International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 6
June 2017

Similarity Measures used in Recommender Systems: A Study

Ajay Agarwal
KIET group of institutions, Ghaziabad
Minakshi Chauhan
KIET group of institutions, Ghaziabad

ABSTRACT:
Information is growing exponentially over the Internet. User gets confused while seeing so many items over the Internet to decide which one to buy. In this scenario filtering of available information is essential to suggest user about items and tell what other users recommend. One user will set his mind to buy only if many like-minded users like a particular item. To get the group of similar users or items the vendor has to find similarity among them by quantifying their recommendations. This is feasible by using user-item rating matrix so that the system can show recommendations for users. This paper describes various methods to calculate similarity between users, and between items over a proposed rating matrix. This paper also highlights their advantages and disadvantages.

KEYWORDS: recommender system, collaborative filtering, similarity measure

INTRODUCTION:

Nowadays, more and more people are using internet on their own mobile phone, tablet PC and other intelligent terminals. This has enabled them to spend more time in accessing all kinds of e-commerce sites (such as flipkart and Amazon). However, the huge amount of available information and items makes them weighed down and indecisive. Users have to waste more time and energy in searching for their expected information. Even then, they find themselves lost in getting desired results. Fortunately, the behaviours of users can be tracked and recorded on these e-commerce sites. This makes it easier to analyze the preference of users. In this regard, recommender systems are used to recommend information as per user expectations and provide services by analyzing the user behaviours, such as the recommendation of videos in YouTube [1], the books in Amazon [2] and so on.

On the basis of their approach to rating estimation, recommender systems are usually classified:

(i) Content-based System
(ii) Collaborative Filtering System
(iii) Hybrid System

In content-based approach, similar items the user preferred in past will be recommended to the user while in collaborative filtering, items that other people with similar tastes and preferences like will be recommended. In order to overcome the limitations of both approach hybrid systems are proposed that combines both approaches in some manner. Among all the above methods, the collaborative filtering [3] has become the most commonly used method to recommend items for users. It makes recommendation based on the similar users with the active user. It also makes such recommendations using similar items with the items which are rated by the active user. The collaborative filtering comprises of memory-based method and model-based method [4]. In memory-based method similarities among users are first calculated then most similar users as the neighbors of the active user are selected. Finally, it provides the recommendations according to the neighbors. However, in the model-based method first a model is constructed to describe the behavior of users and, therefore, to predict the ratings of items. The memory-based method can provide significant recommended accuracy, but the execution time will grow rapidly with the increasing of users and items. Sometimes, it gets difficult to respond in real-time. On the other hand, the model-based method proves to be faster in prediction time than the memory-based method. It may be because the construction of the model can be finished in a considerable amount of time and this process is executed off-line. The demerit of the model-based method is that the recommendation performance is not as good as of the memory-based method. In addition to collaborative filtering, content-based technique [5], social recommendation [6], semantic recommendation [7] are also used to predict user preference. This paper focuses on the recommended performance in memory-based collaborative filtering algorithms. The core of collaborative filtering is to calculate similarities among users or items. The performance of a recommender system depends upon the accuracy; how accurately the recommender system makes predictions. The prediction ability depends on similarity measure used to find similar users. The prediction will get as improved as much similarity measure provides better results. Similarity can be found between users as well as items. Many researchers have been proposed many methods to compute similarity. [8] has categorized similarity measuring methods into three categories; correlation based similarity (Pearson correlation, Constrained Pearson correlation, Spearman, Kendall's τ correlation etc), Vector cosine based similarity, conditional probability based similarity methods [9], [10].

Section 2 describes related work of recommender systems. All Similarity measures used in recommender system are presented in Section 3. Section 4 has results and discussion. Concluding remarks are given in Section 5.

