A Novel Similarity Measure to Identify Effective Similar Users in Recommender Systems

Rajeswari Nakka, G.V.S.N.R.V.Prasad, R.Kiran Kumar

Abstract— In recent years there is a drastic increase in information over the internet. Users get confused to find out best product on the internet of one's interest. Here the recommender system helps to filter the information and gives relevant recommendations to users so that the user community can find the item(s) of their interest from huge collection of available data. But filtering information from the users reviews given for various items seems to be a challenging task for recommending the user interested things. In general similarities between the users are considered for recommendations in collaborative filtering techniques. This paper describes a new collaborative filtering technique called Adaptive Similarity Measure Model [ASMM] to identify similarity between users for the selection of unseen items. Out of all the available items most similarities would be sorted out by ASMM for recommendation which varies from user to user.

Key Words: Collaborative Filtering, CB-Filtering, ASMM and Recommendation Systems.

I. INTRODUCTION

The Recommender system is a type of data filtering technique whose challenge is to take user's priorities and make recommendations based on those priorities. There is a wide range of recommendation system applications. The popularity of referral systems has been steadily increasing and has recently been implemented on almost all online platforms used. It is applied in several neighboring areas, such as information retrieval or human-computer interaction (HCI) [2] [3]. It gathers a huge amount of information on the preferences of users of several items such as online shopping products, movies, taxis, television shows, tourism, restaurants, etc.

Recommender system captures users feedback on movies viewed, places visited, and products purchased. Among various recommender systems available movie recommender systems found to be one of the frequently used systems by many users [5, 7].

Netflix is an American video on demand Service Company. Currently it is having 148 million subscribers all over the world. Although the amount of information available has increased, a new problem has emerged as people have difficulty in selecting what they really want to see. This is where the recommendation system comes in [8] [10]. Everyone has different likes and dislikes. In addition to this, even the taste of a single client may vary according to the context, such as mood, season or type of activity they are interested. For example, the type of film a person prefer to watch will change with the people with whom he is going to watch the movie, like when he wants to watch the movie with his family he may have a different preference or when he wants to watch the movie with his friends he may prefer some other movie.

There are 3 methods which are widely used in recommender systems [12]. One is content-based filtering (CBF), which tries to configure user preferences using, user or item profile. The second one is collaborative filtering (CF), which tries to group the clients based on their similarity and later done the recommendations. The other alternative is Hybrid based filtering (HBF), which combines both CBF and CF. Fig 1 gives information about various types of recommender systems.

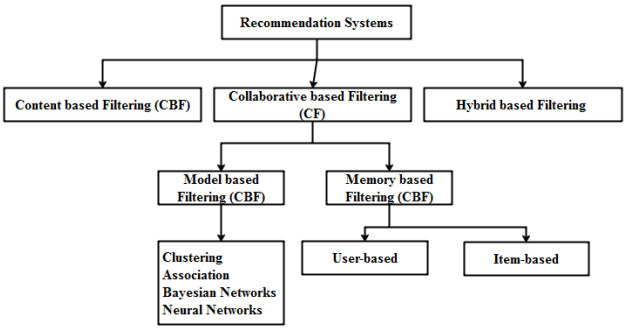

Fig 1. Types of Recommender Systems

1.1 Content-Based Filtering

It is also called cognitive filtering technique that recommends elements based on a comparison between the content of the elements and user profile. The content of each element is represented as a set of descriptors or terms, usually the words that appear in a document. The user profile is represented by the same conditions and the recommendation has given based on analyzing the content of the elements viewed by the user.The main advantages of the CB-Filtering is that, when information or data of other users is available completely, then the recommender system can advise new products that has not been evaluated at present, however, the recommendation system does not recommend the products which are outside the type of products that the user has qualified.

1.2 CF (Collaborative Filtering)

Revised Manuscript Received on July 18, 2019.

Rajeswari Nakka, Research Scholar, Dept. of CSE, JNTUK, Kakinada, India, (rajeswari.gec@gmail.com)

Dr G.V.S.N.R.V.Prasad, Professor of CSE, Gudlavalleru Engineering College, Gudlavalleru, India, (gutta.prasad1@gmail.com)

R.Kiran Kumar, Dept. of Computer Science, Krishna University, Machilipatnam, India, (kirankreddi@gmail.com)

CF is a methodology of creating automatic filtering regarding the interests of a user by assembling preferences from several users (collaborating). The underlying assumption of the CF approach is that if an individual A has identical opinion as an individual B on a difficulty, A is a lot of seemingly to have B's opinion on a special issue than that of a indiscriminately chosen person.

