



A New Similarity Measure for User-based Collaborative Filtering in Recommender Systems

T.Srikanth , M.Shashi

Department of Computer Science and Engineering,
Institute of Technology, GITAM University,
Visakhapatnam, Andhra Pradesh, India.

Email:talk2srk@yahoo.com

Department of Computer Science and Systems Engineering,
College of Engineering, Andhra University,
Visakhapatnam, Andhra Pradesh, India.
Email:smogalla2000@yahoo.com

ABSTRACT

Collaborative filtering is a popular approach in recommender Systems that helps users in identifying the items they may like in a wagon of items. Finding similarity among users with the available item ratings so as to predict rating(s) for unseen item(s) based on the preferences of likeminded users for the current user is a challenging problem. Traditional measures like Cosine similarity and Pearson correlation's correlation exhibit some drawbacks in similarity calculation. This paper presents a new similarity measure which improves the performance of Recommender System. Experimental results on MovieLens dataset show that our proposed distance measure improves the quality of prediction. We present clustering results as an extension to validate the effectiveness of our proposed method.

KEYWORDS

Recommender Systems; Collaborative Filtering; Similarity Measure; Cosine Similarity; Pearson Correlation; Clustering; user-based Collaborative Filtering; Cluster Purity; Similarity

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol.14, No. 9

www.ijctonline.com, editorijctonline@gmail.com



I Introduction

In the age of digital erudition, glut of massive data is generated in every field of science and Technology due to availability of automated tools and techniques for data generation and data collection. Research of the day is to uncover the obscure information from the unrefined data.

Recommender systems call for Intelligent Information Retrieval Techniques to provide a solution to the problem of triumphant information search by applying the practice of knowledge detection in the available colossal data to provide individual personalized recommendations. Recommender systems can be described as services for suggesting a list of products to people who might tend to like the same.

The extent of liking of a product by a user is termed as rating in Recommender systems. e.g., Bob gave a rating of 4 (out of 5) to the movie "Iron Man".

Recommender systems are usually classified into the following categories, based on how recommendations are made [1]

Collaborative recommendation

The user will be recommended items based on recommendations of other people who have similar ratings history as that of current user.

Content-based recommendation

The user will be recommended items based on a comparison between the descriptions of the items and a profile of the user that assigns weight to the characteristics of the item.

Hybrid approaches

The user will be recommended items based on a combination of Content-Based and Collaborative methods to overcome the limitations of both the methods.

Collaborative filtering (CF) is a popular approach in recommender Systems. [6]

Collaborative filtering approaches known as social filtering [18] is to make automated prophecy of the preferences of item(s) for an active user based on the user's earlier likings and the opinions of other users in the closest vicinity of the user referred to as nearest neighbors.

Similarity Measure plays central role in identifying the nearest neighbors when k-nn approach is used in Collaborative Filtering [20]. Cosine similarity and Pearson correlation are widely used for similarity calculation in collaborative filtering [7].

We propose a new similarity measure which performs better than Cosine similarity and Pearson correlation. We tested the proposed method on the MovieLens 100k dataset and as an extension we applied our proposed measure for testing cluster purity on famous Iris dataset and Wine dataset.

The remainder of this paper is organized as follows. In section II review on previous work is presented. In Section III we present our proposed method. In section IV experimentation with Results are discussed followed with conclusion in section V.

II Related Work

Collaborative filtering algorithms can be grouped into two categories; model-based and memory-based [13].

Model-based techniques use data mining and machine learning algorithms to train a model using known data and then the model will be used to make predictions for real data. Some of the successful model-based CF techniques include clustering [4], matrix factorization [21], Dimensionality reduction techniques such as SVD is used in CF to deal effectively with the data sparsity and scalability problems [3]. principal component analysis which transforms an original set of variables into a smaller set of uncorrelated variables is used along with clustering to construct a model [15]. A detailed survey can be found in [22]

Memory-based algorithms operate on the entire database. It tries to find the vicinity of the active user based on the concord of his past ratings and uses their bias to guess ratings for new items. Memory-based algorithms can be further divided into user-based [11], item-based [2] and unification of both [14].

The same set of similarity estimation techniques is applicable for both user-based and item-based CF systems for finding nearest neighbors. Memory-based techniques are more popular due to their simplicity and proven results.

In this paper a new Metric for estimating the similarity is proposed and its applicability to user-based CF is established with by applying it on benchmark data.

