Collaborative Filtering Recommendation with Fuzzy-weighted User Similarity

Soojung Lee
Dept. of Computer Education
Gyeongin National University of Education
Anyang, Korea
E-mail: sjlee@ginue.ac.kr

***Abstract*—Similarity computation plays a critical role in collaborative filtering-based recommender systems. As these systems recommend items based on user ratings, they involve several inherent problems such as data sparsity, cold-start, scalability, and user subjectivity. Much effort has been devoted to handle these problems and simultaneously enhance the performance of the system, but there are still much to be improved. This study focuses on user-based collaborative filtering systems and proposes a new similarity measure which not only considers user ratings for common items but also reflects the rating behavior of all the users on each common item onto similarity. Performance of the proposed measure is investigated extensively under very different ratings data conditions. The results state that it mostly outperforms state-of-the-art similarity measures, where the degree of improvement is significantly high when it incorporates Pearson correlation.**

***Index Terms*—collaborative filtering, recommender system, similarity measure, fuzzy logic, user-based collaborative filtering**

I. Introduction

Collaborative filtering (CF) is the most popular technique used by recommender systems. This technique has been employed by many current commercial systems such as Amazon.com and eBay [1]–[3]. These systems are based on the principle of *memory-based methods* which search for similar users or items using the user ratings data and accumulate their ratings for unrated items to make recommendations. Whether similarity is computed between users or items determines the type of memory-based methods. The former is so-called *user-based* technique, whereas the latter is named *item-based*. That is, the item-based technique recommends items similar to those for which the current user has shown preferences. This study focuses on the user-based CF.

Similar users critically affect the reliability of the user-based CF systems. Thus various techniques in the literature have been devised to measure similarity between two users. Most famous ones are Pearson correlation and the cosine similarity, but other variants also exist. Other similarity measures include constrained Pearson correlation that uses midpoint instead of the mean, Spearman rank correlation, Kendall's τ correlation that uses relative ranks instead of the ranks used by Spearman rank correlation [2], [3].

However, existing similarity measures used for the user-based CF manipulate user ratings only, which is often insufficient to produce reliable similarities, especially when the ratings data are sparse. It is quite common in the current recommender systems that ratings data sparsity occurs in several situations. It occurs when a new user or item enters into the system, known as the *cold start problem* [1], [3]. But, the data sparsity problem arises mainly because many systems maintain a very large set of products.

Model-based CF techniques constitute another approach of CF systems to overcome such shortcomings of memory-based systems. They learn a model from ratings data using machine learning or data mining algorithms to make recommendations. Popular models include Bayesian belief nets, clustering, latent class models, and singular value decomposition (SVD) [3]. However, model construction is usually time-consuming and requires the estimation of many parameters, which is thereby too sensitive to data changes. Hence, several attempts have been made to compensate the shortcomings of the memory-based CF systems by combining additional information with the traditional similarity measures [4]–[10]. Such information includes the number of co-rated items, the entropy of ratings, singularity of ratings, etc. However, it is often heuristic-based or neglects the global rating behavior of users on items.

This study suggests a novel similarity measure for user-based CF systems. The proposed measure is intended to reflect the rating behavior of users on each item onto similarity computation in a more systematic manner. It computes fuzzy ranks of user ratings on an item and combines them with traditional similarity measures. We conduct extensive experiments using three well-known datasets with different characteristics and find that the proposed measure outperforms the state-of-the-art similarity measures in most results, especially when it incorporates Pearson correlation.

The rest of this paper is organized as follows. We present studies on similarity measures for CF systems in Section 2. In Section 3, a new similarity measure is proposed, followed by the experiments and results in Section 4. Section 5 concludes the paper.

II. Background and Related Works

A. Recommendation Procedure

In order to recommend items to an active user u, collaborative filtering systems follow the procedure below [2].

1. Find a set of similar users (nearest neighbors), NN, according to a similarity measure chosen by the system.

2. Estimate a rating that might be given by user u for an item x yet unseen by the user. One of the most well-known prediction formula takes the weighted average of all the ratings given by NN as follows [11].

$$\bar{r}_u + \frac{\sum_{v \in NN} sim(u,v)(r_{v,x} - \bar{r}_v)}{\sum_{v \in NN} |sim(u,v)|},$$

where $sim(u,v)$ is similarity between two users u and v, $r_{v,x}$ the rating given by user v to item x, and $\bar{r}_u$ the average rating by user u on all the other rated items.
3. Recommend those unseen items with predicted ratings higher than a predetermined threshold.

As figured out from the above procedure, determining the nearest neighbors affects the performance of collaborative filtering systems significantly, which indicates the importance of a similarity measure employed by the system.

