\in P}(Rb,p-\bar{R}b)^2}}} \quad (5)$$

Where $p \in I$ is the set of all items rated by user *a* or user *b*. If user *a* rate any item and user *b* did not rate that item, then value of rating for that item for user *b* will be zero.

Jaccard similarity considers the ratio of common rated items between two users, among total items they rated individually[14]. The idea behind this method is that two users are more similar if they have more common ratings.

$$Sim\_jaccard(a,b) = \frac{|\boldsymbol{I_a}\cap\boldsymbol{I_b}|}{|\boldsymbol{I_a}\cup\boldsymbol{I_b}|} \quad (6)$$

The extended Jaccard similarity guarantees that a user who purchases five instances of item1 and one instance of item2 will be different from a user who purchases one instance of item1 and five instances if item2.

$$Sim\_EJ(a,b) = \Sigma_{j\in Ia\cap Ib}\frac{(Ra,j).(Rb,j)}{(Ra,j)^2+(Rb,j)^2-(Ra,j).(Rb,j)} \quad (7)$$

PIP is another popular choice for similarity measure which computes three factors; proximity, impact and popularity between users for each item [15].

The PIP similarity between users *a* and *b* can be calculated as:

$$Sim\_PIP(a,b) = \sum_{j\in Ia\cap Ib} PIP\left(R_{a,j}, R_{b,j}\right) \quad (8)$$

Where *Sim_PIP(a, b)* is the PIP value for the two ratings $R_{a,j}$ and $R_{b,j}$ on items *j* belonging to common items of user *a* and user *b* respectively.

To overcome the limitations of PIP method, a new method has been proposed by *Haifeng Liu et al*, which covers the shortcomings of PIP based measure [14]. They pointed out that PIP method unnecessarily punishes the ratings more than once in situations where it is not required, while computing proximity and impact factors. They computed three factors, namely proximity, significance and singularity using a non-linear function, in the same way as of PIP method.

$$Sim\_PSS(a,b) = \sum_{j\in Ia\cap Ib} PSS\left(R_{a,j}, R_{b,j}\right) \quad (9)$$

These three factors are then combined with modified Jaccard measure. In order to estimate user rating preferences, they modeled an exponential function using user average values and user variances. According to them the NHSM similarity is in 0-1, because each part is from 0 to 1 [14].

Bobadilla et al. [12] proposed a new metric, which is called MJD (Mean–Jaccard–Difference),to solve the cold user problem. This metric includes three steps: first the selection of similarity measures, the new metric has six similarity measures after this step. Then, the weights of each similarity measure will be evaluated by neural network learning. A significance based similarity method was proposed by Bobadilla et al.[14] .This method first computes three types of significances, 1) the significance of an item, 2) the significance of a user, 3) the significance of an item for a user. After this traditional PCC or COS similarity will be used to compute the similarities among users according to the significance.

The prediction is accomplished among the pair of similar items/users and collective ratings. For item-based nearest neighbor method predicted rating is

$$\hat{R}_{a,j} = \bar{R}a + \frac{\Sigma_{p\in RN}\, Sim(a,b)(Ra,j-\bar{R}a)}{\Sigma_{p\in RN}\,|Sim(a,b)|} \qquad (10)$$

Where RN denotes the similar items in ratings dataset.

3. Proposed methodology

In this section, first we have analyzed drawbacks of all above similarity methods in detail followed by the discussion of proposed method.

Table 2 gives an example of a user-item rating matrix. In this table we have four items and five users. If an item is not rated by any user then that rating in the matrix is represented by the symbol -. We calculated the similarities of items in the table according to those similarity measures described the above. Table 4 shows the results of the similarities applied to Table 2. Since user similarity matrix is symmetric, we only show partial values.

(1) From Table 4 we can see that the Item1, Item2 and Item3 have very similar ratings. The rating vector is (2, -, 2, 2, 2) for Item1, (-, 2, 3,-, 2) for Item2 and (3, 2, 3, 5, 2) for Item3. However, we can see that the PCC similarity of (Item1, Item2) and (Item1, Item3) is zero. This drawback is still in the Constrained PCC which gives negative similarity for (Item1, Item3).

