Comparative Study of Different Similarity Functions in Collaborative Filtering

[1]Bharti Sharma, [2]Gunjan
[1]Assistant Professor, [2]Assistant Professor
MSIT

Abstract **- Recommendation System is an information filtering technique, which provides users with information, which they may be interested in. It serves as a guide for a user to navigate through a large dataset helping them in discovering products of their interest. In the last two decades, more than 250 research articles were published about research-paper recommendation systems. On one hand, it is interesting to have many recommendation techniques to produce an output as each has its own advantages but it also makes it strenuous to choose the right fit to address each scenario. Also it is challenging to turn these recommendation techniques to be used in real world situations. So, it is important to help the designing team choose the best technique to get optimal solutions in order to improve serviceability and make it cost efficient. In this paper, we here list out all the techniques i.e. the similarity functions used in collaborative filtering for ready reference and propose genetic function for genetic recommend generating method. On a dataset from MovieLens and different collaborative filtering approaches, experiments were performed, in order to evaluate proposal. Top three recommendations are shown using different similarity functions in collaborative filtering approach.**

keywords **- Recommendation system, collaborating system techniques, similarity functions, genetic algorithm**

I. INTRODUCTION

Recommender system is a subclass of information filtering system that is widely used to address the challenge of overwhelming information. When we want to choose from large number of products, we usually rely on someone else recommendations to clear all doubts. We can get these recommendations either directly [1] through texts or videos. Examples of influencer include book reviewers, film critics, newspapers and online social networks. The popular domains of application where the recommender systems are applied includes E-commerce [2] E-government [3][4] Social Network [5] Academia [7] [8] Entertainment [6] Telecom [9] , and so on. Progress in this topic can be comprehensively seen from the following paper[10][11][12].

The recommender system is categorized, based on the approach used to generate recommendations,[13] into three major categories: (i) Content-based approach, wherein the recommendations are based on user's history and past choices; (ii) Collaborative filtering, wherein the recommendations are based on the items considered by other users with similar interests and; (iii) hybrid approaches that utilizes and combines the techniques of the above said approaches based out of recommendations.

Collaborative filtering (CF) is the most used recommendation techniques. Based on the preferences of their like-minded neighborhood, CF recommendation focuses on providing the active users a list of interesting items. In this, the similarity between

It calculates the resemblance between the users and then suggests the aimed user items that are not yet used based on the similarities [14]. The likeness is evaluated on the basis of previous record of the shared item. When two users appear to be similar, then target user is recommended the item that he has not yet evaluated but his similar user had already evaluated it in the past. The CF approach follows the four basic steps [15]

1. Firstly, a similarity metrics is created by searching for users with the same rating patterns as that of the active user.
2. The metrics obtained from step 1 is used to select the subset of h neighbors that have the maximum correspondence with the active user
3. The ratings are normalized and forecasts the selected items based on the evaluations of neighbors with their weights. Here, weight refers to the value of likeness between the neighbor and the aimed user.
4. The best n items are presented in decreasing order of the forecasted scores to the target user.

The emphasis should lay on the importance of evaluation as an unalienable component for designing a good CF algorithm. There are various metrics through which the performance of the recommender systems can be evaluated. These are - Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).Given the N actual/predicted rating pairs (ru, i, pu, i), where u refers to a user and i to an item, the MAE of the N pairs is evaluated as:

$$\text{MAE} = \frac{|\sum_{i=1}^{N}(p_{u,i}-r_{u,i})|}{N} \quad (1)$$

and the RMSE of the N pairs is evaluated as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(p_{u,i}-r_{u,i})^2}{N}} \quad (2)$$