Distance/Similarity Measures

The Scenario of recommender systems with n users and m items is represented by a $n \times m$ Rating matrix consisting of elements $r_{u,i}$ indicating the ratings made by user u for item i . The similarity between two users is calculated using one of the traditional estimate namely Cosine similarity, Pearson correlation, Spearman's rank correlation, entropy, mean squared difference etc.



Cosine similarity and Pearson correlation are extensively used for finding similarity between pair of items and pair of users respectively in Recommender Systems [7]. Pearson correlation performs better than Cosine similarity in measuring user - user affinity [13] and also the other estimates such as Spearman's rank correlation, entropy, means squared difference in Collaborative Filtering [12]. A detailed survey on different Distance or Similarity Measures is given in [5].

Cosine similarity

Cosine similarity angle between two users rating vectors is given as

$$\text{csim}(u, u') = \frac{\sum_{i \in I_u \cap I_{u'}} (r_{u,i}) (r_{u',i})}{\sqrt{\sum_{i \in I_u \cap I_{u'}} (r_{u,i})^2} \sqrt{\sum_{i \in I_u \cap I_{u'}} (r_{u',i})^2}}$$

Where I_u & $I_{u'}$ denotes the items rated by user u and u' . Cosine similarity normalizes the data with reference to the origin. Cosine similarity has a range from 0 and 1.

Pearson correlation

Pearson correlation between two users u and u' with common ratings is given as

$$\text{corr}(u, u') = \frac{\sum_{i \in I_u \cap I_{u'}} ((r_{u,i} - \bar{r}_u) (r_{u',i} - \bar{r}_{u'}))}{\sqrt{\sum_{i \in I_u \cap I_{u'}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_{u'}} (r_{u',i} - \bar{r}_{u'})^2}}$$

Where I_u & $I_{u'}$ denotes the items rated by user u and u' . Pearson correlation can also be viewed as Cosine similarity normalized by the offset of the corresponding objects to find the degree of linearity. Pearson correlation has a range from -1 to 1.

Pearson correlation is widely used in Recommender systems to identify users who exhibit linear relationship i.e., similar tastes. For example consider the Table 1 that displays a small hypothetical rating matrix similar to MovieLens dataset.

The ratings are on a numeric 5 star scale with 1,2 represents negative ratings, 3 represents satisfactory,4 represents good and 5 represents very good or excellent.

Table 1 Rating database

	Item1	Item2	Item3	Item4
Alice	3	2	3	2
Bob	2	1	2	1
Carol	5	4	5	4
Dave	4	2	4	2
Eve	4	1	4	1

Using Pearson correlation, it can be concluded that Alice, Bob, Carol, Dave and Eve are having similar tastes because for their linear offset from their mean, Pearson correlation groups patterns irrespective of their range of expression. For example Alice might belong to a group of users whose expression has a narrow range (2 to 4) while Eve might belong to a group of users with wider range of expression (1 to 5) to rank the items. We argue that Eve is more similar to Alice than others. Bob did not like any item, Carol liked all the items, Eve is more similar to Alice followed by Dave based on their rating pattern.

III Proposed Method

Correlation coefficient is $\frac{\text{Cov}(X,Y)}{\sigma_X \sigma_Y}$

Covariance (Cov(X, Y)) is dependent upon ranges of both X and Y, the dependency is removed by dividing with their respective standard deviations. i.e., values of X by its standard deviation (σ_X) and values of Y by its standard deviation (σ_Y)

Since dependency is removed, Person Correlation doesn't take magnitude into consideration, but still exhibit commutative property. We derive a measure which exhibit commutative property only when the patterns are linear with respect to distance.

Let two sets of data objects are represented by $X=\{x_1, x_2, \dots, x_n\}$ and $Y=\{y_1, y_2, \dots, y_n\}$

Then



$$\bar{x} = \frac{1}{n} \sum x_i$$

$$\bar{y} = \frac{1}{n} \sum y_i$$

offset of points from the mean for x and y is given as $\alpha_i = x_i - \bar{x}$ and $\beta_i = y_i - \bar{y}$

