Indian Journal of Science and Technology, *Vol 9(29), DOI: 10.17485/ijst/2016/v9i29/91060, August 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Performance Comparison of Different Similarity Measures for Collaborative Filtering Technique

K. G. Saranya*, G. Sudha Sadasivam and M. Chandralekha

Department of Computer Science and Engineering, P. S. G College of Technology, Coimbatore - 641004, Tamil Nadu; India; saranyaa87@gmail.com, sudhasadhasivam@yahoo.com, rdbmchandralekha@gmail.com

Abstract

Objectives: As the plenty of Web services on the Internet increases, developing efficient techniques for Web service recommendation has become more significant. The main objective of this paper is to compare and study the drawbacks of the performance of different existing similarity measures against the proposed similarity measure that use the concept of collaborative filtering technique. **Methods/Analysis:** Collaborative filtering has turned into one of the most used technique to give personalized services for users. The key of this technique is to find alike users or items using user-item rating matrix such that the system can show recommendations for users. Experiments on Web Service (WSDL) data sets are conducted and compared with many traditional similarity measures namely Pearson correlation coefficient, JacUOD, Bhattacharyya coefficient. The result shows the superiority of the proposed similarity model in recommendation performance. **Findings:** However, existing approaches related to these techniques are derived from similarity algorithms, such as Pearson correlation coefficient, mean squared distance, and cosine. These methods are not much efficient, particularly in the cold user conditions. **Applications/Improvement:** This paper presents a new user based similarity calculation model to enhance the recommendation performance and to estimate the similarities for each user. The proposed model incorporates two traditional similarity measures namely Pearson Correlation Coefficient and Jaccard Coefficient.

Keywords: Collaborative Filtering, Recommendation System, Similarity Measures, Web Service

1. Introduction

In recent days people have their own smart phones, tablet PC's and other handy terminals like palmtops and so they spend more time in surfing all kinds of social networking media (such as G+, Facebook, etc.) and e-commerce sites (such as Myntra, Flipkart, etc). The voluminous information available makes them overwhelmed and indecisive. Users spend much time and energy in probing for their anticipated information. Still, they do not get acceptable outcomes. Luckily, the user preferences can be recorded for latter reference on the social networking sites and e-commerce sites, which makes easier to study the behavior of users. Recommender systems are used to suggest information of user anticipations and offer personalized services by analyzing the user's behaviors, for instance, the recommendation of the products in Amazon, photos in Flickr, and results in the query based Web search.

The collaborative filtering has become the most frequently used method to suggest items for users. It makes suggestion in accordance to similar users with the active user or the similar items with the items which are rated by the active user. The collaborative filtering includes model-based method and memory-based method. The model-based method first defines a model to explain the interest of users and, consequently to forecast the ratings of items. The memory-based method first defines the similarities among users and then selects the most similar users as the neighbors of the user to make recommendation. Finally, it gives the suggestions according to the neighbors. The memory-based method gives considerable recommended precision, but the computing time grows rapidly with the increasing number of items and users. In some circumstances, it is hard to take action in real-time. The model-based technique tends to be faster in prediction time than the memory-based technique, because

*Author for correspondence

the creation of the model can be completed in a considerable amount of time and this technique is executed off-line. The limitation of the model-based technique is that the recommendation performance is not as good for the memory-based technique. In addition to collaborative filtering, semantic recommendation, content-based technique, social recommendation are also applied in prediction of user preference.

This paper concentrates on the recommended performance in memory-based collaborative filtering algorithms. The core of collaborative filtering technique is to compute similarities among users or items. The generic traditional similarity measures, such as Pearson correlation coefficient, mean squared distance, and cosine are not enough to capture the effective similar users, particularly for cold user who only rates a small number of items. This paper presents a better heuristic similarity measure model. The new similarity model incorporates two similarity measures namely Pearson Correlation Coefficient and Jaccard Coefficient. In order to evaluate fy the new similarity measure, experiments are conducted on web service data set. In comparison with many state-of-the-art similarity measures, new model can show improved recommended performance and uses the better ratings in cold user conditions.