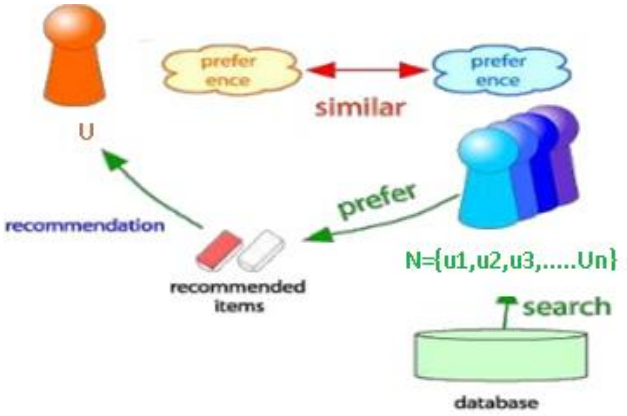

Fig 2.Collaborative Filtering

It is mainly divided into two categories as follows:

1.2.1. Model based Filtering

It is very important technique in recommendation system, developed by using data mining techniques. CF-Based technique can gain high accuracy and solve the critical problem of sparse matrix.

1.2.2. Memory based Filtering

In Memory based filtering, given user u_i, it needs to find a set $N=\{u_1,u_2,u_3,....,u_n\}$ of other users whose ratings are "similar" to u_i'sratings and estimate u_i'sratings on products based on users from N. This is shown in Fig 2.

1.3 Hybrid Based Filtering

CF and CB Recommender techniques have complementary strengths and weaknesses. For instance, CF methods suffer from new-item problems, i.e., they cannot recommend items that have no ratings. A hybrid system combines both the CBF and CF techniques. It is shown in Fig 3.

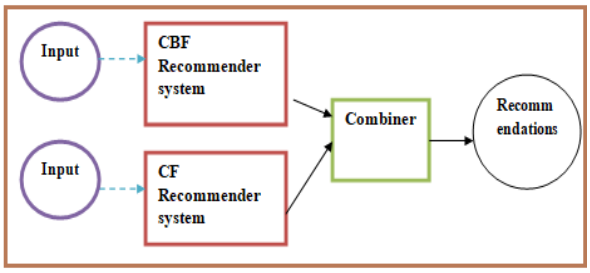

Fig 3.Hybrid Based Filtering

II. BACKGROUND WORK

2.1. Recommender Problem

The recommender problem defines a utility function that predicts the rating of items for users from the ratings of existing ratings given by other users. The utility function is defined on the rating matrix $U\times I$, where U represents set of users and I represents set of items. In real time this matrix is sparse, because the users rate a small subset of available items. The definition of recommender problem is given as follows: Let U be set of users, where $U=\{u_1,u_2,u_3,.....,u_n\}$, and I a set of items where $I=\{\{i_1,i_2,i_3,.....,i_m\}$ then the Recommender System tries to find a utility function UT that stands for the usefulness of $i_1^1...i_n^1 \in I$ touser $ua \in U$,such that UT finds $\{r_1,r_2,r_3,...r_n\}$ for items $i_1^1...i_n^1 \in I$ where $\{r_1,r_2,r_3,...r_n\}$ are the ratings for items $i_1^1...i_n^1 \in I$ which is derived from *user-item* rating matrix.

2.2. Distance/Similarity Measures

In recommendation systems similarities between two users can be calculated by using an approximation of conventional knowledge. In recommendation systems, the following similarity measures are considered.

1. *Cosine Similarity*
2. *Pearson correlation similarity*
3. *Jaccard Measure*

2.2.1. Cosine Similarity:

It mainly finds the similarities among the two user rating vectors and is represented as follows in Eq(1).

$$Cos_{sim}(u,u') = \frac{\sum_{i\in I_u\cap I_{u'}}(r_{u,i})(r_{u',i})}{\sqrt{\sum_{i\in I_u\cap I_{u'}}(r_{u,i})^2}\sqrt{\sum_{i\in I_u\cap I_{u'}}(r_{u',i})^2}} \quad (1)$$

Where I_u & $I_{u'}$ denote the elements qualified by user (u & u'). The similarity of the cosine stabilizes the data with reference to the origin.

