International Journal of Advanced Computer Science and Information Technology (IJACSIT)
Vol. 7, No. 4, 2018, Page: 60-70, ISSN: 2296-1739

www.elvedit.com

Improving Performance of User-Based Collaborative Filtering Recommender Systems Using the Weighted Similarity Method

Authors

Younes Narouiee
Department of Computer, Zahedan Branch, Islamic Azad University, Zahedan, Iran
ynarouiee@yahoo.com
Zahedan, Iran

Hassan Asadollahi
Department of Computer, Damavand Branch, Islamic Azad University, Damavand, Iran
hasan_asadolahi@yahoo.com
Damavand, Iran

Abstract

Recommender systems are an example of the most successful web personalization tools. The most important duty of a recommender system, is finding the user's favorite items in a very large space of selectable items. Similarity-based algorithms, often referred to as memory-based collaborative filtering techniques, are one of the most successful methods in recommendation systems. When explicit ratings are available, similarity is usually defined using similarity functions, such as the Pearson correlation coefficient, cosine similarity or mean square difference. These metrics assume similarity is a symmetric criterion. Therefore, two users have equal impact on each other in recommending new items. In this paper, we introduce new weighting factors that allow us to consider new features in finding similarities between users. These weighting factors, first, transform symmetric similarity to asymmetric similarity by considering the number of ratings given by users on non-common items. Second, they take into account the habit effects on users which are regarded on rating items by measuring the proximity of the number of repetitions for each rate on common rated items. Experiments on dataset were implemented and compared to other similarity measures. The results show that adding weighted factors to traditional similarity measures significantly decreases Error resulting from them.

Key Words

Recommender systems, Collaborative filtering, traditional similarity measures, Weighted Similarity.

I. INTRODUCTION

Today in which data is generated too much, the massive amount of information allows users to spend more time and energy choosing an item. This item may be a book for an Amazon customer, a point of interest for a tourist, or a course for a student. To overcome the overwhelming problem of information, advocate systems help users to find the content they want in a reasonable time by analyzing behavioral data that is relevant to the user's activities [1]. Recommendation systems to provide the most relevant content helps users, depending on their taste, help their relationships and profiles. Various approaches are proposed as the basis for the advisory system [2]. Today, the use of advisory systems has become a necessity and many Internet sites use the proposed system to serve their customers well and sell more of their products [6].

A business website or social network must be maximized in order to be successful in achieving its goal, and it must be excellent in identifying its users' interests in order to provide its users with appropriate services, hence the use of advisory systems he does. A time-consuming advocate system can increase the users and customers of the website, which produces recommendations and suggestions tailored to the tastes and interests of each user, among a wealth of information and thousands of selectable items. To do this, the collaborative refinement system recommends a similarity between users, in this way, to suggest items to which the user is similar to the active user (the user whose system is developing the recommendation for him) to use in the system. The traditional similarity criteria, such as the Pearson correlation coefficient [4-7], cosine similarity [5, 8, 9], and the mean square difference [5, 10, 11], assume that the similarity of a concept is symmetric, which means the two users have the same effect on one another.

Given the ever-increasing amount of information, we can say that we are in the midst of a huge amount of data and information that, without proper guidance and navigation, we may have wrong or non-optimal choices among them [16]. Recommender systems, which are a kind of systems that influence the guidance of the user, are among a huge amount of possible choices to reach their favorite and preferred option, as this process is for the same user personalization. One of the most important challenges that the advisory systems face are the choice of the optimal similarity criterion [14,15], the likelihood of the score matrix and the cold start problem [17], which led to some problems in the proposed process and it reduces the accuracy of the suggestions. In this study, the error of the proposed systems has been dramatically reduced by the proposed method. The introduction of a weighted similarity function, which is a more effective way to calculate the similarity between users, which dramatically reduces the number of shared items and the margin of error rates by taking into account the value of the points.

II. RELATED WORKS

A lot of research has been done implicitly in various areas such as user modeling and web mining to use Web user behavior to create a model of his interests.

A memory-based approach first measures similarity between users using similarity criteria, and then the user's target score points to the desired item in terms of weight or weight of the concessions given by neighbors [9]. Model-based algorithms include methods based on scoring matrix or probabilistic method. For example, we can use the single value divide (SVD), Bayesian network [8], Probabilistic Matrix Factorization (PMF) and cluster models [16]. Model-based methods try to discover the underlying features in the data, and use these latent features to predict unremarkable ratings. The advantage of the methods based on the method is less on the time of computation than on memory-based methods, which becomes clearer when the recommended systems consist of a large number of item operations. On the other hand, the advantage of using a memory-based method is that less parameter should be adjusted [17].