Collaborative Filtering (CF), as a category of personalized recommendation method, has been commonly used in variety of domains. Though, collaborative filtering suffers from a few issues, like cold start, data sparsity, scalability problems. These issues seriously lessen the user experience. This paper concentrates on how to get the better prediction accuracy. Collaborative filtering recommends items to users according to their preferences. Therefore, the past database of users' preference must be available. However, the database is always very sparse, that is, user only rates a lesser number of items. Up to now, there are many researchers who have focused on the prediction accuracy and proposed some solutions.

To improve the precision, many researchers have proposed some new similarity measures. A technique that does Recommendation System and Collaborative Filtering has been proposed[1]. With respect to the data sparsity issue, the approach of user-item based collaborative filtering algorithm is proposed along with an iterative technique and a three step updating algorithm to form a constant recommendation scheme. Finally, based on the scalability of the neighborhood size, cosine similarity is used to calculate a similarity among users. Advantages of this system are solves the data sparse problem and improves consistency. Disadvantage is that it is computed using less number of datasets. A technique that does Qos prediction, Time-aware and Web service has been proposed[2]. The similarity measure used is adjusted Cosine-based similarity. Advantages of this system are to improve prediction accuracy and missing value prediction for QoS. Disadvantages are it does not include many QoS factors and relationships among QoS factors into consideration and it does not incorporate QoS factors into QoS prediction. A technique that does Greedy Filtering, K-nearest neighbor graph and Fast collaborative filtering has been proposed[3]. The similarity measures used are Pearson Correlation Coefficient (PCC) and adjusted Cosine similarity. Advantages of this system are it decreases the execution time and improve the recommendation quality. Disadvantage is that it is computationally expensive. A technique that does Novel approach has been proposed[4]. The similarity measure used is Pearson Correlation Coefficient (PCC). An advantage of this system is predictions are of high precision. Disadvantages are that if the number of users and items become huge, a huge amount of time will be consumed. A technique that does K-nearest neighborhood and K-means has been proposed[5]. The similarity measure used is Pearson Correlation Coefficient (PCC). Advantages of this system are it improves the accuracy, improves lower time consuming level, solves the cold start issue, and solves time and space complexity. Disadvantage is that it is difficult to access user profile. A technique that does Novel algorithm has been proposed[6]. The similarity measure used is cosine similarity. Advantages of the system are it is more robust and it improves prediction accuracy. Disadvantages are it has incomplete comparisons with the previous methods and the analysis of the fusion model is reduced. A technique that does QoS-aware ranking-oriented hybrid Web service recommendation approach has been proposed[7]. The similarity measure used is Pearson Correlation Coefficient (PCC). Advantages of this system are it has higher accuracy rate, predicts the missing QoS values in a given dataset and improves interpretability. Disadvantages are it is computationally expensive and involves more mathematical formulas. A technique that does Context-aware approach, which is a cloud based mobile multimedia has been proposed[8]. The similarity measure used is Pearson Correlation Coefficient (PCC). Advantages of this system are it is used to develop a real-world applications and it improves services provided by service providers. Disadvantage is that it is restricted to the relatively small datasets. A technique that does DBSCAN clustering algorithm[9] is used to perform a clustering on set of items, and then obtains the user's prediction rating of the target item using weighted slope one scheme. The similarity measures used are Pearson Correlation Coefficient (PCC) and Cosine similarity. Advantages of this system are it improves accuracy, solves the problem of sparsity, scalability and cold start and it is more robust to noise. Disadvantage is that it considers only limited number of datasets. A technique that does TrustSVD, a trust-based matrix factorization method has been proposed[10]. This technique considers both explicit and implicit ratings and trust information during rating prediction on unknown items. A weighted-λ-regularization technique was adapted and used to further regularize the user- and item-specific latent feature vectors studied and the similarity measure used is Pearson Correlation Coefficient. Advantages of this system are it solves the problem of data sparsity and cold start and it is outperformed in predictive accuracy. Disadvantage is that it does not consider the influence of trusters and trustees. A technique that does Ensemble Method has been proposed[11]. The similarity measures used are modified Pearson Correlation Coefficient (PCC) and modified Cosine-based similarity measures. Advantages of this system are it has less computational cost and linear space complexity, running time complexity. Disadvantage is that it does not focus on the application of the ensemble methods. A technique that does Behavior Factorization has been proposed[12]. The similarity measure used is Jaccard similarity. Advantage of this system is it improves the performance. Disadvantages are it does not work well when users have very sparse or no data and it concentrate on social media platform like Google+. A technique that does the comparison of least mean square algorithm and fractional least mean square has been proposed[13]. It is observed that fractional LMS has proved very well in case of deterministic signal because of the higher rate of convergence and smaller amount of errors occur in it though LMS algorithm has better performance for random signals. A technique that does Karl's Pearson Coefficient (KPC) has been proposed[14]. In order to make the system personalized, Felder's learning styles catalogue is applied and to build many e-learning systems where the reliability of their recommended learning styles are analyzed using KPC.