2. RELATED WORK

Collaborative filtering (CF), as a type of personalized recommendation method, has been extensively used in many domains [1,2,11-13]. However, collaborative filtering also suffers from a number of issues, for instance, data sparsity, cold start problem, scalability and so on. These problems badly diminish the user experience. Collaborative filtering recommends items to users according to their ratings. Therefore, a history database of users' ratings is desired. However, the data sparsity problem is always there, as user only rates a very few of items. This paper analyzed the disadvantages of Pearson correlation coefficient [14] and cosine similarity [15]. Ahn [16] proposed a new similarity measure called PIP (Proximity-Impact-Popularity). This new similarity considered three aspects: proximity, impact and popularity of the user ratings. But, this similarity considers only the local information about the ratings and does not consider the global preference of user ratings. General Pearson correlation coefficient does not take into the account of the size of the set of common users. To resolve the problem, weighted Pearson correlation coefficient has been introduced [17]. The basic idea of capturing the confidence is considered which can be placed on the neighbor. The confidence will naturally augment with the number of common rated items. Ester and Jamali [18] proposed a similarity measure based on the sigmoid function. This technique can deteriorate the similarity of small common items among users. The adjusted cosine similarity measure [16] was proposed to add up to the shortage of traditional cosine similarity. Bobadilla et al. [19] projected a new metric which combined the Jaccard measure [20] and mean squared difference [4]. The assumption made was that these two measures could complement each other. MJD (Mean–Jaccard–Difference) was anticipated to solve the cold user problem. This metric includes three steps: first the selection of similarity measures, the new metric has six similarity measures after this step. Then, the weights of each similarity measure will be evaluated by neural network learning. Finally, the prediction can be obtained according to the new metric. Recently, a singularity based similarity measure [21] was also offered. This measure proposed that the results obtained by applying traditional similarity measures could be improved by taking contextual information. This paper first considered the rating as positive and non-positive. Then the singularity values of each user and each item is computed. The similarity is replaced with singularity value. Moreover, Bobadilla et al. [22] introduced a significance based similarity measure. This measure first calculates three kinds of significances, which is the significance of an item, the significance of a user to recommend to other users and the significance of an item for a user. Then the traditional Pearson correlation coefficient or cosine similarity will be used to calculate the similarities among users according to the significance. Data smoothing technique is another most used method to improve the recommend performance in collaborative filtering. Various sparsity measures [23] were used to enhance accuracy of collaborative filtering. These sparsity measures were computed based on local and global similarities. Then, an estimating parameter scheme for weighting the various sparsity measures was proposed.

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 6
June 2017

The experimental results demonstrated that the proposed estimate parameter outperforms the schemes for which the parameter was kept constant on accuracy of prediction ratings.

3. SIMILARITY MEASURES:

In this section various similarity measures have been explained.

3.1 PEARSON CORRELATION: This method [14] is most commonly used method. This method is used to find linear correlation between two vectors. PCC results a value between -1 and +1. -1 represents a negative co relation while +1 represents high positive correlation. 0 value shows no relation sometimes called zero order correlation. For the user-based algorithm, Pearson correlation is given in Table 1.

3.2 CONSTRAINED PEARSON CORRELATION: Constrained Pearson correlation uses median value instead of average of rating co-rated by both users. Median value of scale [1-5] is 3. For the user-based algorithm, Constrained Pearson correlation is given in Table 1.

3.3 COSINE SIMILARITY: This method **[15]** is also most commonly used method in collaborative filtering in recommender systems. Cosine similarity finds how two vectors are related to each other using measuring cosine angle between these vectors. For the user-based algorithm, Cosine similarity is given in Table 1. The major drawback with cosine similarity is that it considers null preferences as negative preference.

3.4 ADJUSTED COSINE SIMILARITY: Cosine similarity measure **[24]** does not consider the scenario in which different users use different rating scale. Adjusted cosine similarity solves it by subtracting the average rating provided by the user u. Adjusted cosine similarity considers the difference in rating scale used by each user. Adjusted cosine similarity is slightly different from Pearson Correlation; Pearson Correlation considers the average rating of user u for co-rated the items. Adjusted cosine similarity subtracts the average rating of user u for all the items rated by user u. For the user-based algorithm, Adjusted cosine similarity is given in Table 1.