(2) PCC, Constrained PCC, Cosine, NHSM, Extended Jaccard and My Jaccard give maximum similarity for (Item2, Item3). Adjusted cosine and PIP gives maximum similarity for (Item1, Item3). Jaccard gives maximum similarity for (Item2, Item4).

(3) If we ignore absolute value of ratings then it will become hard to differentiate different items/users. On the other hand, the Jaccard similarity only considers the fraction of common rating, and ignores the absolute value of rating. This makes it difficult to differentiate between the items/users. From table 4 we can see that Jaccard similarity values are 0.6 for (Item1, Item3), (Item1, Item4) and (Item2, Item3) which makes it difficult to make a distinction among them.

(4) PCC results in zero rating if absolute value of a rating is equal to average rating of an item/user. Same drawback will be present in constrained PCC when an item rating value is equal to median value.

(5) It can also be observed from the similarity table that NHSM values are very small and PIP values are very large. This makes it difficult to make a comparison with other similarity measures and combine with other similarity measures. Moreover if Item1 rating vector is (5, 5, 5, 5, 5) and Item2 rating vector is also (5, 5, 5, 5, 5) then NHSM results in 0.12275. But it should be 1.0 according to [14].

(6) NHSM and PIP computations are lengthy and complex which makes it difficult to generate result in real time.

Based upon these drawbacks our proposed method is given below.

1. Our algorithm works in the fashion of Jaccard similarity.
2. Drawback of Jaccard similarity is that it considers the ratio of shared items but not the actual ratings.
3. We take into account the ratio of total common ratings to number of ratings which are equal in absolute value between two users/items or where average values of two users or items are equal.

$$N_T(a,b) = \begin{cases} N_T(a,b)+1;\ if\ \Sigma_{j\in Ia\cap Ib}\ Ra,j == Rb,j\ OR\ \bar{R}a == \bar{R}b \\ N_T(a,b); otherwise \end{cases} \qquad (11)$$

Where $N_T(a,b)$ is the number of ratings which are equal in absolute value or where average rating of two items/users is equal.

$$My\_jaccard(a,b) = \frac{|N_T(a,b)|}{|I_a \cap I_b|} \qquad (12)$$

Range of similarity value is from 0-1 and a higher value indicates more similarity.

4. Dataset Description

We tested our approach on two publicly available datasets CIAODVD and Epinions. CIAODVD [11, 16] ratings dataset contains, 72,665 ratings in the scale of [1-5]. The rating values range from 1.0 to 5.0 with step of 1.0. The dataset contains 16,121 items rated by 17,615 users.

Epinions.com [11, 17] is an online community website that allow users to review different products and services. Ratings dataset contains 6, 64,823 ratings in the scale of [1-5]. The ratings take values form 1.0 to 5.0 with step of 1.0. The dataset contains 1, 39,738 items rated by 40,163 users.

5. Evaluation Methodology We performed five-fold cross validation by dividing randomly the dataset into training and test set by using 80-20 rule and presented the average results.

Many evaluation metrics are used to compare performance of recommender systems [43]. In our work we

Table4. Results of applying various similarity measures on user-item rating matrix of Table2

PCC	Item2	Item3	Item4	Constrained PCC	Item2	Item3	Item4
Item1	0.0	0.0	0.0	Item1	0.707	-0.223	0.0
Item2		1.0	-0.188	Item2		1.0	-0.316
Item3			0.0	Item3			-0.288
Cosine	Item2	Item3	Item4	**Adjusted Cosine**	Item2	Item3	Item4
Item1	0.980	0.948	0.964	Item1	0.097	0.910	-0.307
Item2		0.982	0.904	Item2		-0.363	0.392
Item3			0.934	Item3			-0.688
PIP	Item2	Item3	Item4	**NHSM**	Item2	Item3	Item4
Item1	487	1957	924	Item1	0.0179	0.0234	0.00718
Item2		1089	909	Item2		0.0310	0.0190
Item3			1551	Item3			0.0274
Jaccard	Item2	Item3	Item4	**Extended Jaccard**	Item2	Item3	Item4
Item1	0.4	0.6	0.6	Item1	0.909	0.702	0.782
Item2		0.6	0.75	Item2		0.933	0.718
Item3			0.2	Item3			0.77
My_Jaccard	Item2	Item3	Item4				
Item1	0.5	0.25	0.34				
Item2		1.0	0.67				
Item3			0.5				

evaluated the performance of our proposed method in terms of both Mean absolute error (MAE) and Root mean square error (RMSE). MAE, measures the degree to which a prediction is close to the original value and a smaller value indicates a better accuracy. Hence, the smaller the MAE value is, the closer a prediction is to the original value. MAE is defined using Eq. (13).