Traditional similarity criteria, such as PCC, COS and MSD, are not reliable under certain conditions. For example, when there are not enough ratings to calculate the similarity (cold-start conditions) among users, the results from these criteria are contradictory. Different approaches to this weakness have been proposed. Liu and his colleagues suggested that the similarity between users is measured based on their average and standard variance of their previous rankings [9]. They considered three aspects for user rankings: proximity, importance, and singularity. Bobadilla and his colleagues presented a metric based on neural learning (a model based CF) and adapted it to the newly-started cool conditions of the new user [2].

Random walk-like similarities are other ways to deal with the start-cold problem [6]. Jamali and colleagues used the trust of users who were trusted from a trust network to help solve the problem of data privacy and start-ups. Their frameworks combine trust-based and item-based CF-based recommendations to benefit from more revenue without interfering with noise data [5]. One of the outstanding studies on shared memory-based advices and increased trust for cold-start users was done by Massa et al. [11]. They used explicit trust with a user-item rating matrix as inputs to predict rank. Release of trust can help them as long as new users provide at least one trusted friend.

There is another strategy in trust based systems where trust values are computed automatically, for example, based on past users' behavior in providing reliable advice, or based on the rules of similarity the user is introduced by Papagelis and his colleagues [14]. In a trust based CF, the computational model of trust works according to the similarity of items that are ranked by users [13]. The Papagelis model is based on trust inference, which is a transitional relationship between users who participate in underlying social networks. [15] Valuable resources, such as additional information, help to cope with issues of dullness (coldness) and cold start.

In recent years, some experts have tried to solve the problem of data embedding by combining common filters and material-based recommendations. Kampus et al. [10] proposed a hybrid method to take advantage of both methods. By using Bayesian networks, the relationship between users and items, as well as their strengths, can be demonstrated. Combined recommendations based on clustering models divide the <user, item> score matrix into smaller clusters, and use a neighboring cluster to predict unknown rankings [19]. Saranya and colleagues proposed a new hybrid method that uses the latent features extracted from the items. The

extracted properties are represented by a multi-attribute record using a probabilistic model [12]. In some studies, to integrate the benefits of complementary methods, integration of different methods has been proposed. For example, the JCard's average square difference (JMSD) of MSD is used to calculate the rating grabbing and uses the Jaccard index to consider the ratio of shared privileges among users [7]. Candillier and colleagues advised to use the Jaccard index as a weighting method and to combine it with other similarity criteria. The product of the Jaccard index with cosine similarity, the Pearson correlation coefficient, and the Manhattan distance confirm that Pearson has the best-performing weight among them [3].

In addition, a graph-based approach has been used to exploit its benefits in representing the relationship between users and items in a simple and flexible manner. In graph-based methods, data is represented as a graph whose nodes are either users or items (or both), and promotes interactions or similarities between users and the items. The correlation relations obtained with graph-based methods can be used to recommend the items [4]. Fouss and his colleagues suggested that Euclidean travel distance be used, which is a random walk method for calculating the similarities between nodes. The SimRank algorithm is another graph based model used to calculate similarity [16]. For example, Shine et al. [18] used the SimRank-based algorithm to build a public recommendation system, but Shardanan et al. [6] suggested that SimRank be combined with clustering to accommodate users in online dating networks.

Additionally, a dimming dimension, such as SVD, is usually used to reduce the dimension of the database of the recommendation system. Barragns-Martnez et al. Used SVD to diminish the adjacent dimensions of active items, and then implemented an object-based filtration with this low-level representation to produce predictions. [1] Kannan et al. developed a matrix factor-enhancement for this matrix, we introduce an invariant, called finite matrix factorization, and impose the lower bounds on any non-existent estimated imprinting element [7].

III. Proposed Method

One of the most important parts of the collaborative refinement algorithm is to determine the similarity between users. The correct choice of a similarity function is a critical factor in the refinement algorithm for determining the similarity between users, since it strongly influences the accuracy of the suggestions. One of the most commonly used criteria for obtaining similarities is the Pearson correlation coefficient. Studies have shown that Pearson's correlation coefficient is better than other similarity criteria [11]. This coefficient makes a linear relationship between two distinct variables and its value varies from -1 to +1. The value of +1 indicates the complete relationship between the two variables and the value of the non-relationship between the two variables. In other words, +1 indicates that two users have totally related interests. If the number -1 is a conflict of interest between two users, Pearson correlation coefficient is widely used as a similarity criterion in the proposer systems. This has some disadvantages as outlined below.