The lower MAE and RMSE , the more accurate the predictions which is a symbol of better performance of a recommender system. The following four steps should be performed to calculate MAE/RMSE:
(1) The dataset is divided into a test set (acting as active users) and a training set.
(2) Based on similarity metrics for a rated item of the present test user in the test set, the top-k neighbors are computed and thereafter selected from the training set.
(3) The ratings of the top k nearest neighbors are combined to calculate a prediction for the item.
(4) Step 2 and 3 are repeated until every rated item of every user in the test set obtains its prediction and then MAE and/ or RMSE are computed for the test users.
The CF based algorithms can be categorized into: memory-based algorithms, model-based algorithms and hybrid of memory-based algorithms and model-based algorithms. The difference lies in the processing of matrix of ratings (User χ Item). The model-based algorithms involve building a model based on the dataset of rating. In the first step, the algorithm extracts the necessary information from the original rating matrix and uses the same as a standard to make suggestions without needing to refer to the complete matrix every time. In next step, the model generated above is used as input matrix to calculate prediction rating for target user.[16]. On the other hand, the whole matrix is used to estimate the predictions by the memory-oriented collaborative filtering. Firstly, it uses data (votes, likes, reviews, etc) to establish correlation between the target-users and the users that are similar to the target-user. Then, based on similarity measures item i is recommended to target-user who's never seen it before but its similar neighbor had seen it. The memory based method further divided into two parts: (i) user-based algorithm, which recommends items by finding similar users [17]. (ii) item-based algorithm, which calculate similarity between items and make recommendations. In some cases both the algorithm become inefficient, so there is a need for the development of new collaborative filtering algorithm that may provide more accurate recommendations in all scenarios [18]. For that it is important to determine the likeness of the users in an accurate manner. Various methods like Pearson, Euclidean and Tanimoto correlation are elucidated in this literature to accomplish this. It is also important to have a large amount of options to choose from so that there may exist good solutions for all situations. However, sometimes it becomes difficult to choose which recommendation processes is a perfect fit. Each approach has its own advantages and performance parameters that are based out of the context of application. Therefore, it become necessary to carefully analyze which approach is suitable in the present scenario. Based on the study, various combined approaches were used [19], the use of search algorithm for newer combine techniques were not explored. Therefore, the list created by GA is proportional to the performance of each technique. Through this, the RMSE is lowered, hence the error is reduced.

II. Related Work

The primitive system which uses CF approach was the Tapestry [15][16][17], which was used to filter E-documents. For instance, a user can frame filtering guidelines for e-mail like "Show me all documents answered by other members of my research group". The above application can be used only in small communities where each person knows the interest and duties of others, so that each user can determine the relevant predictive relationships with different users. After that, there has been no looking back in the development of CF systems. Various studies have improved the performance of CF systems. For instance, in order to obtain personalized outcomes, [20] combined the basic and advance mechanisms of the CF techniques. The adjustments in CF technique that are proposed by [16] are based on applying weights on the similarity coefficient belonging to each rating and the number of items that have been calculated with the aimed user. Content-based approach is used in proposed algorithm in order to attract users towards the set of multidimensional vector modal. After that, the CF is implemented to search for the target customers that are synonymous to that of the most similar neighbors. Adding to this, the authors [21][16] proposed hybridization of single-class classification and collaborative filtering to produce a cascade hybrid recommendation. It has two levels. The first level uses the content-oriented method by using the model of One-Class classification to include the individual user choices. The purpose of the next level is to assign specific scores to the items so as to classify them. Therefore, it is a hurdle for the recommender systems to become effective as well as handy in real-world scenarios. It's become a hot topic now-a-days. The examples of such hurdles include retrieval of information, as can be inferred from the study of [22] that handles the usual problem faced during various documents online. As per the authors' reports, people while reading face difficulty in understanding a passage. Also they wish to learn more about topics covered in paragraph, In that case they look up for more resources about the concerned topic. These resources should be similar to the concerned topic.

III. Different Similarity Functions

In Recommendation System Similarity measurement model plays a vital role. In memory based Collaborative Filtering similarity measure: Traditional measures such as cosine similarity, Euclidean distance, Pearson correlation coefficient are frequently used similarity measures in personalized recommendation systems. A list of existing similarity measures in user based CF algorithm is given in table 1

Table 1. Advantages and disadvantages of existing Similarity Measures.

Sr. No.	Similarity Measure	Formula	Major Drawbacks
1.	Cosine Similarity (COS)	$sim(u,v) = \frac{\sum_{i \in I}(r_{u,i})(r_{v,i})}{\sqrt{\left(\sum_{i \in I'} r_{u,i}^2 \sum_{i \in I'} r_{v,i}^2\right)}}$	1) It provides high similarity Regardless of the significant difference in rating made by two users. 2) This is the method suffers from the problem of few co-rated items