$$MAE = \frac{\Sigma\, a,j_{\in N}|R_{a,j} - \hat{R}_{a,j}|}{TR} \qquad (13)$$

Where *TR* is the total number of test ratings. We also used RMSE to measure the accuracy of recommender systems. Formally RMSE is measured as follows in Eq. (14)

$$RMSE = \sqrt{\frac{\Sigma a,j_{\in N}|R_{a,j} - \hat{R}_{a,j}|^2}{TR}} \qquad (14)$$

6. Results and Comparison with existing approaches

We compared the performance of our method with PCC, Constrained PCC, Cosine, Adjusted Cosine, Jaccard, Extended Jaccard and PIP similarity measure. A threshold *hs* is defined for all similarity measurements whose value is set to *0.01*.Item prediction is then generated by using Eq. (10). All above methods are K-nearest neighbor based methods and value of K is 25 for all methods.

Pearson (PCC): Computes item similarity using Eq. (1). Its value range is from -1 to +1. Only those users/items are selected whose similarity is above the predefined similarity threshold *hs* for Eq (1).

Constrained PCC: This method is based on PCC. Difference is that it considers the median value in rating scale instead of items/ users average rating values. Item similarity is computed using Eq. (3). Value of R_{med} is *3.0* for all datasets. Its value range is also from -1 to +1.

Cosine: Computes item similarity using Eq. (5). Its range is also from 0 to 1.Drawback of Cosine is that it did not consider the user/item average rating value. This drawback is solved in Adjusted Cosine.