The unit dimension known as Standard deviation is measured as

$$\sigma_x = \sqrt{\frac{1}{n} \sum (\alpha_i)^2}$$

$$\sigma_y = \sqrt{\frac{1}{n} \sum (\beta_i)^2}$$

When there is a perfect correlation between X and Y

$$\frac{\beta_i}{\sigma_y} = \pm \frac{\alpha_i}{\sigma_x}$$

as $y_i = \lambda x_i + \xi_i$ for some constant ξ and λ

Therefore

$$\bar{y} = \frac{1}{n} \sum y_i$$

$$\bar{y} = \frac{1}{n} \left(\lambda \sum x_i + \sum \xi_i \right)$$

When both X and Y are same then $\lambda = 1$ and $\xi = 0$

i.e.,

$$\bar{y} = \frac{1}{n} \sum x_i \text{ and } \bar{x} = \frac{1}{n} \sum y_i \text{ then } \beta_i^x = y_i - \bar{x} \text{ and } \alpha_i^y = x_i - \bar{y}$$

According to Pearson, correlation coefficient between X and Y is given as

$$\text{corr}(X, Y) = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(x_i - \bar{x})^2} \sqrt{(y_i - \bar{y})^2}}$$

The above equation measures the linear relationship between X and Y. They are perfectly correlated when $r = 1$ i.e., when $\frac{\bar{x}}{\bar{y}} = c$

Since Pearson correlation does not take into consideration the magnitude the authors propose a new similarity estimate by substituting \bar{x} for \bar{y} or \bar{y} for \bar{x} in the equation

$$\text{sim}(X, Y) = \frac{\sum (x_i - \bar{x})(y_i - \bar{x})}{\sqrt{(x_i - \bar{x})^2} \sqrt{(y_i - \bar{x})^2}}$$

X is the user whose rating is being predicted and Y is the neighboring user. Two objects are perfectly correlated with respect to magnitude if and only if $\frac{\beta_i}{\sigma_y} = \pm \frac{\alpha_i^y}{\sigma_{xy}}$ and $\frac{\alpha_i}{\sigma_x} = \pm \frac{\beta_i^x}{\sigma_{yx}}$

That is they are similar with respect to linear relationship and also magnitude. Substituting \bar{x} for \bar{y} or \bar{y} for \bar{x} in the equation of correlation coefficient, does not have any effect on variance. But there will be change in standard deviation as it describes how the points are scattered with respect to the offset. The proposed similarity measure has a range from -1 to 1.

Prediction of Rating

The purpose of similarity estimate is used as given below to predict the ratings of a user 'u' for additional item 'i' based on the ratings given by the other users u' in his neighborhood, N.

$$\text{Pred}_{u,i} = b_{u,i} + \frac{\sum_{u' \in N} \text{sim}(u, u') * (r_{u',i} - b_{u',i})}{\sum_{u' \in N} |\text{sim}(u, u')|}$$

$$b_{u,i} = \mu + b_u + b_i$$

b_u and b_i are the baseline predictors for user and item respectively given by



$$b_u = \frac{1}{|I_u|} \sum_{i \in I_u} (r_{u,i} - \mu)$$

$$b_i = \frac{1}{|U_i|} \sum_{u \in U_i} (r_{u,i} - b_u - \mu)$$

Where μ is the global mean of ratings available in training set for all items and users. We choose baseline predictors [16] as it is used to adjust the effect of giving over ratings by a user or receiving higher ratings by a item and also provide ratings for new users.

IV Experimentation and Results

Experiments were conducted with java and open source R.

a. Data sets

MovieLens-100K

MovieLens datasets [17] were maintained by University of Minnesota as part of GroupLens Research Project. The rating record is maintained as a triplet <UserID, MovieID, rating>

The dataset u.data contains

Users	943
Movies	1682
ratings	1,00,000
Density	0.063047
Min No. of Items Rated by user	20

The 100k dataset ships with a 5 random 80%-20% splits of train and test sets (u1.base:u5.base and u1.test:u5.test) respectively that can be used for 5 fold cross validation. All the 5 test sets are disjoint.

The data set also comes with ua.base/ua.test and ub.base/ub.test with test set having exactly 10 ratings for each user. both ua.test and ub.test are disjoint.

Iris and Wine datasets

We also used Iris [9] and Wine [19] datasets to test the effectiveness of our measure in grouping similar objects.

b. Evaluation Metrics

b.i Prediction Evaluation Metrics

Success of a recommender system is gauged by the quality of prediction.