2. The New Similarity Model

This section presents the drawbacks of the existing similarity measures. Then, it introduces the motivation and hypothesis of the proposed similarity measure approach. Finally, this paper presents the mathematic formalization of the proposed novel similarity measure approach. In this system, Web service users are represented as $U = \{u_1, u_2, \dots, u_m\}$ m represents the total number of users; set of web service items with similar functionality are represented as $I = \{i_1, i_2, \dots, i_n\}$ where n represents total number of web services.

2.1 The Disadvantages of Existing Similarity Measures

The Pearson Correlation Coefficient (PCC), Jaccard Uniform Operator Distance (JacUOD) and Bhattacharyya coefficient are the most widely used similarity measures in collaborative filtering.

2.1.1 Pearson Correlation Coefficient (PCC)

In many recommendation systems, Pearson Correlation Coefficient (PCC)[15] measure has been applied to compute the similarity between the users and items. PCC measure based similarity between two users is computed using the following formula

$$pcc(u,v) = \frac{\sum_{i \in I_v \cap I_u} (r(u,i) - \bar{r}(u))(r(v,i) - \bar{r}(v))}{\sqrt{\sum_{i \in I_v \cap I_u} (r(u,i) - \bar{r}(u))^2}\sqrt{\sum_{i \in I_v \cap I_u} (r(v,i) - \bar{r}(v))^2}} \quad (1)$$

where $I_v \cap I_u$ represents co-invoked set of web services by user v and u. $\bar{r}(v)$ and $\bar{r}(u)$ represent the average QoS values of all the web services invoked by user u and v respectively. The similarity value computed using above equation falls within the range of [-1, 1]. The larger the similarity value represents, that two uses are more similar to each other. However, the main draw back of this equation is that it does not consider the personal influence of web services on similarity calculation. i.e., Co-invoked web services are given equivalent weights in the computation of similarity between two users. Therefore, a weighted PCC has been developed which incorporates the personal influences of web services into similarity computation between two users. Weight of web service i based on QoS deviation is calculated using the following steps.

- **QoS Normalization:** This step transforms each QoS value of web service i, r (u, i), to a real number between

0 and 1. This could be done by comparing it with the maximum and minimum QoS values of web service i. Here two cases are to be considered. If the QoS criterion concerned is positive then r(u,i) is normalized using Equation (2);, if the QoS criterion is negative then r(u,i) is normalized using Equation (3).

$$n(u,i) = \frac{r(u,i) - min\, r(i)}{maxr(i) - minr(i)} \quad (2)$$

$$n(u,i) = \frac{maxr(i) - r(u,i)}{maxr(i) - minr(i)} \quad (3)$$

where set of QoS values of web service I is represented as r (i). n(u,i) is set to 1, in the case of $\max r(i) = \min r(i)$.

- **Computation of Standard Deviation using Normalized QoS Values:** This is computed using the following formula

$$d_i = \begin{cases} \sqrt{\sum_{u \in U_i} (n(u,i) - \bar{n}(i))^2 / |U_i|}, & if\,|U_i| \geq \theta \\ \sqrt{\sum_{u \in U_i} (n(u,i) - \bar{n}(i))^2 / |U_i| \times \frac{|U_i|}{\theta}}, & if\,|U_i| < \theta \end{cases} \quad (4)$$

where $\bar{n}(i)$ is the average QoS value of Web service i,θ is a threshold for the number of users that have invoked i, i.e., U_i. If U_i is very small, the standard deviation is likely to be overestimated by the original standard deviation computation formula. The θ is used to address the above issue.

- **Weigh Generation:** the weight of a Web Service i is obtained using the following formula.

$$w_i = d_i \quad (5)$$

The value of weight is always in the range (0, 1).