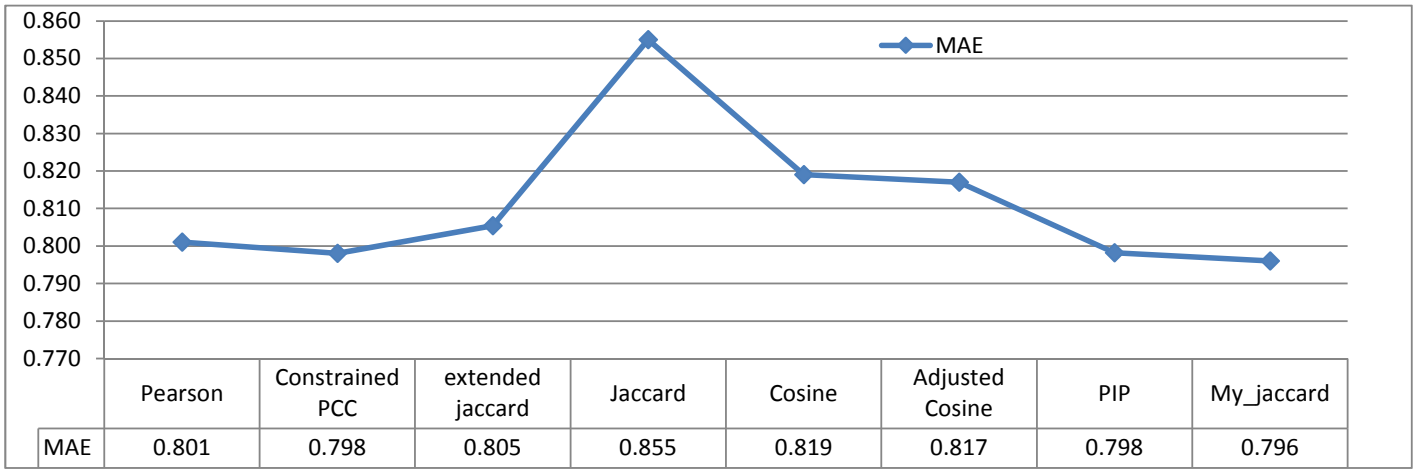
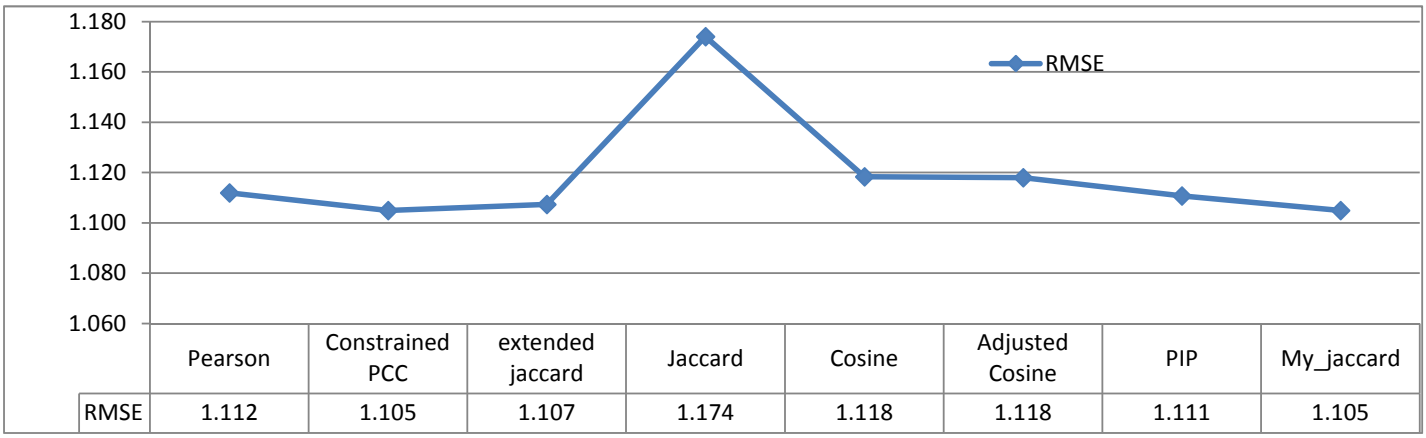
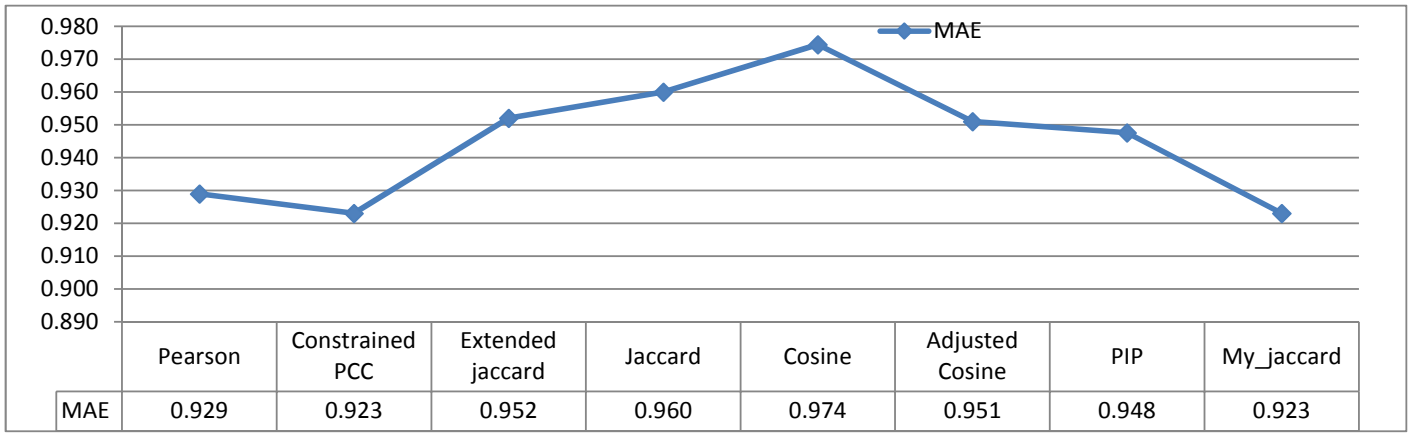