We used MAE (mean absolute error) to report prediction accuracy on the test set.

$$MAE = \frac{\sum_{u \in U} \sum_{i \in \text{testset}_u} |\text{pred}(u, i) - r_{u,i}|}{\sum_{u \in U} |\text{testset}_u|}$$

MAE is a statistical accuracy metric which compares the prediction with the actual rating in the test set for a particular user.

b.ii Recommendation evaluation metrics

To evaluate the top-N recommendation given to a user we used **Normalized Discounted Cumulative gain (NDCG)**

NDCG is used to find the effectiveness of top k items that are retrieved compared to actual list

NDCG [10] measures the performance of the Recommender system by discounting the positions logarithmically, assuming each user u has a gain g_{ui} from being recommended an item i , the Discounted Cumulative Gain (DCG) for a list of items until position J is defined as

$$DCG_J = \frac{1}{N} \sum_{u=1}^N \sum_{j=1}^J \frac{g_{uij}}{\log_2(j+1)}$$



$$g_{uij} = \begin{cases} 1 & \text{if the predicted item is in the original top p list} \\ 0 & \text{otherwise} \end{cases}$$

Numbers of items to consider for prediction in the test set vary from user to user. To achieve consistency across all users the cumulative gain of each user should be normalized with maximum possible DCG till position J or a threshold.

$$NDCG_j = \frac{DCG_j}{IDCG_j}$$

IDCG_j Ideal Discounted Cumulative Gain represents the maximum possible DCG till position J. NDCG value lies between 0 and 1. Higher the NDCG the closer the predicted ranked list is closer to the original ranked list.

b.iii Cluster Purity

Cluster Purity – evaluation of Cluster Quality

C = {C₁, C₂, C_i, ..., C_k} represent Clusters.

L = {L₁, L₂, ..., L_j} represent the class labels

n_jⁱ = L_i ∩ C_j Represent the number of objects of ith cluster belongs to class j

Compute the Cluster purity by finding the class label that belongs to majority objects in each cluster and to find the total number of labeled objects of such class in each cluster [8] given as

$$purity(L, C) = \frac{1}{N} \sum_k \max_j |L_j \cap C_k|$$

Cluster purity has a range of [0; 1], 1 indicates that all objects belong to one class.

c. Experiments

c.i Prediction Experiment

First we worked on 5 fold cross validation of train/test set from u1...u5

We calculated bias of each user and for each item in the train set, this process is repeated for all the five folds separately and calculated prediction for each user –item pair in the test set of each fold.

The number of common items for similarity calculation is kept as minimum two but the prediction for active user is calculated based on top-k nearest neighbors. We repeated the experiment taking k-value from 5-100 increasing by 5. We calculated MAE of each fold and took the average of all five folds.

The same process is continued with UA and UB datasets.

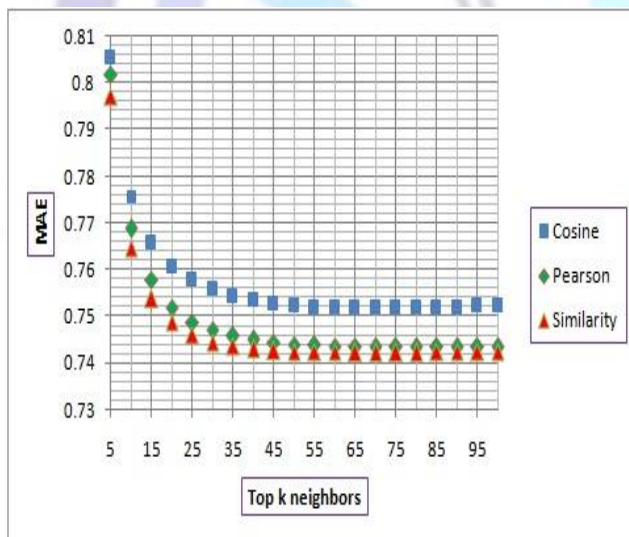


Figure 1: MAE comparisons of Cosine, Pearson and Similarity on 5 folds cross validation.

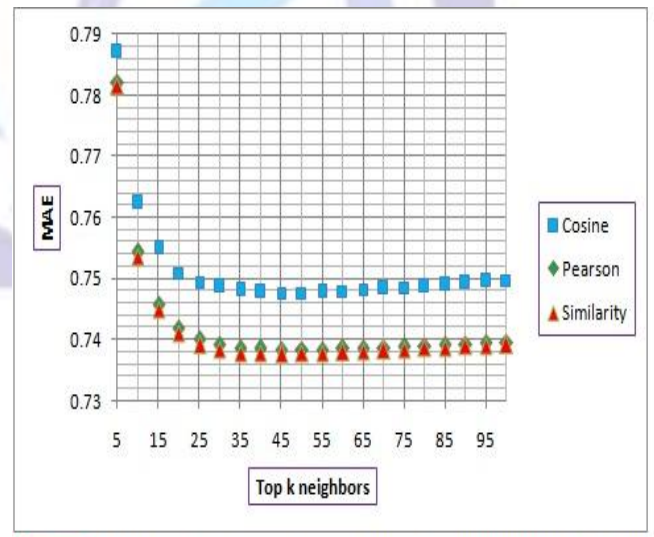


Figure 2: MAE comparisons of Cosine, Pearson and Similarity on 5 folds cross validation with damp factor.