After weight generation, the similarity between user u and v is computed using the following formula.

$$sim(u,v) = \frac{\sum_{i \in I_v \cap I_u} w_i (r(u,i) - \bar{r}(u))(r(v,i) - \bar{r}(v))}{\sqrt{\sum_{i \in I_v \cap I_u} w_i (r(u,i) - \bar{r}(u))^2} \sqrt{\sum_{i \in I_v \cap I_u} w_i (r(v,i) - \bar{r}(v))^2}} \quad (6)$$

The above formula incorporates both the personal influence of Web services and user rating value during user similarity measurement. It implies that the weights of the web services with larger values will contribute more during the similarity computation between two users.

In the next step, similarity between the web services I ad j is calculated using the standard PCC measure and the same is expressed in the following formula.

$$pcc(i,j) = \frac{\sum_{u \in U_i \cap U_j} (r(u,i) - \bar{r}(i))(r(u,j) - \bar{r}(j))}{\sqrt{\sum_{u \in U_i \cap U_j} (r(u,i) - \bar{r}(i))^2} \sqrt{\sum_{u \in U_i \cap U_j} (r(u,j) - \bar{r}(j))^2}} \quad (7)$$

Moreover, this approach provides low similarity value regardless of the similar ratings made by two users on items and if the co-rated items present in the user-item rating matrix is very few, then it will not provide a reliable similarity value.

The working principle for Pearson Correlation Coefficient is computed using the following formula

$$sim(i,j) = \frac{\sum_{u \in U_i \cap U_j} w_u (r(u,i) - \bar{r}(i))(r(u,j) - \bar{r}(j))}{\sqrt{\sum_{u \in U_i \cap U_j} w_u (r(u,i) - \bar{r}(i))^2} \sqrt{\sum_{u \in U_i \cap U_j} w_u (r(u,j) - \bar{r}(j))^2}} \quad (8)$$

$U_i \cap U_j$ = Set of users invoked both web services i and j.
r(u,i) = web service i's QoS value.
r(u,j) =web service j's QoS value.
$\bar{r}(i)$ = web service i's average QoS value.
$\bar{r}(j)$ = web service j's average QoS value.
w_u = weight associated with user u (Standard deviation of the normalized QoS values of web services invoked by user u).

When the similarity measure is calculated, the following matrix is obtained from Table 1.

Table 2 denotes the final values when the similarity measure is calculated using Pearson Correlation Coefficient (PCC)

2.1.2 *Jaccard Uniform Operator Distance (JacUOD)*

JacUOD[16] approach investigates the characteristics of similarity measurement for different multidimensional

Table 1. Example user-item rating matrix where the value 99 corresponds to null (not rated)

Users\ Items	Item1	Item2	Item3	Item4	Item5
User1	-7.82	8.79	-9.66	-8.16	-7.52
User2	4.08	-0.29	6.36	4.37	-2.38
User3	99	99	99	99	9.03
User4	99	8.35	99	99	1.8
User5	8.5	4.61	-4.17	-5.39	1.36

Table 2. From the user similarity matrix in Table 1, applying PCC

Users\ Users	User 1	User 2	User 3	User 4	User 5
User1	1	-0.18665	0.154549	0.380217	0.298611
User2		1	-0.21076	-0.05925	-0.14916
User3			1	0.306739	0.239582
User4				1	0.360049
User5					1

vector spaces, and leads to better prediction accuracy. JacUOD approach can be adopted as a promising approach for hybrid recommender systems to provide more accurate similarity measurement. Data sparsity in user profile decreases the performance and quality of any recommender systems. This similarity measurement only focuses on rating-based collaborative filtering approaches. Moreover, JacUOD approach is ineffective for ranking-based collaborative filtering approaches and also suffers from few or no overlapping items.

The working principle for JacUOD is computed using the following formula

$$sim(u,v) = \begin{cases} \frac{|S_{uv}|}{|S_u \cup S_v|} \times \frac{\sqrt{m\,(V_{max}-V_{\min})^2}}{\sqrt{\sum_{s\in S_{uv}}(r_{u,s}-r_{v,s})^2}}, & \text{if } \exists s \in S_{uv}, r_{u,s} \neq r_{v,s}, \\ \frac{|S_{uv}|}{|S_u \cup S_v|} \times \frac{\sqrt{m\,(V_{max}-V_{\min})^2}}{0.9+\sqrt{\sum_{s\in S_{uv}}(r_{u,s}-r_{v,s})^2}}, & \end{cases} \quad (9)$$

$|S_{uv}|$ = number of common ratings made by both users u and v

$|S_u \cup S_v|$ = number of items rated by both users u and v

m = cardinality value of $|S_{uv}|$

V_{max} = maximum rating value given by user u

$V_{\min}$ = minimum rating value given by user u

$r_{u,s}$ = rating value of user u on item s

$r_{v,s}$ = rating value of user v on item s

When the similarity measure is calculated, the following matrix is obtained from Table 1.