2.2.2. PC (Pearson Correlation)

The PC finds correlation between two users u & u' with common movie ratings is given by Eq (2).

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

Where I_u & $I_{u'}$ denote the elements qualified by user (u & u'). The PC can also be seen as the CS normalized by the movement of the equivalent items to find the degree of linearity. The PC has a range of - 1 to 1. It is widely used in RS to identify users that show a linear relationship, that is, similar tastes.

2.2.3. Jaccard Measure

Here similarity between users is calculated by number of common selections. This can be done by using Eq(3).

$$Jacc_{sim}(u,u') = \frac{|I_u| \cap |I_{u'}|}{|I_u| \cup |I_{u'}|} \quad (3)$$

The above measures suffer from the drawbacks like, low similarity regardless of similar ratings of two users. High similarity regardless of the difference between the ratings of the two users. Ignoring the proportion of common ratings will result in low accuracy.

III. IMPROVED COLLABORATIVE FILTERING TECHNIQUE

This section mainly describes CF Recommendation system based on a modified similarity model, which helps to overcome the drawbacks of traditional similarity measures[4,9]. First, the framework of the proposed model is introduced. Next, the algorithm of the modified similarity model is provided. The user's movie rating matrix is constructed using the Movie Lens data set. After the formulation of the matrix from the data set, the proposed similarity model is applied between the target user and the rest of the users of the matrix.

Algorithm: Adaptive Similarity Measure Model.

Input: user-movie rating matrix.

Output: user-movie rating matrix with predicted ratings.

Begin $\mathbf{U} = \{\mathbf{u_1}, \mathbf{u_2}, \mathbf{u_3}, \ldots\ldots, \mathbf{u_n}\}$ the number of all users in the dataset.

$\mathbf{M}$ Number of movies in the dataset.

$\mathbf{U_m}$ the number of users rated movie m

Step1: For all $\mathbf{M}$ in user-movie rating matrix do

Compute the inverse user frequency (IUF) for each movie $\mathrm{m} \in \mathbf{M}$ using the Eq(4)

$$IUF(m) = \log \frac{U}{U_m} \quad (4)$$

Step2: For all $\mathbf{U}$ in user-movie rating matrix do

Compute the similarity between each pair of users using the Constraint Pearson correlation coefficient with its modification using IUF(m) , the modified constraint Pearson correlation coefficient by using Eq(5).

$$Sim_{ASM}(u,u') = \frac{\sum_{movies\ m \in M} IUF(m)^2 (r_{u,m} - \bar{r}_u)(r_{u',m} - \bar{r}_{u'})}{\sqrt{\sum_{movies\ m \in M} IUF(m)^2 (r_{u,m} - \bar{r}_u)(r_{u',m} - \bar{r}_{u'})}} \quad (5)$$

Step3:For all U in user-movie rating matrix do

Compute the similarity between each pair of users using the Jaccard similarity distance to consider the proportion of common movies using the formula given in Eq(6).

$$Sim_{jaccard}(u,u') = 1 - \frac{sum(m_u \text{ and } m_{u'})}{sum(m_u | m_{u'})} \quad (6)$$

Step4: For all users **U**, in user-movie rating matrix do multiply similarities obtained from Eq(5) and Eq(6) for user $\mathbf{u}$ and user $\mathbf{u'}$ to get the similarity for the proposed similarity model, this is given in Eq(7):

$$Sim_{modified}(u,u') = Sim_{ASM}(u,u') \times Sim_{jaccard}(u,u') \quad (7)$$

Step5: The proposed similarity estimate is used as given below in Eq(8) to predict the ratings of a user '**u**' for new item '**i**' based on the ratings given by the other users in his neighborhood **N**

$$Pred_{u,i} = \delta_{u,i} + \frac{\sum_{u' \in N} Sim(u,u')(r_{u',i} - \delta_{u',i})}{\sum_{u' \in N} |Sim(u,u')|} \quad (8)$$

$$\delta_{u,i} = \mu + \delta_u + \delta_i \quad (9)$$

$\boldsymbol{\delta_u}$ and $\boldsymbol{\delta_i}$ are the baseline predictors for user and item respectively given by Eq(10) and Eq(11).