			by both users.
2.	Adjusted Cosine Similarity (ACOS)	$sim(u,v)=\frac{\sum_{i\grave{o}I}\left(r_{u,i}-\overline{r_u}\right)(r_{v,i}-\overline{r_v})}{\sqrt{\left(\left(r_{u,i}-\overline{r_u}\right)^2\right)}\sqrt{\left(\left(r_{u,i}-\overline{r_v}\right)^2\right)}}$	1) It provide slow similarity regardless of the similar rating made by two users. 2) If set of co rated items by both users u and v is very then the similarly value produced by this model will not be reliable.
3.	Pearson Correlation Coefficient (PCC)	$sim(u,v)=\frac{\sum_{i\grave{o}I}\left(r_{u,i}-\overline{r_u}\right)(r_{v,i}-\overline{r_v})}{\sqrt{\sum_{i\ I}\left(r_{u,i}-\overline{r_u}\right)^2}\sqrt{\sum_{i\ I}\left(r_{u,i}-\overline{r_v}\right)^2}}$	1) It provides low similarity regardless of the similar rating made by two users. 2) If set of co rated items by both users u and v is very then the similarly value produced by this model will not be reliable.
4.	Constrained Pearson Correlation (CPCC)	$sim(u,v)=\frac{\sum_{i\epsilon I}(r_{u,i}-r_{med})(r_{vri}-r_{med})}{\sqrt{\left(\sum_{i\epsilon I}(r_{u,i}-r_{med})^2\right)}}$ $I^{'}=set\ of\ corated\ items\ by\ both\ users$ $r_{med}=medianalue\ in\ rating\ scale$	1) It suffers from few co-rated problems.
5.	Sigmoid Pearson	$sim(u,v)=sim(u,v)^{pcc}\frac{1}{1+\exp^{(-\frac{\bar{I}}{2})}}$	1) It provides high similarity value regardless of the difference between the two user's ratings.
6.	Mean Square Difference (MSD)	$sim(u,v)=1-\frac{\sum_{i\grave{o}I'}\left(r_{u,i}-\overline{r_{v,i}}\right)^2}{\lvert I'\rvert}$	1) It ignores the proportion of common rating. This may lead to low accuracy.
7.	Jaccard Coefficient	$sim(u,v)=\frac{\lvert I_u\rvert\cap\lvert I_v\rvert}{\lvert I_u\rvert\cup\lvert I_v\rvert}$	1) This approach does not consider absolute rating value of two user while calculating a similarity.
8.	Jaccard mean Square Difference	$sim(u,v)^{jmsd}=sim(u,v)^{Jaccard}.sim(u,v)^{msd}$	1) It particularly addresses the drawback of jaccard and MSD. This measure utilizes all rating provided by two user u and v.
9.	PIP (proximity Impact popularity)	$sim(u,v)^{PIP}=\sum_{i\in I'}PIP(r_{u,i},r_{v,i})$	1) It is not normalised. 2) Global preference of the user behaviour is not addressed.
10.	NHSM (New Heuristic Similarity Measure)	$sim(u,v)^{NHSM}=sim(u,v)^{JPSS}.sim(u,v)^{URP}$ $sim(u,v)^{PSS}=\sum_{i\in I'}PSS(r_{u,i},r_{v,i})$ $sim(u,v)^{JPSS}=sim(u,v)^{PSS}.sim(u,v)^{JACCARD}$ $sim(u,v)^{URP}=1-\frac{1}{1+\exp(-\lvert\mu_u-\mu_v\rvert.\lvert\sigma_u-\sigma_v\rvert)}$	1) Ratings on non co-rated items are neglected in this approach. 2) Similarity computation is very complex.
11.	Bhattacharya coefficient	$BC(i,j)=BC(\overline{P_i},\overline{P_j})=\sum_{n=1}^{m}\sqrt{(\overline{P_{in}},\overline{P_{jn}})}$	1) This approach can't be used to find a similarity between the pair of users if they rate on few or no similar items.
12.	Jaccard Uniform Operator Distance (JacUOD)	$sim(u,v)=\begin{cases}\frac{\lvert s_{u,v}\rvert}{\lvert s_u\cup s_v\rvert}*\frac{\sqrt{m(v_{\max}-v_{\min})^2}}{\sqrt{\sum s\in s_{u,v}(r_{u,s}-r_{v,s})^2}} \\ if\ \exists\ s\in s, r\neq r_{v,s}\end{cases}$	1) This approach suffers from few or no co-rated items.

13.	A New Similarity Measure Using Bhattacharya coefficient for CF.	$sim(u,v) = Jacc(u,v) + \sum_{i \in I_u} \sum_{i \in I_v} BC(i,j) loc(r_{u,i}, r_{v,j})$ $loc(r_{u,i}, r_{v,j}) = \frac{(r_{u,i} - \overline{r_u})(r_{u,i} - \overline{r_v})}{\sigma_u \sigma_y}$ $loc(r_{u,i}, r_{v,j}) = \frac{(r_{u,i} - r_{med})(r_{u,i} - r_{med})}{\sqrt{\sum_{k \in I_u} (r_{u,k} - r_{med})^2} \sqrt{\sum_{k \in I_v} (r_{u,k} - r_{med})^2}}$	1) This approach is not scalable and similarity computation is very complex.

On implementing these similarity functions and evaluating their performance using MAE and RMSE values, it was found that they gave the following mean values:

TABLE 2. PERFORMANCE COMPARISON

S. No.	SIMILARITY FUNCTION	MEAN RMSE	MEAN MAE
1	Pearson	1.0486	0.8393
2	MSD	0.9858	0.7807
3	Cosine	1.0359	0.8218
4	Pearson Baseline	1.0069	0.7919
5	Adjusted Cosine	1.0456	0.8193
6	Constrained Pearson	0.6863	0.8810
7	Sigmoid Pearson	1.1356	0.9418
8	Jaccard Coefficient	1.0949	0.9819
9	Jaccard Mean Square Difference	1.0256	0.9893
10	PIP(proximity impact popularity)	1.0318	0.8912
11	NHSM	1.0678	0.9822
12	Bhattacharya Coefficient	1.0256	0.9785

Therefore, on comparing these similarity functions, since the mean RMSE value of MSD is the lowest therefore out of these similarity functions, MSD gives the best similarity on using MovieLens dataset. However, on comparing the mean MAE values, Cosine Similarity gives the best results.