We conducted second experiment on UA dataset with shrunk in similarity coefficient by $sim(u, u') = sim(u, u') \times \text{number of common_items} / \text{dampfactor}$ where damp factor is kept as 25 if the number of common items less than 10.

Choosing damp factor more than 25 or increasing the number of common items does not have much effect.



The same process is continued with UB dataset.

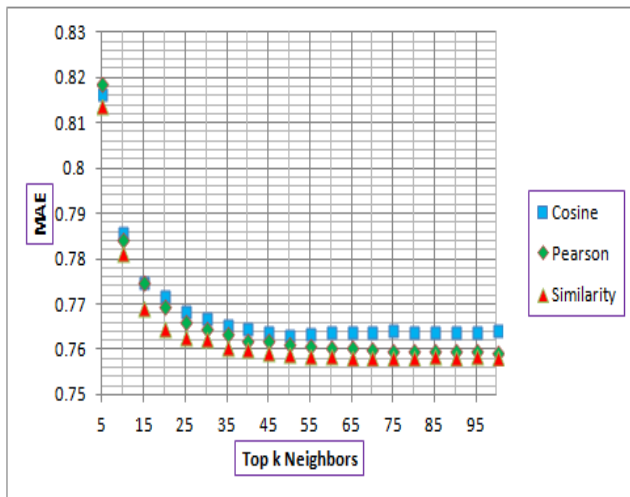


Figure 3: MAE comparison of Cosine, Pearson and Similarity on ua dataset.

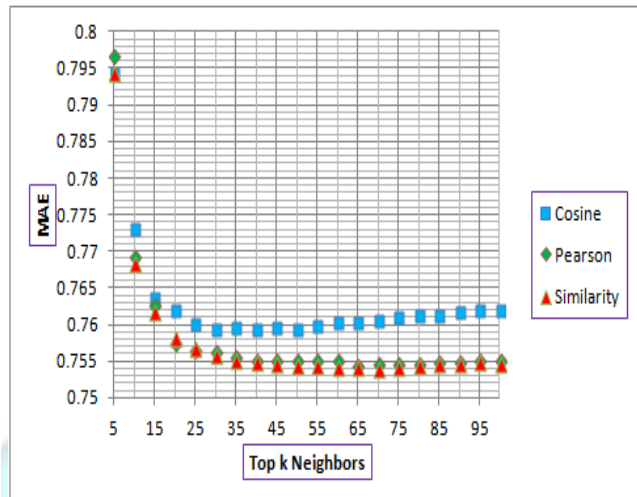


Figure 4: MAE comparison of Cosine, Pearson and Similarity on ua dataset with damp factor.

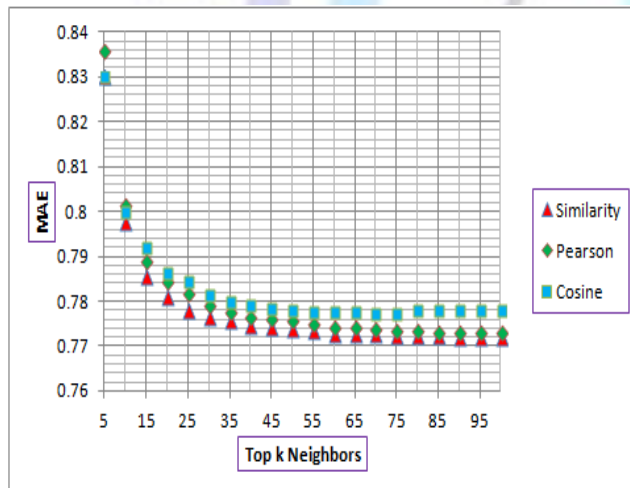


Figure 5: MAE comparison of Cosine, Pearson and Similarity on ub dataset.