- Not counting the number of items in the calculation of similarity

- Failure to include the distance points in the calculation of similarity

One of the disadvantages of Pearson's correlation coefficient in calculating similarity is the effect of the number of items. Consider the 2 users in 5 common items have the same views that according to the Pearson method, the likeness of these 2 users is +1 obtained, and also assume that 2 users in the 100 common items have the same views, then the amount of likeness of these two users is also +1. In other words, Pearson correlation coefficient does not consider the number of items and the number of common items. To do this, we use the relation (1) as a coefficient to con

Table 1: is an example of a user-item matrix. Unknown rates are shown with *.

	Item1	Item2	Item3	Item4	Item5	Item6
User a	2	*	4	*	*	*
User b	2	1	4	1	2	3
User c	1	3	4	2	*	5
User d	2	1	5	4	2	*

These similarity criteria give the same value to each user. That is, these methods are based on the assumption that sim (a, b) = sim (b, a). However, as explained in the introduction, the similarity between the two users is expected to be asymmetrical. In the previous section, we analyzed the traits of traditional similarity criteria. To overcome the above deficiencies and improve the performance of the advisory systems, two factors are considered in similarity criteria.

A method for smoothing similarities is to assign asymmetric weights to traditional similarity criteria. The first proposed weighting factor considers the percentage of items ranked by an active user with others, and is based on the proportion of items with a common score among users, with the number of ranking active items that user normalized. The traditional similarity functions do not take into account the bias affecting the relationships between users (for example, the desire of specific users to give higher points to items). The second weighing factor attempts to find these similarities in the user rankings and to give the same similarity to users with the same scoring habits.

Definition 1 (First weighing factor) With respect to the two user u and v, the proposed weighting factor is shown as follows:

$$C(u,v) = \frac{|Iu \cap Iv|}{|I|} \qquad (1)$$

In which, Iu and Iv are collections of two-item items. And I is the total number of items.

Definition 2 (Weighting factor II) the proposed second weighing factor can be calculated as follows:

$$A(u,v) = \frac{\vec{V}_u . \vec{V}_v}{||\vec{V}_u|| . ||\vec{V}_v||} \qquad (2)$$

$\vec{V}_u$ Represents a vector based on the user u rating.

IV. PROPOSED METHOD FLOW CHART

In Figure 1, the work flow of the proposed method is shown. To predict the rating, the most similar people should be found in the active user, which is the same as the neighboring users, through the sequential formula (3). After finding similar users, the score is calculated by formula (2) and eventually the program finishes.

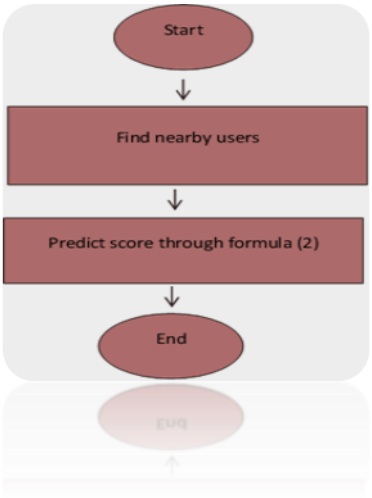

FIGURE 1 FLOW DIAGRAM OF THE PROPOSED METHOD

V. THE PSEUDOCODE ALGORITHM IS THE PROCESS OF MAKING SUGGESTIONS

In the following, the form of (2) quasi-code of the routine is also given. The matrix of the rank and test data sets are given as inputs to the system and predictive results are placed in the result matrix.

Algorithm1: Recommendation Procedure

Algorithm1: Recommendation Procedure

Input: R:RatingMatrix, T=Test Dataset
Output: Result
Begin For i=1:Size(T)
    Neighbors = Find Neighbors with Proposed Method from R
    P = Prediction Result from Neighbors
    Put P to Result
End

FIGURE 2: PSEUDOCODE ALGORITHM THE PROCESS OF DOING THE SUGGESTIONS.

VI. ANALYSIS AND EVALUATION OF PROPOSED METHOD

First, we introduce the datasets used to evaluate the performance of the proposed method, in the next section we will introduce the evaluation criteria, and finally we will show the results in the last section.