However, these values are dependent on the type and the size of the data set used. On using a different dataset with a different size, the performance of the similarity function changes. Therefore, implementation of these functions on multiple data sets with different sizes is required which we plan to work on next.

IV. CONCLUSION AND FUTURE WORK

This paper discusses about the existing similarity functions under collaborative filtering in recommendation systems. It also discusses on the drawbacks of these functions. We implemented these functions using MovieLens (ml-100k) data set with the help python. On evaluating the performance of recommender system using MAE and RMSE values, the performance of different similarity functions were compared. For future work, to get a better knowledge of the performance of these similarity functions we intend on implementing them using other datasets and evaluating their performance.

V. REFERENCES

[1] Shardanand, Upendra, and Pattie Maes. "Social information filtering: algorithms for automating “word of mouth”." *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1995.

[2] Zhang, Zui, et al. "A hybrid fuzzy-based personalized recommender system for telecom products/services." *Information Sciences* 235 (2013): 117-129.

[3] Lu, Jie, et al. "BizSeeker: a hybrid semantic recommendation system for personalized government-to-business e-services." *Internet Research: Electronic Networking Applications and Policy* 20.3 (2010): 342-365.

[4] Al-Hassan, Malak, Haiyan Lu, and Jie Lu. "A semantic enhanced hybrid recommendation approach: A case study of e-Government tourism service recommendation system." *Decision Support Systems* 72 (2015): 97-109.

[5] Liu, Xin, and Karl Aberer. "SoCo: a social network aided context-aware recommender system." *Proceedings of the 22nd international conference on World Wide Web*. 2013.

[6] He, Qi, et al. "Context-aware citation recommendation." *Proceedings of the 19th international conference on World wide web*. 2010.

[7] Sugiyama, Kazunari, and Min-Yen Kan. "Scholarly paper recommendation via user's recent research interests." *Proceedings of the 10th annual joint conference on Digital libraries*. 2010.

[8] McDonald, David W. "Evaluating expertise recommendations." *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*. 2001.

[9] Barragáns-Martínez, Ana Belén, et al. "A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition." *Information Sciences* 180.22 (2010): 4290-4311.

[10] Lee, Seok Kee, Yoon Ho Cho, and Soung Hie Kim. "Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations." *Information Sciences* 180.11 (2010): 2142-2155.

[11] Liu, Xin, and Karl Aberer. "SoCo: a social network aided context-aware recommender system." *Proceedings of the 22nd international conference on World Wide Web*. 2013.

[12] Lu, Jie, et al. "Recommender system application developments: a survey." *Decision Support Systems* 74 (2015): 12-32.

[13] Adomavicius, Gediminas, and Alexander 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* 17.6 (2005): 734-749.

[14] Herlocker, Jonathan L., et al. "An algorithmic framework for performing collaborative filtering." *ACM SIGIR Forum*. Vol. 51. No. 2. New York, NY, USA: ACM, 2017.

[15] Cacheda, Fidel, et al. "Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems." *ACM Transactions on the Web (TWEB)* 5.1 (2011): 1-33.

[16] Resnick, Paul, et al. "GroupLens: an open architecture for collaborative filtering of netnews." *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. 1994.

[17] Shardanand, Upendra, and Pattie Maes. "Social information filtering: algorithms for automating "word of mouth"." *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1995.

[18] Goldberg, David, et al. "Using collaborative filtering to weave an information tapestry." *Communications of the ACM* 35.12 (1992): 61-70.

[19] Adomavicius, Gediminas, and Alexander 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* 17.6 (2005): 734-749.

[20] Herlocker, Jonathan Lee. *Understanding and improving automated collaborative filtering systems*. PhD thesis: University of Minnesota, 2000.

[21] Lampropoulos, Aristomenis S., Dionysios N. Sotiropoulos, and George A. Tsihrintzis. "Evaluation of a cascade hybrid recommendation as a combination of one-class classification and collaborative filtering." *2012 IEEE 24th International Conference on Tools with Artificial Intelligence*. Vol. 1. IEEE, 2012.

[22] Liu, Lei, Georgia Koutrika, and Shanchan Wu. "Learningassistant: A novel learning resource recommendation system." *2015 IEEE 31st International Conference on Data Engineering*. IEEE, 2015.