Figure1. MAE Comparison on CiaoDVD dataset



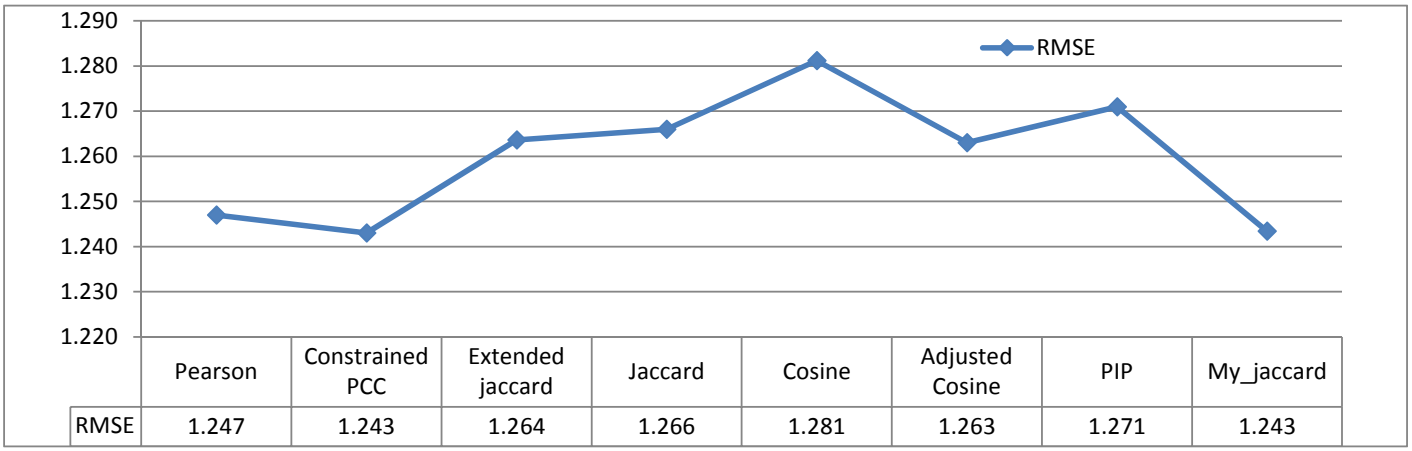
	Pearson	Constrained PCC	extended jaccard	Jaccard	Cosine	Adjusted Cosine	PIP	My_jaccard
RMSE	1.112	1.105	1.107	1.174	1.118	1.118	1.111	1.105

Figure2. RMSE Comparison on CiaoDVD dataset



	Pearson	Constrained PCC	Extended jaccard	Jaccard	Cosine	Adjusted Cosine	PIP	My_jaccard
MAE	0.929	0.923	0.952	0.960	0.974	0.951	0.948	0.923

Figure3. MAE Comparison on Epinions Dataset



	Pearson	Constrained PCC	Extended jaccard	Jaccard	Cosine	Adjusted Cosine	PIP	My_jaccard
RMSE	1.247	1.243	1.264	1.266	1.281	1.263	1.271	1.243

Figure4. RMSE Comparison on Epinions Dataset

Adjusted Cosine: Computes item similarity using Eq. (4). Its range is also from -1 to +1. It's calculations are similar to PCC but difference is that it is applied to all items instead of co-rated items as in PCC

Jaccard: Computes item similarity using Eq. (6). Its value range is also from 0 to 1.

Extended Jaccard: Computes item similarity using Eq. (7). Its value range is also from 0 to 1.

PIP: Computes item similarity using Eq. (8). According to [14] PIP can only be applied to only those datasets whose rating scale is odd, such as CiaoDVD and Epinions datasets. On the other hand rating scale of filmtrust dataset is even. But for all datasets we used the value of median rating as 3.0, while computing PIP factors.

It can be seen that results of MAE for CiaoDVD dataset are best for our proposed method. Results of PIP and Constrained PCC are 0.798, which are very close to our proposed method 0.796. In terms of RMSE on CiaoDVD our proposed method also has a very improved value 1.105, and the only match is with Constrained PCC. For Epinions dataset MAE value of My_Jaccard is 0.923 and only competing method is Constrained PCC whose MAE value is also 0.923.

In terms of RMSE value is 1.243 for My_Jaccard method and again its only match is with Constrained PCC.

For Filmtrust dataset value of MAE for My_Jaccard is 0.625.This value is 0.624 for Pearson and Adjusted Cosine. This value is 0.623 for constrained PCC. But the difference is very close. One important point to be noted here is that Constrained PCC is dependent upon median value of rating scale which is not defined for even rating scale datasets and we used *3.0* in our evaluation. In connection with RMSE values, value of RMSE for My_Jaccard method is 0.823 and this value is 0.818 for PIP, Constrained PCC and Extended Jaccard.