3.5 JACCARD SIMILARITY: Jaccard similarity [25] takes number of preferences common between two users into account. This does not consider the absolute ratings rather it considers number of items rated. Two users will be more similar, when two users have more common rated items. For the user-based algorithm, Jaccard similarity is given in Table 1. Jaccard produces limited number of values which makes the task of user distinction difficult.

3.6 MEAN SQUARED DIFFERENCES:

For the user-based algorithm, MSD similarity is given in Table 1. MSD does not consider number of common rating rather it considers absolute ratings. Various similarity measures have been proposed in combination of jaccard similarity as JMSD [19], JPSS [26]. Jaccard and MSD similarity can be combined to form another similarity measure method JMSD. For the user-based algorithm, JMSD similarity is given in Table 1.

3.7 PIP SIMILARITY: PIP [16] stands for Proximity, Impact, Popularity. Proximity factor calculates the arithmetic difference between two ratings, along with consideration of agreement or disagreement of ratings, giving penalty to ratings in disagreement. Two ratings occurs in agreement if they lie on one side of the median of the rating scale. The Impact factor represents the extent to which an item is preferred or disliked by users. If two users have rated 1 on an item, it will show more strong dislike than they rate 3. Popularity is calculated around average rating of item provided by all users. Popularity gives high similarity when average of two rating far from average ratings of the item. If both users average rating has a large difference with the average of total users' ratings, the two ratings can provide more information about the similarity of the two users. For the user-based algorithm, PIP similarity is given in Table 1. The Development of PIP was based on utilizing domain specific interpretation of user ratings on products It was developed to overcome the weakness of traditional similarity and distance measures in new user cold-start conditions. PIP performed well for new user cold-start conditions. PIP penalizes on proximity as well as Impact when there is disagreement in ratings. Sometimes it misleads about the similarity between similar users and similarity between dissimilar users. To overcome the problem faced by PIP, a new similarity measure was developed based on sigmoid function, called PSS an improved PIP measure. PSS stands for Proximity, significance, similarity. For the user-based

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 6
June 2017

algorithm, PSS similarity is given in Table 1. Proximity of two ratings is computed as in PIP. The second factor was taken as significance. Significance of two ratings was based on the median value of rating scale. It is based on the concept that the ratings, more distant from the median value, would be more significant. Singularity defines how two rating different from other ratings. PSS can be combined with Jaccard similarity measure. For the user-based algorithm, JPSS is given in Table 1.Where JPSS uses an improved version of Jaccard similarity measure.

3.8 NEW HEURISTIC SIMILARITY MODEL (NHSM): For this method [26], PIP has been taken as initial heuristic method. NHSM similarity measure is combination of JPSS and User Rating Preference similarity measures. User Rating Preference similarity measure is based on mean and variance of the ratings of user. For the user-based algorithm, NHSM similarity measure is given in Table 1. The value produced by NHSM ranges from 0 to 1. This similarity measure considers the fact that different users have different preferences scale and models user preference based on mean and standard variance of user ratings.

3.9 SPEARMAN RANK CORRELATION: Spearman Rank Correlation uses ranks instead of ratings for calculating similarity. For the user-based algorithm, Spearman Correlation is given in Table 1. Spearman Rank Correlation does not work well for partial orderings [27]. Weak orderings occur whenever there are at least two items in the ranking such that neither item is preferred over the other. If there is difference between user ranking ordering and system ordering. When the system ranks same rated items at different levels, then the Spearman correlation will be penalized for every pair of items rated same by the user. Since the user shouldn't care the system orders items that the user has rated at the same level. Kendall's Tau metric also suffers from the same.