Test settings:

The tests performed in MATLAB software and in a system with the following characteristics:

4 GB RAM, 5 GHz core i5 processor at 3 gigahertz.

Data collection:

To evaluate the performance of the proposed method, the MovieLens100K dataset is used

Score Type	Scale Scores	Number of Scores	Item Type	Item Number	Number of Users	Datasheet
correct	+1 to +5	100000	Movie	1682	943	MovieLens100K

In the MovieLens100K dataset, the scores range from +1 to +5, with a score of +1 means no interest and a plus +5 means the highest interest in the movie. Also, users rated at least 20 videos [1, 7, 11, 32, and 33].

To examine the performance of the proposed method, the data are divided into two sets of educational data and experimental data, the training package consisting of 80% of the data and a test set of 20% of the data [11].

Evaluation criteria:

We used the Root Mean Squared Error (RMSE) method to evaluate the performance of recommended systems. The RMSE measures the difference between actual values and values obtained. The smaller the RMSE corresponds to the better predicted quality.

One common criterion for estimating the proposed systems is the mean squared error, which is defined as follows [7, 17].

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} |r_{a,i} - r_{p,i}|}{n}} \quad (3)$$

$r_{a,i}$ is the real score of user a to item i. $r_{p,i}$ is the predicted value of user a to item i and N is the number of predictions.

Results and analyzes of tests

In this section we report the results of the experiments and compare them with previous experimental work.