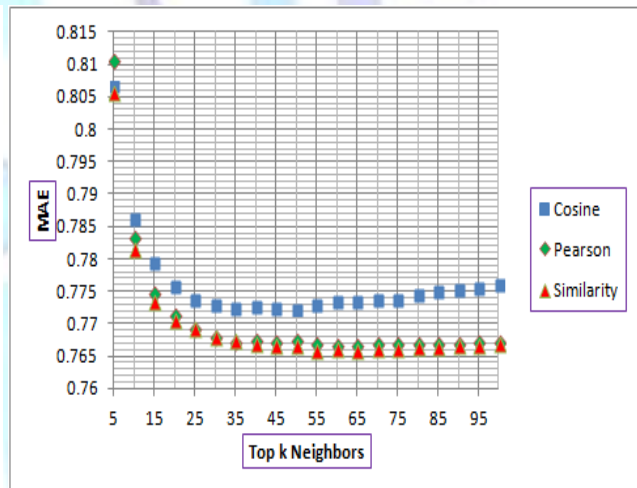


Figure 6: MAE comparison of Cosine, Pearson and Similarity on ub dataset with damp factor.

Discussion on Prediction results

Fig: 1-6 shows that our method consistently performed better than Cosine similarity and Pearson correlation. We considered common items list size as minimum 2 for all validations unless otherwise stated.

As Cosine similarity always gives 1 if number of common items 1, Pearson correlation and our proposed method requires minimum 2 common items.

Cosine similarity performed better than Pearson correlation when top-5 neighbors is considered on ua dataset (Fig:3 & Fig:4)but Pearson correlation performed better when neighborhood size is increased. Our method consistently performed better compared to other two measures.

Cosine similarity performed better than Pearson correlation when top-5 and top-10 neighbors is considered for prediction on ub dataset (Fig:5),but Pearson correlation performed better when neighborhood size is increased. Our method consistently performed better compared to other two measures.

It is clear from Fig: 2, Fig: 4 and Fig: 6 that when a damping factor is applied to shrink the similarity coefficient if the number of common items is less than a threshold (we considered 10) MAE decreased considerably to stabilize the prediction.

**Table 2** NDCG of UA dataset with and without shrink

		Cosine	Pearson	Similarity
Top -5	With shrink	0.642052	0.645474	0.647143
	Without shrink	0.641147	0.642285	0.64279
Top - 7	With shrink	0.792881	0.797189	0.798582
	Without shrink	0.796654	0.796471	0.797906

c.ii Top-k recommendation experiment

In this we took ua and ub datasets for evaluation as they contain 10 ratings for each user in the test set..

We sorted the actual ratings given by a user in the test set and also the predicted ratings separately. As each user is having ten ratings in the test set we considered only seven ratings in the actual ratings list and compared with the predicted list if the items in the actual list appear in the same

order and calculated discounted cumulative gain of their positions for top-5 list and top-7 list.

Table 3 NDCG of UB dataset with and without shrink

		Cosine	Pearson	Similarity
Top - 5	With shrink	0.654585	0.666924	0.669414
	Without shrink	0.655299	0.662777	0.665314
Top - 7	With shrink	0.80193	0.806154	0.806548
	Without shrink	0.800010	0.802529	0.803975

Discussion on Top-k Recommendations

Table: 2 and Table: 3 show that our method out performed both Cosine similarity and Pearson correlation in providing top-k list of most liked items by the user. NDCG shows that our proposed is efficient in retrieving the most liked items from the available list.

c.iii Cluster purity experiment**Table 4** cluster purity comparison of Cosine similarity, Pearson correlation and proposed Similarity measure on wine dataset

K-Means –Wine Dataset – 3 clusters	
Distance/Similarity Measure	Cluster Purity
Cosine similarity	0.8932584
Pearson correlation	0.8876404
Similarity	0.9213483



Table 5 cluster purity comparison of Cosine similarity, Pearson correlation and proposed Similarity measure on Iris dataset

K-Means –Iris Dataset – 3 clusters	
Distance/Similarity Measure	Cluster Purity
Cosine similarity	0.7266667
Pearson correlation	0.74
Similarity	0.86

Discussion on Cluster purity

Table 4 and Table 5 show that our proposed measure forms Quality clusters compared to Cosine similarity and Pearson correlation on normalized iris and wine datasets