3.10 KENDALL'S TAU CORRELATION: This [27] is also rank based method to compute correlation. Kendall's correlation considers relative ranks instead of ratings for calculating similarity. It computes the correlation of the rankings that is independent of the variable values. Kendall's correlation suffers from the same problem faced by Spearman correlation.

Table 1: Equations of all similarity measures

S. No.	Name	Formula
1	Pearson correlation	$PCC\_Sim(u,v)=\frac{\sum_{i\in I}(r_{u,i}-\bar{r}_{u,I})(r_{v,i}-\bar{r}_{v,I})}{\sqrt{\sum_{i\in I}(r_{u,i}-\bar{r}_{u,I})^2+\sum_{i\in I}(r_{v,i}-\bar{r}_{v,I})^2}}$ **Where** $\bar{r}_{u,I}$ **and** $\bar{r}_{v,I}$ **represents** Average rating of user u and user v , respectively ,for co-rated items represented by set I
2	Constrained Pearson correlation	$CPCC\_sim(u,v)\frac{\sum_{i\in I}(r_{u,i}-r_{med})(r_{v,i}-r_{med})}{\sqrt{\sum_{i\in I}(r_{u,i}-r_{med})^2+\sum_{i\in I}(r_{v,i}-r_{med})^2}}$ r_{med} Median value for rating scale.
3	Cosine similarity	Cos_Sim(u,v)= $\frac{\vec{R}_u\bullet\vec{R}_v}{\lvert\vec{R}_u\rvert\cdot\lvert\vec{R}_v\rvert}$ Where "$\bullet$" represents dot product of two vectors. $\vec{R}_u$ and $\vec{R}_v$ are rating vectors of user u and v respectively.
4	Adjusted cosine similarity	$Adjusted\_COS\_Sim=\frac{\sum_{i\in I}(r_{u,i}-\bar{r}_u)(r_{v,i}-\bar{r}_v)}{\sqrt{\sum_{i\in I}(r_{u,i}-\bar{r}_u)^2+\sum_{i\in I}(r_{v,i}-\bar{r}_v)^2}}$ where $\bar{r}_i$ Average rating of user i for all items rated by user i itself.

5	Jaccard Similarity	$Jaccard\_sim(u,v)=\frac{\lvert I_u \cap I_v\rvert}{\lvert I_u \cup I_v\rvert}$ Where I_u is a set of items rated by user u and I_v is a set of items rated by user v.
6	Mean Squared Differences	$MSD(u,v)=\sum_{i\in I}(r_{u,i}-r_{v,i})^2$ $MSD\_Sim(u,v)=\frac{L-MSD(u,v)}{L}$ Where L is a threshold
7	PIP similarity	$PIP\_Sim(u,v)=\sum PIP(r_{u,i}-r_{v,i})$ $PIP(r_{u,i},r_{v,i})=\Pr oximity(r_{u,i},r_{v,i}).\mathrm{Im}\,pact(r_{u,i},r_{v,i}).Popularity(r_{u,i},r_{v,i})$
8	New Heuristic Similarity Model	$URP\_Sim(u,v)=1-\frac{1}{1+\exp(-\lvert\bar{r}_u-\bar{r}_v\rvert.\lvert\pm_u-\pm_v\rvert)}$ $\pm_u=\sqrt{\sum_{i\in I_u}(r_{u,i}-\bar{r}_u)^2/\lvert I_u\rvert}$ $NHSM\_Sim(u,v)=JPSS\_Sim(u,v).URP\_Sim(u,v)$ Where where $\bar{r}_i$ Average rating of user i for all items rated by user and $\pm_u$ and $\pm_v$ are the standard variance of user u and v respectively.
9	Spearman Rank Correlation	$Spearman\_sim(u,v)=1-\frac{6\sum_{h_0}^{n_i}d_h^{\,2}}{n_i(n_i^{\,2}-1)}$ Where d_h difference in ranks of item h co-rated by both users. n_i is Number of items co-rated by both users.
10	Kendall's Tau correlation	$kendall\_Tau=\frac{N_c-N_d}{\sqrt{(N_c+N_d+T_r)(N_c+N_d+T_p)}}$ Where N_c is number of concondant pairs items that the system predicts in the proper ranked order. N_d is the number of discordant pairs that the system predicts in the wrong order. T_r is number of pairs of items that the system predicts the true ordering while T_P is the number of pairs of items in the predicted ordering