$$\delta_u = \frac{1}{|I_u|} \sum_{i \in I_u} (r_{u,i} - \mu) \quad (10)$$

$$\delta_i = \frac{1}{|U_i|} \sum_{u \in U_i} (r_{u,i} - b_u - \mu) \quad (11)$$

Where $\boldsymbol{\mu}$ is the global grade point average available in the training set for all articles and users. These baseline predictors are used to adjust the effect of granting higher ratings by a user or receiving higher ratings for an item

End

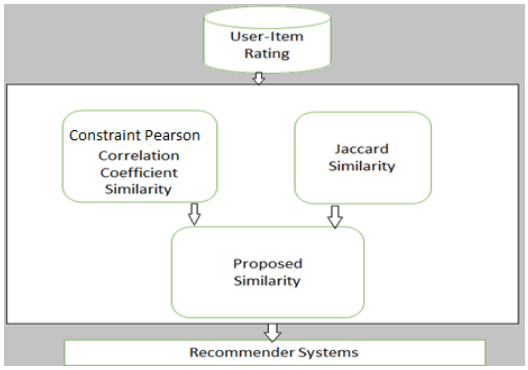

Fig.4. Architecture of Adaptive Similarity Measure Model

IV. ASSESSMENT BASED ON THE RATINGS

The techniques which evaluate the accuracy of recommender systems will do the comparison of the forecast ratings with real values. Particularly, these techniques find the normal error value based on three indicators RMSE, MAE and MSE which are calculated by using Eq(12), Eq(13) and Eq(14). A model is evaluated well when these indicators shows a small value.

4.1.1. RMSE (Root Mean Square Error):

$$RMSE = \sqrt{\frac{\sum_{u,u' \in M}((R_{u,u'}) - (\bar{R}_{u,u'}))^2}{|M|}} \quad (12)$$

4.1.2. MSE (Mean Squared Error):

This is the average of the squared difference between the actual and estimated ratings. This is the square of RMSE, it contains the same information.

$$MSE = \frac{\sum_{u,u' \in M}(R_{u,u'} - \bar{R}_{u,u'})^2}{|M|} \quad (13)$$

4.1.3. Mean Absolute Error (MAE)

This is the average absolute difference between actual and expected ratings.

$$MAE = \frac{1}{|M|} \sum_{u,u' \in M} |R_{u,u'} - \bar{R}_{u,u'}| \quad (14)$$

Where M = set of all user ratings for items, $R_{u,u'}$ = Real rating value of user u for item u'.

$\bar{R}_{u,u'}$ = is predicted rating value of user u for item u'

4.1.4. Evaluation based on the recommendations

One method evaluates the accuracy of the model by comparing the model recommendations to user choice. This approach uses the confusion matrix to calculate the value of three indicators: precision, recall and F-measure. The model is priced right if three indices are of high value[6,11].

V. RESULTS

5.1. Data description

All experiments are conducted on Movie Lens Data. This data set is compiled from the evaluation results of 943 users for 1682 films (the ratings ranges from 1 to 5) from Movie Lens site (movielens.umn.edu) for 7 months (from 19.09.1997 to 22.04.1998). As this data set have 943 users and 1682 items, in general the total rating matrix contains 943 X 1682 entries i.e. totally 1,586,126 values. But the rating matrix has 1, 00,000 users rating values, because each user can watch only their favourite movies. In general, experiments were implemented with users and movies from the Movie Lens Data set.Fig5 give the exploration of the Movie Lens Data set.

Fig 5(a)

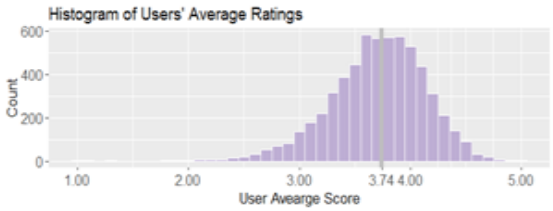

Fig 5(b)

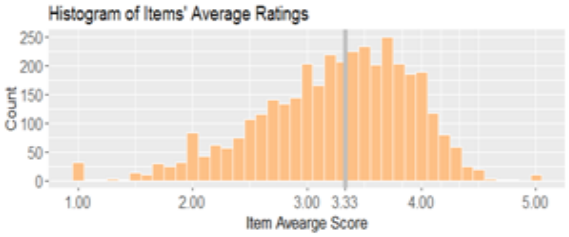

Fig 5(c)