B. Traditional Similarity Measures

In the literature, representative similarity measures include Pearson correlation, cosine similarity, and the mean squared differences [1], [3], [12]. Pearson correlation is a representative correlation-based similarity measure. It estimates similarity between users u and v as below.

$$COR(u,v) = \frac{\sum\limits_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum\limits_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)^2}\sqrt{\sum\limits_{i \in I_{u,v}} (r_{v,i} - \bar{r}_v)^2}}$$

In this equation, $I_{u,v}$ refers to the set of items co-rated by two users u and v, $r_{u,i}$ indicates the rating given by user u to item i, and $\bar{r}_u$ the average rating of user u of the items in $I_{u,v}$.

Similarity between two users can be also estimated by using the cosine similarity. It treats ratings of a user as a vector and computes the cosine angle between the two vectors of user ratings. Formally, the cosine similarity is defined as follows.

$$COS(u,v) = \frac{\sum\limits_{i \in I_{u,v}} r_{u,i} r_{v,i}}{\sqrt{\sum\limits_{i \in I_{u,v}} r_{u,i}^2}\sqrt{\sum\limits_{i \in I_{u,v}} r_{v,i}^2}}$$

Another popular similarity measure is the mean squared differences. It computes the mean squared difference between normalized ratings of two users for each common item. Specifically,

$$MSD(u,v) = 1 - \frac{1}{|I_{u,v}|} \sum_{i \in I_{u,v}} (r'_{u,i} - r'_{v,i})^2,$$

where $r'_{u,i}$ is a normalized rating of $r_{u,i}$ within [0, 1].

As all of these measures derive similarity from the ratings given to the common items by two users, they suffer from the cold-start problem [13] or the data sparsity problem which often makes the resulting similarity values unreliable.

C. Weight-based Similarity Measures

In order to overcome shortcomings of the traditional similarity measures, many researches devised various functions to be combined with the measures as weights [6], [9], [14]–[17]. These functions usually make use of the number of items co-rated by the two users for whom similarity computation is made. This approach is based on the assumption that two users with more commonly rated items would demonstrate higher similarity between their ratings. Examples of such function include a sigmoid function of the number of common users [6], the degree of rating overlap [18], and the ratio of the number of co-rated items named *Jaccard index* [15]. Although these strategies of combining weights with previous similarity measures are simple, they basically rely on the number of ratings for common items, disregarding any context information inherent in them.

As another type of weight, fuzzification is adopted to reflect vagueness and subjectivity of the user ratings on similarity computation. Son integrates the fuzzy similarity based on the users' demographic data with Pearson correlation [10]. It is reported that this method obtains higher accuracy than other relevant methods. The study presented in [4] incorporated the fuzzy rating values and rating deviation values into existing similarity metrics as weights.

Recently, some studies employed the concept of entropy proposed by Shannon [19] into similarity computation [7], [19]–[23]. In [22], the rating of an unrated item is complemented by the entropy. Li and Zheng measure similarity based on Pearson correlation but take Bhattacharyya Coefficient and entropy into account [21]. Kwon et al. estimated the entropy of ratings of each user and incorporated the entropy difference between two users into the conventional similarity measures [7]. Wang et al. measures the relative difference between ratings using the entropy to be combined with Pearson correlation [23]. Although the entropy is a useful tool to improve the quality of the CF system, it is mostly estimated with respect to a user, instead of an item. Hence, its effectiveness is largely dependent on the number of ratings given by a user. As another solution using entropy as weight, Lee estimated entropy with respect to each item and combined it with the conventional similarity measures [8]. The study is distinct from previous ones in that the merge of the entropy weight is done at each item level.

There have been few studies on developing weights with respect to an item for similarity computation. The work in [5] suggested a new definition of weight named *singularity* which is incorporated into the mean squared differences. Singularity measures the degree of uniqueness of a high or low rating on each item. Two similar ratings both with higher singularities contribute to higher similarity. However, the similarity measure proposed by their study requires to determine proper thresholds for distinguishing between high and low ratings for performance. Also, these thresholds equally apply to all the items regardless of the statistical distributions of ratings associated with the items.

TABLE I
ILLUSTRATION OF USER RATINGS AND SIMILARITY

	$u1$	$u2$	$u3$	$u4$	$u5$	$u6$	$u7$	$u8$
$i1$	2	4	8	3	2	6	9	4
$i2$	4	6	4	6	6	5	4	5
Pearson correlation ($u1$, $u2$)							1.0	
cosine similarity ($u1$, $u2$)							0.9923	
mean squared differences ($u1$, $u2$)							0.9506	

III. PROPOSED MEASURE

A. Motivation

The proposed similarity measure aims to reflect the rating behavior of users for each item. Table I illustrates ratings made by eight users $u1$ to $u8$ on two items $i1$ and $i2$. Consider similarity between users $u1$ and $u2$. The table also presents the similarity computed using the three conventional measures.

The rating differences made by users $u1$ and $u2$ on the two items are the same, i.e., two, but they can be interpreted differently considering the ratings by other users. Assuming the rating range of the system is [1,10], the rating difference on $i1$ between the two users is relatively much smaller than that on $i2$. The latter difference is in fact the maximum among all the differences on $i2$. Nevertheless, it is observed that Pearson correlation yields the similarity of one. This implies that Pearson correlation would be most improved when our strategy of reflecting the global rating behavior is applied.