Table 3 denotes the final values when the similarity measure is calculated using Jaccard Uniform Operator Distance (JacUOD)

2.1.3 Bhattacharyya Coefficient

The formula for Bhattacharyya coefficient is computed using the following formula

$$BC(u,v) = \sum_{i=1}^{n} \sqrt{P_{u,i} P_{v,i}} \quad (10)$$

Table 3. From the user similarity matrix in Table 1, applying JacUOD

Users\ Users	User 1	User 2	User 3	User 4	User 5
User1	1	0.247754	0.230845	0.305642	0.307504
User2		1	0.256345	0.279499	0.389933
User3			1	0.441945	0.353995
User4				1	0.398219
User5					1

n = number of partitions.

$P_{u,i}$ = ratio of the number of items with rating value i by user u to the total number of items rated by user u.

$P_{v,i}$ = ratio of the number of items with rating value i by user v to the total number of items rated by user v.

The working principle for Bhattacharyya coefficient[17] is discussed below

Let $u_1 =$ (1,0,2,0,1,0,2,0) and $u_2 =$ (0,1,0,2,0,1,0,2) be the rating vectors of user u_1 and u_2.The ratings lie in {1,2}, hence the number of partitions is 2. BC coefficient between user u_1 and u_2 is computed the following formula

$$BC(u_1,u_2) = \sum_{i=1}^{2} \sqrt{P_{u,i} P_{v,i}} \quad (11)$$

$$= \sqrt{\frac{2}{4} \times \frac{2}{4}} + \sqrt{\frac{2}{4} \times \frac{2}{4}} = 1$$

It can be noted that there is no common rated items between u_1 and u_2. This measure could not calculate user similarity in this situation. It can also be noted that u_1 and u_2 both have preferences for giving low ratings. In addition to that, two users have an identical rating distribution which can be inferred that u_1 and u_2 are similar in rating habits. Let u_3 = (0, 5, 0, 4, 0, 5, 0, 3) be the rating vector of user u_3. A problem occurs where BC coefficient between u_1 and u_3 equals 0 because there is no overlap at all in every partition and still have certain similarities as their ratings are relatively centralized in distribution. Moreover, this approach cannot be used to find a similarity between pair of users if they rate on few or no similar items, also not scalable and computation is very complex. When the similarity measure is calculated, the following matrix is obtained from Table 1.

Table 4 denotes the final values when the similarity measure is calculated using Bhattacharyya Coefficient (BC)

2.2 Discussions on the New Similarity Measure Model

The new similarity measure incorporates Pearson Correlation Coefficient (PCC) and Jaccard Coefficient. The mathematical formalization of the proposed novel similarity measure can be calculated using the following formula:

$$W * PCC + W * JCC \quad (12)$$

Table 4. From the user similarity matrix in Table 1, applying Bhattacharyya coefficient

Users\ Users	User 1	User 2	User 3	User 4	User 5
User1	1	0.86023206	0.7473063	0.73827124	0.889582
User2		1	0.6999996	0.67838615	0.9539392
User3			1	0.8247859	0.7038481
User4				1	0.7111426
User5					1

$$PCC(i,j) = \frac{\sum_{u \in U_i \cap U_j}(r(u,i)-\bar{r}(i))(r(u,j)-\bar{r}(j))}{\sqrt{\sum_{u \in U_i \cap U_j}(r(u,i)-\bar{r}(i))^2}\sqrt{\sum_{u \in U_i \cap U_j}(r(u,j)-\bar{r}(j))^2}} \quad (13)$$

$U_i \cap U_j$ = set of users invoked both web service i and j.
r (u, i) = web service i's QoS value.
r (u,j) = web service j's QoS value.
$\bar{r}(i)$ = web service i's average QoS value.
$\bar{r}(j)$ = web service j's average QoS value.

$$JCC = \frac{U_1 \cap U_2}{U_1 \cup U_2} \quad (14)$$

When the similarity measure is calculated, the following matrix is obtained from Table 1.