In Table 2, new similarity measures are defined as a combination of previously defined similarity measures.
Table 2: New similarity measures in terms of similarity measures defined in Table 1.

S. No.	Name	Formula
11	Jaccard Mean Squared Differences	$JMSD\_Sim(u,v)=Jaccard\_Sim(u,v).MSD\_Sim(u,v)$
12	PSS similarity	$PSS\_Sim(u,v)=\sum PSS(r_{u,i}-r_{v,i})$ $PSS(r_{u,i},r_{v,i})=\Pr oximity(r_{u,i},r_{v,i}).Significance(r_{u,i},r_{v,i}).\mathrm{Sin}\,gularity(r_{u,i},r_{v,i})$
13	Jaccard PSS similarity	$JPSS\_sim(u,v)=Jaccard'\_Sim(u,v).PSS\_Sim(u,v)$ $Jaccard'\_sim(u,v)=\frac{\lvert I_u\cap I_v\rvert}{\lvert I_u\rvert\times\lvert I_v\rvert}$

4. RESULTS AND DISCUSSION:

In this paper similarity between users has been calculated over a proposed rating matrix shown in Figure 1 using some of similarity measuring methods mentioned in Table 1. User preferences for items can be represented as formally matrix of user-item matrix R_{NxM} where N is numbers of users and M is number of items. U represents set of all users and P represents the set of all items. Let I be the set of co-rated items by two users. $r_{u,i}$ reprsents rating of user u for item i. Sample matrix showing user ratings of five users for five items is given in Figure 1 where U={u1, u2, u3, u4, u5}, P={i1,i2,i3,i4,i5}, $I_{u1,u2}$={i1,i3,i4}and $r_{u1,i1}$=5

item user	i1	i2	i3	i4	i5
u1	5		0.5	1	5
u2	2		5	4	
u3	4	3	1	4	5
u4	4	1	5	5	2
u5	5	1		4	1

Figure 1: Matrix of users (u1,u2,u3,u4,u5) preferences on items (i1,i2,i3,i4,i5)

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} 0.344 & 0.845 & 0.622 & 0.724 \\ & 0.528 & 0.938 & 0.591 \\ & & 0.783 & 0.820 \\ & & & 0.778 \end{bmatrix} \end{array}$$

(a) **COS_ Sim** between users

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} -0.973 & 0.723 & -0.814 & -0.277 \\ & -0.756 & 0.945 & -1 \\ & & -0.345 & 0 \\ & & & 0.885 \end{bmatrix} \end{array}$$

(b) **PCC _Sim** between users

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} -0.833 & 0.721 & -0.731 & 0.296 \\ & -0.771 & 0.585 & -0.033 \\ & & -0.580 & 0.472 \\ & & & 0.124 \end{bmatrix} \end{array}$$

(c)**Adj_COS** sim between users

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} -0.780 & 0.670 & -0.471 & -0.070 \\ & -0.367 & 0.661 & -0.013 \\ & & -0.128 & 0 \\ & & & 0.716 \end{bmatrix} \end{array}$$

(d) **CPCC _Sim** between users

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} 0.75 & 0.8 & 0.8 & 0.6 \\ & 0.6 & 0.6 & 0.4 \\ & & 1 & 0.8 \\ & & & 0.8 \end{bmatrix} \end{array}$$