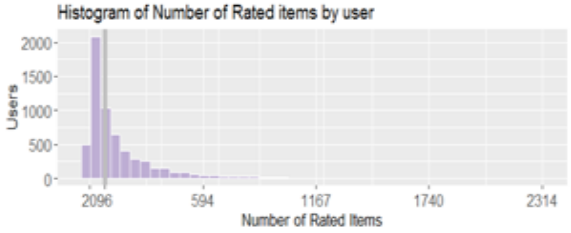

Fig 5(d)

Fig 5(e)

Fig 5. Exploration of 1M Movie Lens data set

Fig 5(a) shows that, from the available ratings majority of the items enjoys ratings 3 to 5.From Fig 5(b) & 5(c), it is evident that majority of the average scores for users and

items falls in the range of 3 to 4.From Fig 5(d), 5(e) it is clear that much of the user ratings are unavailable. Which results in to a Sparse User-Item rating matrix.

Fig 6 shows the number of movies co-rated among 6 users. These numbers are necessary during the similarity calculation.

	u_6	u_5	u_4	u_3	u_2	u_1
u_6	201	39	0	8	26	87
u_5		165	1	1	3	73
u_4			14	6	3	4
u_3				44	7	87
u_2					52	15
u_1						262

Fig6. The Number of Correlated (Co-Rated) Movies between 6 Users

	u_6	u_5	u_4	u_3	u_2	u_1
u_6	0.0000	0.2327	0.0000	0.0730	0.2236	0.4122
u_5		0.0000	0.0132	0.0235	0.0498	0.3646
u_4			0.0000	0.2357	0.1277	0.0523
u_3				0.0000	0.1259	0.0508
u_2					0.0000	0.1478
u_1						0.0000

Fig7. Cosine Similarity Matrix

Fig 7 presents the similarities values among users after applying the cosine similarity formula. It was demonstrated from the results, this measure cannot be used to generate the neighborhood because the values are far away from the average rating of each user. The use of this measure will lead to a poor neighborhood formation. As shown in Fig 7, these similarities in values cannot distinguish positive or negative impact among users. So this may not be an effective way of recognizing similar users.

	u_6	u_5	u_4	u_3	u_2	u_1
u_6	1.0000	0.9355	0.0000	0.8808	0.9565	0.9527
u_5		1.0000	1.0000	1.0000	0.9829	0.9285
u_4			1.0000	0.9484	0.9918	0.9318
u_3				1.0000	0.9522	0.8555
u_2					1.0000	0.9545
u_1						1.0000

Fig 8. Pearson Similarity Matrix

Fig 8 shows the similar values obtained with Pearson Correlation, which gives no indication among users who rated few items. The similarities calculated by using PC differ slightly although the number of co-rated movies differs diversely and largely.

	u_6	u_5	u_4	u_3	u_2	u_1
u_6	0.0000	0.8807	1.0000	0.9664	0.8857	0.7687
u_5		0.0000	0.9946	0.9953	0.9862	0.7936
u_4			0.0000	0.8848	0.9526	0.9856
u_3				0.0000	0.9092	0.9768
u_2					0.0000	0.9497
u_1						0.0000

Fig9. Jaccard Similarity Matrix

Fig 9 shows the similar values with Jaccard similarity formula, from this result it was shown that the users who experienced more movies in common gets more similarity. The Jaccard measure gives no influence alone.

	u_6	u_5	u_4	u_3	u_2	u_1
u_6	1.000	0.423	0.000	–0.434	0.469	0.609
u_5		1.000	NaN	1.000	0.868	0.496
u_4			1.000	–0.198	0.826	0.306
u_3				1.000	–0.665	–0.108
u_2					1.000	0.637
u_1						1.000

Fig 10. Constraint Pearson Similarity Model Matrix (CPSMM)

Fig 10, shows that theCPSMM distinguishes between the positive and negative impacts of users according to their ratings. This CPSMMhas a good impact during the generation of neighborhoods, because of considering the positive similar values and discarding the negative similar values. Due to this reason, the use of this measure in the proposed model can be contribute to well for finding similar users.