Table 5 denotes the final values when the similarity measure is calculated by incorporating Pearson Correlation Coefficient (PCC) and Jaccard Coefficient.

First, from the above matrix we can see that the similarity between User 1 and User 3 is higher when compared to the similarity between User 1 and User 2. However, this is not accurate in PCC, Jaccard and Bhattacharyya coefficient. This indicates that the new similarity measure model is able to overcome the drawback of low similarity regardless of similar ratings by two users.

Second, the similarity between User 3 and User 5 is also higher than the similarity between User 4 and User 5. However, the misleading still exists in PCC, Jaccard and Bhattacharyya coefficient similarity. This demonstrates that the new similarity measure can avoid the misleading.

Table 5. From the user similarity matrix in Table 1, applying combined similarity measure

Users\ Users	User 1	User 2	User 3	User 4	User 5
User1	1	0.02089	0.05520	0.00475	0.02440
User2		1	0.0183	0.00464	0.03561
User3			1	0.00636	0.02500
User4				1	0.01531
User5					1

Third, each user becomes comparable, that is each user has different similarities. This can be seen in the above matrix. Each pair has different similarities. However, this is not the case in existing similarity measures. This also can be seen from the above matrix.

3. Results and Discussion

3.1 Data set

The data set of web service (www.wsdream.com) is used in our experiments. This data set consists of 339 users and 5825 web services as user – item matrix. This matrix also includes trough put and response time for each web service. This Web service dataset is used for web service recommendation system. 80% of users are used for training while 20% is used for testing.

3.2 Evaluation Metrics

The performance of the new similarity measure is evaluated using two metrics called precision and recall. The main draw back of using these measures is that if number of items increases in the top-N recommendation list then recall increases while the precision decreases. Therefore, F-Measure which combines precision and recall is used to measure the accuracy of predicting number of nearest neighbors and performance of the recommendation system.

$$FMeasure = 2 * (Precision \times Recall | Precision + Recall)$$

Experiments were conducted on web service data set and the proposed similarity measure is compared with other traditional similarity measures. K-Neighbors and the number of recommendations are the two parameters which can impact the performance of recommendation systems. The results are compared with different values of these two parameters.

3.2.1 Performance of different Similarity Measures on Web Service Data Set

3.2.1.1 K-Neighbors

When the k value increases, the precision value increases while the recall values get increases as shown in the Figure 1, Figure 2, Figure 3.

3.2.1.2 Number of Recommendations

When the k value increases, the precision value increases while the recall values get increases as shown in the Figure 4, Figure 5, Figure 6.

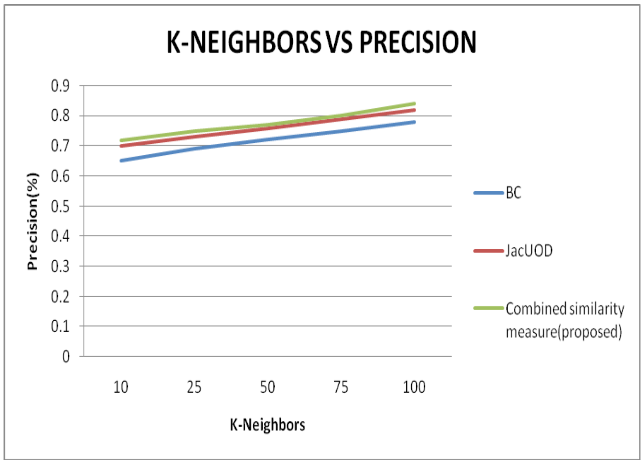

Figure 1. Comparison of precision against k-neighbors on web service data set.

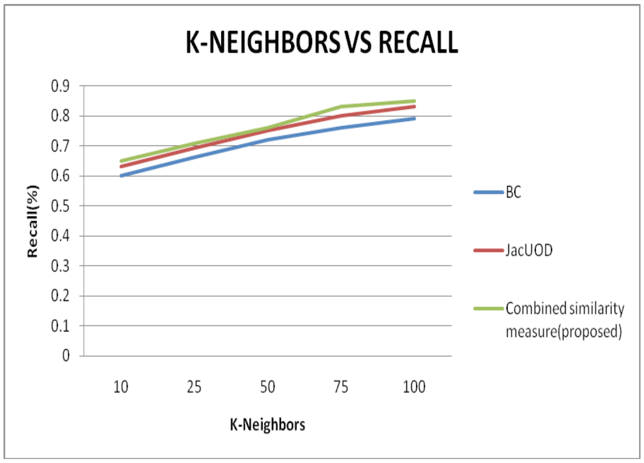

Figure 2. Comparison of recall against k-neighbors on web service data set.