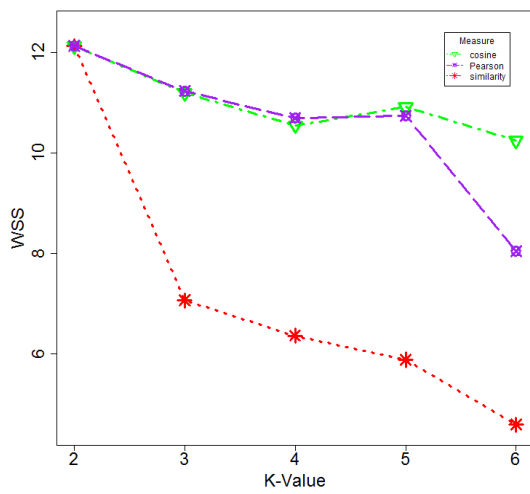


Figure:7 IntraCluster SSE - Iris Dataset

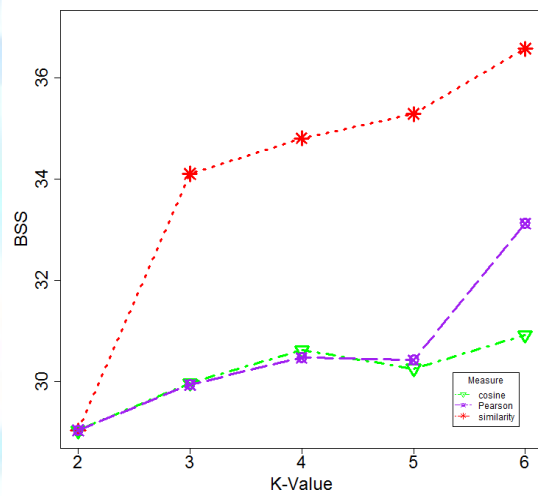


Figure:8 InterCluster SSE - Iris Dataset

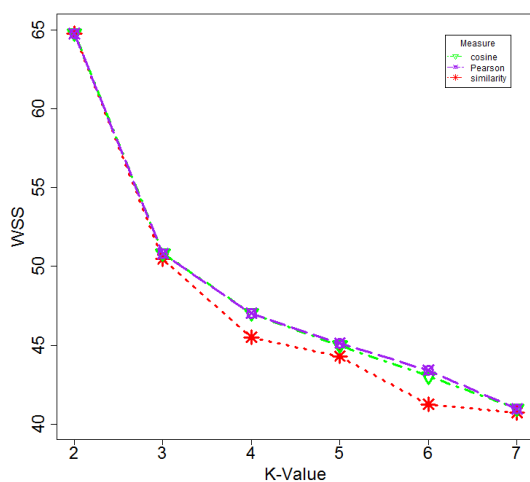


Figure:9 IntraCluster SSE - Wine Dataset

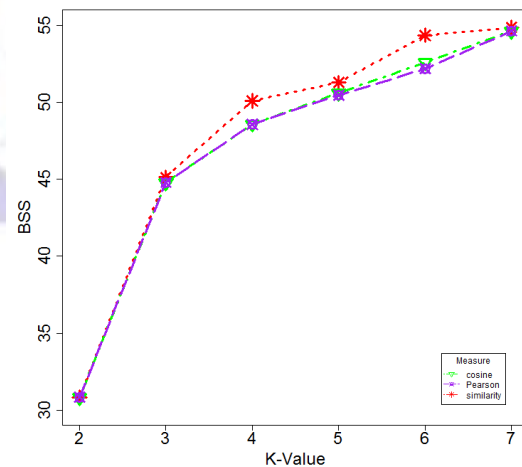


Figure:10 InterCluster SSE - Wine Dataset

Fig: 7 and Fig: 9 show that intra-class similarity is high with our proposed measure.

Fig: 8 and Fig: 10 show that inter-class similarity is low with our proposed method.



V Conclusions

Existing Distance/similarity Metrics or measures are not enough to deal with all kinds of data analysis. In Recommender Systems, finding similarity among users or Items to improve the prediction Quality is still a open research area. Our proposed similarity measure is consistent and has done exceedingly well compared to Cosine similarity and marginally well with Pearson correlation. We have also shown that our proposed measure is efficient in clustering similar objects.

Acknowledgement

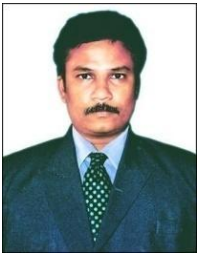
We Thank Movielens,Iris and Wine dataset providers for making them available on the web.