Table-1, shows the IUF calculations for 6 movies from the Movie Lens dataset.

Number of users to Movie	Movie ID1 392	Movie ID2 121	Movie ID3 85	Movie ID4 198	Movie ID5 79	Movie ID6 23
Weight (IUF)	0.877	2.0543	2.4054	1.5618	2.4776	3.7236

	u_6	u_5	u_4	u_3	u_2	u_1
u_6	0.000	0.218	0.000	–0.038	0.409	0.454
u_5		0.000	NaN	0.997	0.912	0.396
u_4			0.000	0.348	0.926	0.696
u_3				0.000	–0.465	–0.768
u_2					0.000	0.767
u_1						0.000

Fig 11. Proposed Similarity Model Matrix

Fig 11 shows the result of similarity values after applying Adaptive Similarity Measure Model described in section 3. The suggested model is better

than the three measures to identify similar users (neighbors) to the target user during the similarity calculation. In this model the Jaccard measure considers the positive impact and the number of films viewed by each userA and considers the influence of the film in the similarity calculation by using the IUF transformation by focusing on movies that aspects little.

MODEL	RMSE	MSE	MAE
IBCF_cos	1.3767018	1.8961917	1.0443078
IBCF_pea	1.2114741	1.4686051	0.9035095
IBCF_proposed	1.3767018	1.8961917	1.0443078

Table 2-Performance parameters of different models for IBCF

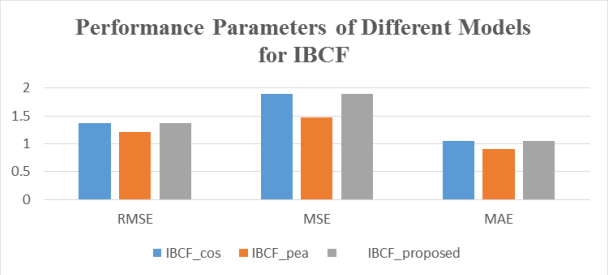

Fig12.Performance Parameters of Different Models for IBCF

Table 2 and Fig 12 shows a comparison between the traditional similarity measures and the proposed Adaptive Similarity Measure Model with Item Based Collaborative Filtering (IBCF). It is evident from this table that the proposed model gives the same effect as Cosine similarity.

MODEL	RMSE	MSE	MAE
UBCF_cos	1.0033265	1.0070652	0.7969657
UBCF_pea	0.9939421	0.988355	0.7873376
UBCF_proposed	0.9938316	0.9881305	0.7875188

Table 3-Performance parameters of different models for UBCF

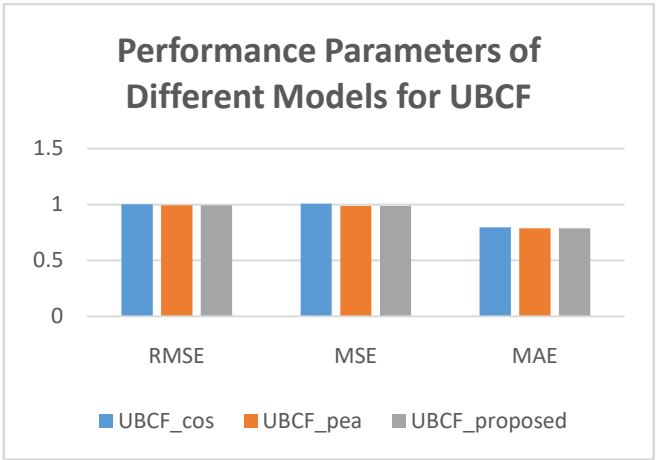

Fig13.Performance Parameters of Different Models for UBCF

Table 3 and Fig 13gives a comparison between the traditional similarity measures and the proposed Adaptive Similarity Measure Model with User Based Collaborative Filtering (UBCF). It is evidence from this table that the proposed model outperforms than the existing techniques in case of user based collaborative filtering techniques.

5.2. Identifying the most suitable model

This paper compares the models by building their ROC curves and precision / recall as shown in Fig 14 andFig15. Based on the curve of the ROC the largest AUC is at k =10. Another good choice of k is 5, but it can never have a high TPR[1]. This means that even if the k value is set high, the algorithm will not be able to recommend a large percentage of items that the user liked. The IBCF with k = 5 only recommend some similar items. Therefore, it cannot be used to recommend numerous articles. Based on the precision / recall plot, k should be set at 10 to achieve the highest return.