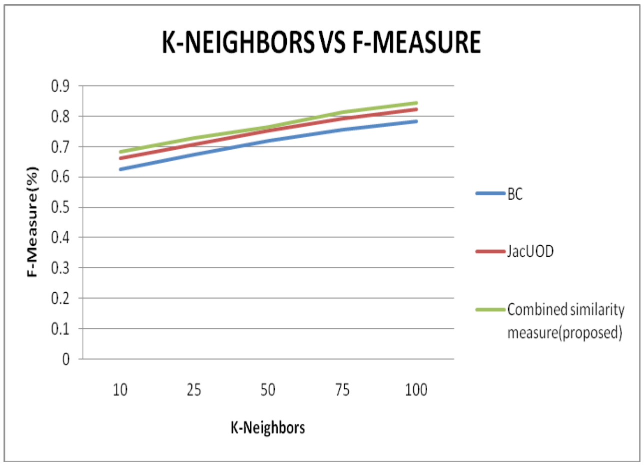

Figure 3. Comparison of F-measure against k-neighbors on web service data set.

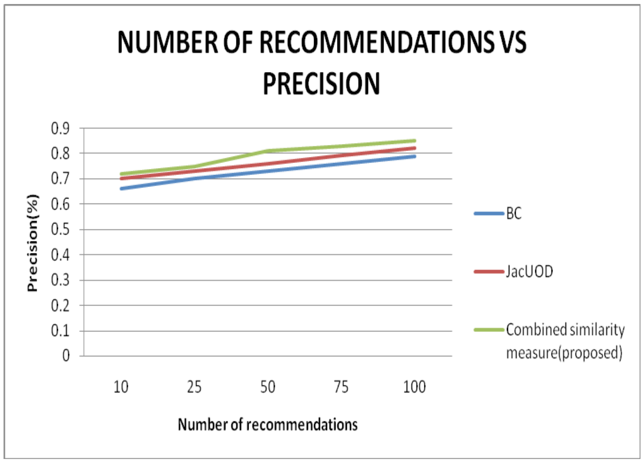

Figure 4. Comparison of precision against number of recommendations on web service data set.

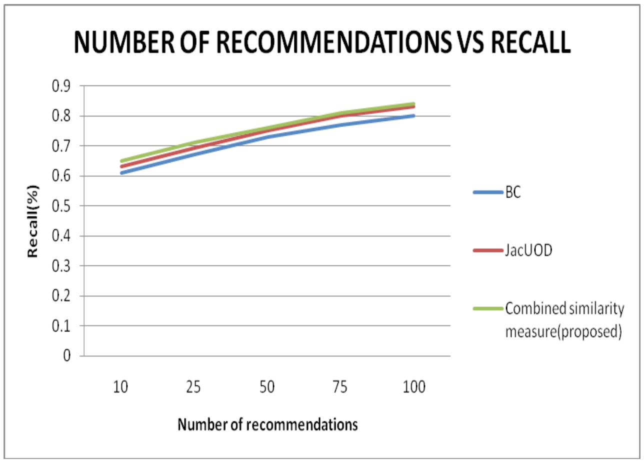

Figure 5. Comparison of recall against number of recommendations on web service data set.

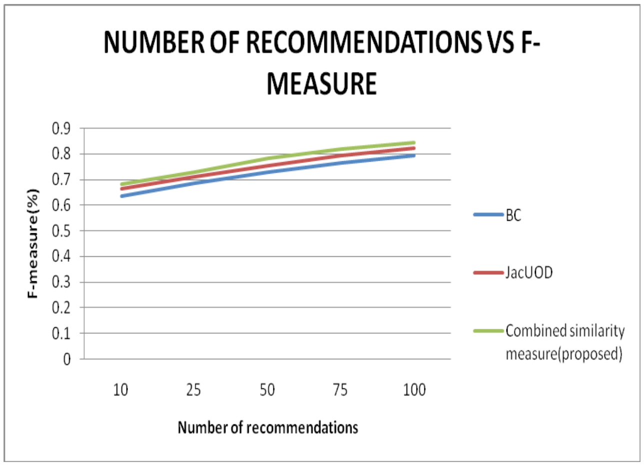

Figure 6. Comparison of F-measure against number of recommendations on web service data set.