(e) **Jaccard_Sim** between users

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} 0.490 & 0.897 & 0.538 & 0.667 \\ & 0.733 & 0.933 & 0.820 \\ & & 0.760 & 0.790 \\ & & & 0.970 \end{bmatrix} \end{array}$$

(f) **MSD** Sim between users

$$\begin{array}{cc} & \begin{array}{cccc} u1 & u2 & u3 & u4 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} 0.368 & 0.718 & 0.430 & 0.4 \\ & 0.440 & 0.560 & 0.328 \\ & & 0.760 & 0.632 \\ & & & 0.776 \end{bmatrix} \end{array}$$

(g) **JMSD Sim** between users

$$\begin{array}{cc} & \begin{array}{cccc} u2 & u3 & u4 & u5 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \end{array} & \begin{bmatrix} 0.158 & 0.015 & 0.051 & 0.091 \\ & 0.243 & 0 & 0.092 \\ & & 0.174 & 0.013 \\ & & & 0.101 \end{bmatrix} \end{array}$$

(h)**Spearman Rank** Sim between users

5. CONCLUSION:

The generic traditional similarity measures, such as Pearson correlation coefficient, cosine, Mean Squared Difference are not enough to capture the effective similar users, especially for cold user who only rates a small number of items. Further, in future these similarity measures should be implemented on mostly used real data sets in order to get their real feedback/output.

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) 1–33.

[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.

[13] Q. Liu, T. Chen, J. Cai, D. Yu, Enlister: baidu's recommender systems for the biggest Chinese Q&A website, in: Proceedings of the ACM Conference on Recommender Systems, 2012, pp. 285–288.

[14] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, GroupLens: an open architecture for collaborative filtering of netnews, in: Proceeding of the ACM Conference on Computer Supported Cooperative Work, 1994, pp. 175–186.

[15] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.

[16] H.J. Ahn, A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem, Inform. Sci. 178 (1) (2008) 37–51.

[17] J.L. Herlocker, J.A. Konstan, A. Borchers, J. Riedl, An algorithmic framework for performing collaborative filtering, in: Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1999, pp. 230–237.

[18] M. Jamali, M. Ester, TrustWalker: a random walk model for combining trustbased and item-based recommendation, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 397–406.

[19] J. Bobadilla, F. Ortega, A. Hernando, J. Bernal, A collaborative filtering approach to mitigate the new user cold start problem, Knowledge-Based Syst. 26 (2011) 225–238.

[20] G. Koutrica, B. Bercovitz, H. Garcia, FlexRecs: expressing and combining flexible recommendations, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, 2009, pp. 745–758.

[21] J. Bobadilla, F. Ortega, A. Hernando, A collaborative filtering similarity measure based on singularities, Inform. Process. Manage. 48 (2012) 204–217.

[22] J. Bobadilla, A. Hernando, F. Orteqa, A. Gutirrez, Collaborative filtering based on significances, Inform. Sci. 185 (2012) 1–17.

[23] D. Anand, K.K. Bharadwaj, Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities, Expert Syst. Appl. 38 (2011) 5101–5109.

[24] Sarwar, Badrul; Karypis, George; Konstan, Joseph; Riedl, John (2001). "Item-based collaborative filtering recommendation algorithms". Proceedings of the 10th international conference on the World Wide Web. ACM: 285–295.

[25] Laurent Candillier, Frank Meyer, and Françoise Fessant. Designing Specific Weighted Similarity Measures to Improve Collaborative Filtering Systems. ICDM, pp. 242-255, 2008.

[26] Haifeng Liu , Zheng Hu, Ahmad Mian, Hui Tian, Xuzhen Zhu, A new user similarity model to improve the accuracy of collaborative filtering, Knowledge-Based Systems 56 (2014) 156–166.

[27] Jonathan l. Herlocker, joseph a. Konstan, loren g. Terveen, and john t. Riedl, Evaluating Collaborative Filtering Recommender Systems ,ACM Transactions on Information Systems, Vol. 22, No. 1, January 20