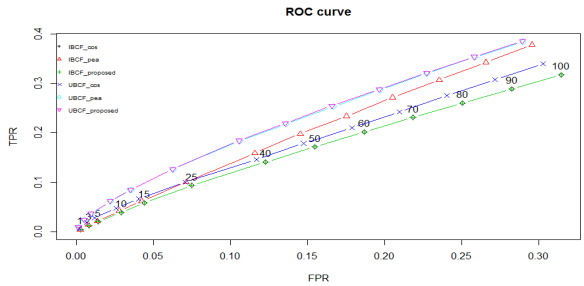

Fig 14.The ROC curves of four recommender systems on the Movie Lens dataset

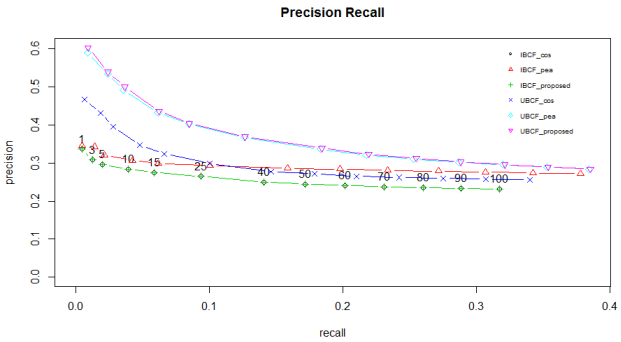

Fig 15. The Precision – Recall curves on the Movie Lens data set

VI. CONCLUSION

The traditional generic similarity measures, like Pearson correlation coefficient, the cosine are not enough to capture effective similar users, especially for cold start users who only experience no or small number of items. The proposed Adaptive Similarity Measure Model method address this problem and the results show that this method works well compared to other methods of similarity. In addition, in the future, the proposed method Adaptive Similarity Measure Model may be implemented on the other data sets.

REFERENCES

1. B. Shumeet, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, M. Aly, Video suggestion and discovery for YouTube: taking random walks through the view graph, in: International Conference on World Wide Web, 2008, pp. 895–904.
2. E. Brynjolfsson, Y.J. Hu, M.D. Smith, Consumer surplus in the digital economy: estimating the value of increased product variety at online booksellers, Manage.

Sci. 49 (11) (2003) 1580–1596.
3. J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, 1998, pp. 43–52.
4. F. Cacheda, V. Carneiro, D. Fernández, V. Formoso, Comparison of collaborative filtering algorithms: limitations of current techniques and proposals for scalable, high-performance recommender system, ACM Trans. Web 5 (1) (2011)
5. M.J. Pazzani, D. Billsus, Content-based recommendation systems, The Adap. Web (2007) 325–341.
6. H. Junming, C. Xueqi, G. Jiafeng, S. Huawei, Y. Kun, Social recommendation with interpersonal influence, ECAI 10 (2010) 601–606.
7. L. Jie, S. Qusai, X. Yisi, L. Qing and Z. Guangquan. BizSeeker: a hybrid semantic recommendation system for personalized government-to-business e-services, Internet Res. 20(3) (2010) 342–365.
8. Xiaoyuan Su and Taghi M. Khoshgoftaar, A Survey of Collaborative Filtering Techniques, Advances in Artificial Intelligence Volume 2009 (2009), Article ID 421425.
9. G. Karypis, “Evaluation of item-based top-N recommendation algorithms,” in Proceedings of the International Conference on Information and Knowledge Management (CIKM '01), pp. 247–254, Atlanta, Ga, USA, November 2001.
10. M. Deshpande and G. Karypis, “Item-based top-N recommendation algorithms,” ACM Transactions on Information Systems, vol. 22, no. 1, pp. 143–177, 2004.
11. N. Zheng, L. Qiudan, L. Shengcai, Z. Leiming, Which photo groups should I choose? A comparative study of recommendation algorithms in Flickr, J. Inform. Sci. 36 (6) (2010) 732–750.
12. M. Ye, P. Yin, W.C. Lee, Location recommendation for location-based social networks, in: Proceedings of the SIGSPATIAL International Conference on Advance in Geographic Information Systems, 2010, pp. 458–461.