4. Conclusion

The paper first analyzes the disadvantages of the existing similarity measures. In order to deal all these shortages, a novel similarity measure approach which combines PCC and Jaccard is proposed. Experiments were conducted on web services data set to demonstrate the performance of the new similarity measure. Experimental results shows the effectiveness of the novel similarity measure and it can overcome the drawbacks of the traditional similarity measures.

5. References

1. Luo B, et al. Recommendation scheme via improved iteratively collaborative filtering algorithm with neighborhood scale research. 2013 5th International Conference on IEEE Computational and Information Sciences (ICCIS); Shiyang. 2013 Jun 21-23. p. 609-12. DOI: 10.1109/ICCIS.2013.167.
2. Yu C, Huang L. Time-aware collaborative filtering for qos-based service recommendation. 2014 International Conference on IEEE Web Services (ICWS); Anchorage, AK. 2014 Jun 27-Jul 2. p. 265–72. DOI: 10.1109/ICWS.2014.47.
3. Park Y, et al. Fast collaborative filtering with a k-nearest neighbor graph. 2014 International Conference on IEEE Big Data and Smart Computing (BIGCOMP); 2014. DOI: 10.1109/BIGCOMP.2014.6741414.
4. Mase H, Hayato O. A collaborative filtering incorporating hybrid-clustering technology. 2012 International Conference on IEEE Systems and Informatics (ICSAI); Yantai. 2012 May 19-20. p. 2342–6. DOI: 10.1109/ICSAI.2012.6223524.
5. Xiong L, et al. A novel nearest neighborhood algorithm for recommender systems. 2012 3rd Global Congress on IEEE Intelligent Systems (GCIS); Wuhan. 2012 Nov 6-8. p. 156–9. DOI: 10.1109/GCIS.2012.58.
6. Wang J, De Vries AP, Reinders MJT. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. ACM Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval; 2006. DOI: 10.1145/1148170.1148257.
7. Chen M, Ma Y. A hybrid approach to web service recommendation based on QoS-aware rating and ranking. arXiv preprint arXiv:1501.04298; 2015.
8. Lee G-Y, Tseng W-P. An enhanced memory-based collaborative filtering approach for context-aware recommendation. Proceedings of the World Congress on Engineering; 2015.
9. Haipeng, et al. An improved collaborative filtering recommendation algorithm combining item clustering and slope one scheme. Proceedings of the International Multi Conference of Engineers and Computer Scientists; 2015.
10. Guo G, Zhang J, Yorke-Smith N. TrustSVD: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI); 2015.
11. Bar A, et al. Improving simple collaborative filtering models using ensemble methods. Multiple Classifier Systems. Springer Berlin Heidelberg. 2013; 7872:1–12. DOI: 10.1007/978-3-642-38067-9_1.
12. Zhao Z, et al. Improving user topic interest profiles by behavior factorization. Proceedings of the 24th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee; 2015. DOI: 10.1145/2736277.2741656.
13. Imran R. Adaptive filtering algorithms for channel equalization in wireless communication. Indian Journal of Science and Technology. 2015 Aug; 8(17). DOI: 10.17485/ijst/2015/v8i17/57805.
14. Muruganandam S, Srinivasan N. Appraisal of felder - silverman learning style model with discrete data sets. Indian Journal of Science and Technology. 2016 Mar; 9(10). DOI: 10.17485/ijst/2016/v9i10/88992.
15. Liu J, Tang M. Zheng Z. Liu XF, Lyu S. Location-aware and personalized collaborative filtering for web service recommendation. IEEE Transactions on Services Computing. 2015. DOI: 10.1109/TSC.2015.2433251.
16. Li M, Kai Z. A collaborative filtering algorithm combined with user habits for rating. International Conference on Logistics Engineering, Management and Computer Science; 2015.
17. Sun H-F, et al. JacUOD: A new similarity measurement for collaborative filtering. Journal of Computer Science and Technology. 2012. DOI: 10.1016/j.ins.2007.07.024.