7. Conclusion and future work

Although many algorithms for collaborative recommender systems have been developed but still there is a desire for more accurate system. People need an intelligent system which can help them in the decision making process in various online environments such as online shopping and online communities. Moreover, being a part of expert systems recommender systems are supposed to provide as precise recommendations as possible, so as to minimize a human involvement in a specific area. Therefore, we proposed a similarity method that improves the accuracy of CF recommender systems. There will be the situations when the similarity between two users/items is measured as zero and prediction cannot be performed. This is due to the fact that the similarity of users is engraved only if they share a certain amount of equal value ratings. In such situations, a different similarity measure needs to be used, until sufficient ratings are available in the dataset to make the proposed Jaccard base approach usable.

Our future work will concentrate on finding some thresholds on user average values, while calculating N_T using Eq. (8), using some machine learning techniques such as SVM for different datasets, and then investigate the effect of these thresholds on recommendation accuracy. Another possible improvement is including domain information, such as tags/folksonomy to precisely enhance the model. Another possible future direction could be the prediction using My_jaccad on some cold start users/items using friendship network of the users. In this work we evaluated the performance of our method using predictive accuracy measures. Our future study will also include the extension of our approach to situations when sparsity is very high and finding similar users or items is very difficult.Another future work could be the evaluation of our method in terms of classification accuracy measures such as precision and recall.

8. References

1. N. Zheng, L.Q., L. Shengcai, Z. Leiming,, *Which photo groups should I choose? A comparative study of recommendation algorithms in Flickr,* . Inform. Science, 2010. **36**(6): p. 732-750.
2. E. Brynjolfsson, Y.J.H., M.D. Smith,, *Consumer surplus in the digital economy: estimating he value of increased product variety at online booksellers.* Manage. Sci, 2003. **49**(11): p. 1580-1596.
3. B. Shumeet, R.S., D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, M. Aly, , *Video suggestion and discovery for YouTube: taking random walks through the view graph* International Conference on World Wide Web, 2008: p. 895-904.
4. X. Zhang, Y.L., *Use of collaborative recommendations for web search: an exploratory user study.* J. Inform. Sci., 2008. **34**(2): p. 145-161.
5. Nikolaos Polatidis, C.K.G., *A multi-level collaborative filtering method that improves recommendations.* Expert Systems With Applications, 2016. **48**: p. 100-110.
6. www.netflixprize.com, h., *http:// www.netflixprize.com.* Accessed on 29th August 2017.
7. A. T. Gediminas Adomavicius, *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, 2005. **17**: p. 734-749.
8. Prugel-Bennett, M.A.G.a.A., *A Scalable, Accurate Hybrid Recommender System.* Third International Conference on Knowledge Discovery and Data Mining, 2010.
9. J.L. Herlocker, J.A.K., L.G. Terveen, J.T. Riedl, *Evaluating collaborative filtering recommender systems.* ACM Transactions on Information Systems, 2004: p. 5-53.
10. U. Shardanand, P.M., *Social information filtering: algorithms for automating word of mouth* Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1994: p. 210-217.
11. http://www.librec.net/datasets.html.
12. Guo, G.a.Z., J. and Yorke-Smith, N., *A Novel Bayesian Similarity Measure for Recommender Systems.* Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI), 2013: p. 2619-2625.
13. Greg Linden, B.S., and Jeremy York. 2003, *Amazon.com recommendations: Item-to-item collaborative filtering.* IEEE Internet Computing, 2003. **7**: p. 76-80.
14. Haifeng Liu , Z.H., Ahmad Mian, Hui Tian, Xuzhen Zhu, *A new user similarity model to improve the accuracy of collaborative filtering.* Knowledge-Based Systems, 2014. **56**: p. 156-166.
15. Ahn, H.J., *A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem.* Information Sciences, 2008. **178**(1): p. 37-51.
16. Guo, G.a.Z., J. and Thalmann, D. and Yorke-Smith, N., *ETAF: An Extended Trust Antecedents Framework for Trust Prediction.* Proceedings of the 2014 International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2014: p. 540-547.
17. http://www.epinions.com/.