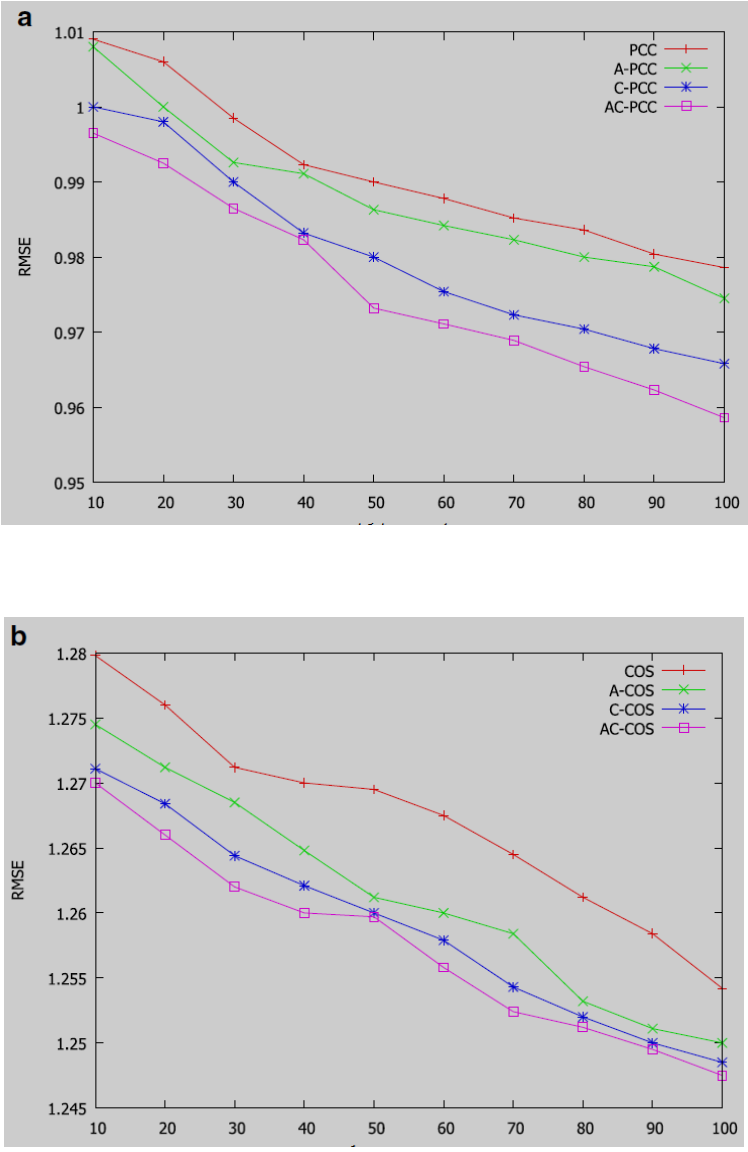

FIGURE 3: COMPARISON OF RMSE WITH (A) SIMILARITY OF PEARSON AND PEARSON WEIGHT (B) SIMILARITIES OF COS AND COS WEIGHT

VII. CONCLUSION

The proposed systems are intelligent systems that refine existing information on the internet by identifying the interests and priorities of each user and providing relevant suggestions to users. The most commonly used algorithm in the proposed systems is a collaborative refinement algorithm that has relatively better results than other proposing systems. The main idea of the collaborative refinement is that if two users have the same rating points on common items, then they have the same interest. Therefore, in this way, offers are made to the active user based on neighboring users. One of the most important parts of the proposed systems is the neighborhood finder, which can be greatly improved if properly selected. One of the ways to find neighbors is the use of similarity measurement metrics. Similarity measurement uses common point's privileges to calculate the similarity between active users and other users. Several similarity measurements have been reported to account for similarity in work. The traditional similarity criteria have disadvantages such as not counting the number of common items, not taking into account the distance of points. In the proposed method, these disadvantages are overcome and a new benchmark has been introduced using a similar weighting method to find similarity between two users. This research first showed that the traditional similarity measures have disadvantages. To overcome these weaknesses, two weighty factors were presented. The first factor, contrary to the traditional similarity criteria, assigns an asymmetric value between two users, which causes the number of common items to be specified between the two users. The second factor is the behavioral effects associated with ranked items as a factor in measuring the similarity between users, which also takes into account the distance between points. The results of the experiments showed that weight factors can greatly improve the performance of the recommended systems. This result is confirmed with the MovieLens dataset.

REFERENCES

[1] Zhang, F., Gong, T., Lee, V. E., Zhao, G., Rong, C., & Qu, G. (2016). Fast algorithms to evaluate collaborative filtering recommender systems. Knowledge-Based Systems, 96, 96-103.

[2] Liu H, Hu Z, Mian AU, Tian H, Zhu X (2014) A new user similarity model to improve the accuracy of collaborative filtering.Knowl-Based Syst 56:156–166

[3] Choi, S. M., Ko, S. K., & Han, Y. S. (2012). A movie recommendation algorithm based on genre correlations. Expert Systems with Applications, 39(9), 8079-8085

[4] Pham XH, Jung JJ (2014) Recommendation system based on multilingualentity matching on linked open data. J Intell Fuzzy Syst 27(2):589–599

[5] Rahul Katarya, Om Prakash Verma, An effective collaborative movie recommender system with cuckoo search, In Egyptian Informatics Journal, Volume 18, Issue 2, 2017, Pages 105-112, ISSN 1110-8665

[6] Cleger-Tamayo, S., Fernández-Luna, J. M., & Huete, J. F. (2012). Top-N news recommendations in digital newspapers. Knowledge-Based Systems, 27, 180-189.

[7] Koohi, H., & Kiani, K. (2016). User based Collaborative Filtering using fuzzy C-means. Measurement, 91, 134-139.

[8] Massa P, Avesani P (2009) Trust metrics in recommender systems. In: Computing with social trust. Springer, pp 259–285

[9] Garcia, I., Sebastia, L., & Onaindia, E. (2011). On the design of individual and group recommender systems for tourism. Expert systems with applications, 38(6), 7683-7692.

[10] Jamali M, Ester M (2009) Trustwalker: a random walk model for combining trust-based and item-based recommendation. In: Proceedings ACM SIGKDD (2009), 15th international conference on knowledge discovery and data mining. ACM, pp 397–406

[11] Carrer-Neto, W., Hernández-Alcaraz, M. L., Valencia-García, R., & García-Sánchez, F. (2012). Social knowledge-based recommender system. Application to the movies domain. Expert Systems with applications, 39(12), 10990-11000.

[12] Li, Q., Myaeng, S. H., & Kim, B. M. (2007). A probabilistic music recommender considering user opinions and audio features. Information processing & management, 43(2), 473-487

[13] Nilashi, M., & Ibrahim, O. B. (2014). A model for detecting customer level intentions to purchase in B2C websites using TOPSIS and fuzzy logic rule-based system. Arabian Journal for Science and Engineering, 39(3), 1907-1922.

[14] Al-Shamri, M. Y. H., & Al-Ashwal, N. H. (2014). Fuzzy-weighted similarity measures for memory-based collaborative recommender systems. Journal of Intelligent Learning Systems and Applications, 2014.

[15] Al-Shamri, M. Y. H. (2014). Power coefficient as a similarity measure for memory-based collaborative recommender systems. Expert Systems with Applications, 41(13), 5680-5688.

[16] Chang, C. C., & Chu, K. H. (2013, June). A recommender system combining social networks for tourist attractions. In Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on (pp. 42-47). IEEE.

[17] Bobadilla J, Ortega F, Hernando A, Bernal J (2012) A collaborative filtering approach to mitigate the new user cold start problem.Knowl-Based Syst 26(1–2):225–238

[18] Gorgoglione, M., & Panniello, U. (2009, May). Including context in a transactional recommender system using a pre-filtering approach: Two real e-commerce applications. In Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on (pp. 667-672). IEEE.