References

- [1] Adomavicius, G. and A. Tuzhilin (2005). "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): 734-749.
- [2] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl.(2001)."Item-based collaborative filtering recommendation algorithms,"Proceedings of the 10th International Conference on World Wide Web, page 285-295. ACM
- [3] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl.(2002)."Incremental SVD-based algorithms for highly scaleable recommender systems," in proceedings of the 5th International Conference on Computer and Information Technology.
- [4] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl.(2002)."Recommender Systems for Large Scale E-Commerce:Scalable Neighborhood Formation Using Clustering," in proceedings of the 5th International Conference on Computer and Information Technology.
- [5] Cha, S. H. (2007). "Comprehensive survey on distance/similarity measures between probability density functions." City.
- [6] Christian Desrosiers and George Karypis.(2011). Advances in Collaborative Filtering,Recommender Systems Handbook, Springer.
- [7] Dietmar Jannach , Markus Zanker , Alexander Felfernig , Gerhard Friedrich .(2010).Recommender Systems: An Introduction , Cambridge University Press.
- [8] Hinrich Schütze Christopher D. Manning, Prabhakar Raghavan.(2008). Introduction to Information Retrieval.,Cambridge University Press.
- [9] Iris dataset: <http://archive.ics.uci.edu/ml/datasets/Iris>
- [10] Järvelin, K. and J. Kekäläinen. (2002). "Cumulated gain-based evaluation of IR techniques." ACM Transactions on Information Systems 20(4): 422-446.
- [11] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl.(1999).An Algorithmic Framework for performing collaborative filtering in ACM SIGIR,pp. 230-237,ACM.
- [12] J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl.(2004).Evaluating collaborative filtering recommender systems.,ACM Transactions on Information Systems 22(1):5-53
- [13] John S. Breese, David Heckerman, and Carl Kadie.(1998).Empirical Analysis of Predictive Algorithms for Collaborative Filtering.,Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, page 43-52.
- [14] J.wang,A. De Vries,and M. Reinders.(2006).Unifying user-based and item-based collaborative filtering by similarity fusion in ACM SIGIR,pp.501-508,ACM.
- [15]K. Goldberg, T. Roeder, D. Gupta, and C. Perkins.(2001). Eigentaste: A Constant Time Collaborative Filtering Algorithm.,Information Retrieval 4(2):133-151
- [16] Koren, Y. (2008). "Factorization meets the neighborhood: A multifaceted collaborative filtering model." Proc ACM SIGKDD: 426-434.
- [17]Movielens dataset: <http://grouplens.org/datasets/movielens>
- [18] Upendra Shardanand, and Patti Maes.(1995).Social Information Filtering: Algorithms for Automating ``Word of Mouth"., Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems, 1, page 210-217.
- [19]Wine dataset: <http://archive.ics.uci.edu/ml/datasets/Wine>
- [20] Xavier Amatriain,Alejandro Jaimes,Nuria Oliver,and Josep M. Pujol. (2011). Advances in Collaborative Filtering.,Recommender Systems Handbook, Springer.
- [21] Y. Koren, R. Bell, and C. Volinsky.(2009).Matrix Factorization Techniques for Recommender Systems., Computer 42(8):30-37
- [22] Yehuda Koren, and Robert M. Bell.(2011).Advances in Collaborative Filtering.,Recommender Systems Handbook, Springer.



AUTHORS



T.Srikanth received his M.Tech. Degree in Computer Science and Technology from Andhra University. He is presently working as an Associate Professor in the department of Computer Science and Engineering, Institute of Technology, GITAM University, Visakhapatnam, Andhra Pradesh, India. He is pursuing his Ph.D in J.N.T.U, kakinada. His areas of interest include Machine learning, Artificial intelligence, Data Mining, Recommender Systems, Softcomputing.



M.Shashi received her B.E. Degree in Electrical and Electronics and M.E. Degree in Computer Engineering with distinction from Andhra University. She received Ph.D in 1994 from Andhra University and got the best Ph.D thesis award. She is working as a Professor of Computer Science and Systems Engineering at Andhra University, Visakhapatnam, Andhra Pradesh, India. She received AICTE career award as young teacher in 1996. She is a coauthor of the Indian Edition of text book on "Data Structures and Program Design in C" from Pearson Education Ltd. She published technical papers in National and International Journals. Her research interests include Data Mining, Artificial intelligence, Pattern Recognition and Machine Learning. She is a life member of ISTE, CSI and a fellow member of Institute of Engineers (India).