B. Fuzzy Functions

To implement the observation above, we exploit the fuzzy ranks of ratings on each item for similarity computation between two users. Let $k_{u,i}$ be the rank of a rating by user u given to item i with respect to all the user ratings for i. Also let $k_{u,i}$ be represented using membership values of n fuzzy sets as $< m^1_{u,i}, m^2_{u,i}, \ldots, m^n_{u,i} >$. We use three different functions shown in Fig. 1 for fuzzification of ranks. For instance, if $k_{u,i} = 4$ and the maximum rank (max) is 5, the rank can be substituted by <0.25, 0.75> using the function in Fig. 1(a). In the next section, we experiment with all types of fuzzy functions shown in Fig. 1 and discover the function yielding the best performance.

The definition of the membership functions can be easily formulated from Fig. 1. Specifically, a rank k is converted into

$$m^1(k) = \frac{max - k}{max - 1}$$

$$m^2(k) = \frac{k-1}{max-1},$$

according to Fig. 1(a). Using the three fuzzy sets in Fig. 1(b), when d indicates the median, $(max+1)/2$,

$$m^1(k) = \begin{cases} \dfrac{k}{1-d} - \dfrac{d}{1-d} & \text{if } 1 \le k \le d \\ 0, & \text{if } d \le k \le max \end{cases}$$

$$m^2(k) = \begin{cases} \dfrac{k}{d-1} - \dfrac{1}{d-1}, & \text{if } 1 \le k \le d \\ \dfrac{k}{d-max} + \dfrac{max}{max-d}, & \text{if } d \le k \le max \end{cases}$$

$$m^3(k) = \begin{cases} 0, & \text{if } 1 \le k \le d \\ \dfrac{k}{max-d} - \dfrac{d}{max-d}, & \text{if } d \le k \le max \end{cases}$$

The five fuzzy sets in Fig. 1(c) are defined as follows, where $d = (max-1)/9$.

$$m^1(k) = \begin{cases} 1, & \text{if } 1 \le k \le 1+d \\ -\dfrac{k}{3d} + \dfrac{1}{3d}(1+4d), & \text{if } 1+d \le k \le 1+4d \\ 0, & \text{if } k \ge 1+4d \end{cases}$$

$$m^2(k) = \begin{cases} \dfrac{k}{3d} - \dfrac{1}{3d}(1-d), & \text{if } 1 \le k \le 1+2d \\ 1, & \text{if } 1+2d \le k \le 1+3d \\ -\dfrac{k}{3d} + \dfrac{1}{3d}(1+6d), & \text{if } 1+3d \le k \le 1+6d \\ 0, & \text{if } k \ge 1+6d \end{cases}$$

$$m^3(k) = \begin{cases} \dfrac{k}{3d} - \dfrac{1}{3d}(1+d), & \text{if } 1+d \le k \le 1+4d \\ 1, & \text{if } 1+4d \le k \le 1+5d \\ -\dfrac{k}{3d} + \dfrac{1}{3d}(1+8d), & \text{if } 1+5d \le k \le 1+8d \\ 0, & \text{if } k \le 1+d, k \ge 1+8d \end{cases}$$

$$m^4(k) = \begin{cases} \dfrac{k}{3d} - \dfrac{1}{3d}(1+3d), & \text{if } 1+3d \le k \le 1+6d \\ 1, & \text{if } 1+6d \le k \le 1+7d \\ -\dfrac{k}{3d} + \dfrac{1}{3d}(1+10d), & \text{if } 1+7d \le k \le max \\ 0, & \text{if } k \le 1+3d \end{cases}$$

$$m^5(k) = \begin{cases} \dfrac{k}{3d} - \dfrac{1}{3d}(1+5d), & \text{if } 1+5d \le k \le 1+8d \\ 1, & \text{if } 1+8d \le k \\ 0, & \text{if } k \le 1+5d \end{cases}$$

C. Proposed Similarity Measures

Now that all the fuzzy sets are defined in the previous subsection, we propose a weight measure which favors a smaller difference between two fuzzy ranks of ratings on an item co-rated by two users u and v. The weight on item i for users u and v is defined as

$$w_{u,v}(i) = 1 - \frac{1}{n}\sum_{j=1}^{n} |m^j_{u,i} - m^j_{v,i}|.$$

Using the ratings listed in Table I, we compute fuzzy ranks using two membership functions depicted in Fig. 1(a). The fuzzy ranks of ratings for the two items $i1$ and $i2$ are presented in Table II. Then the weights for users $u1$ and $u2$ are calculated as $w_{u1,u2}(i1)$=0.57 and $w_{u1,u2}(i2)$=0. Note that for item $i2$, the rating difference between the users is locally the maximum, thus yielding the lowest weight.

We incorporate this weight for each item into most representative similarity measures in CF systems, Pearson correlation

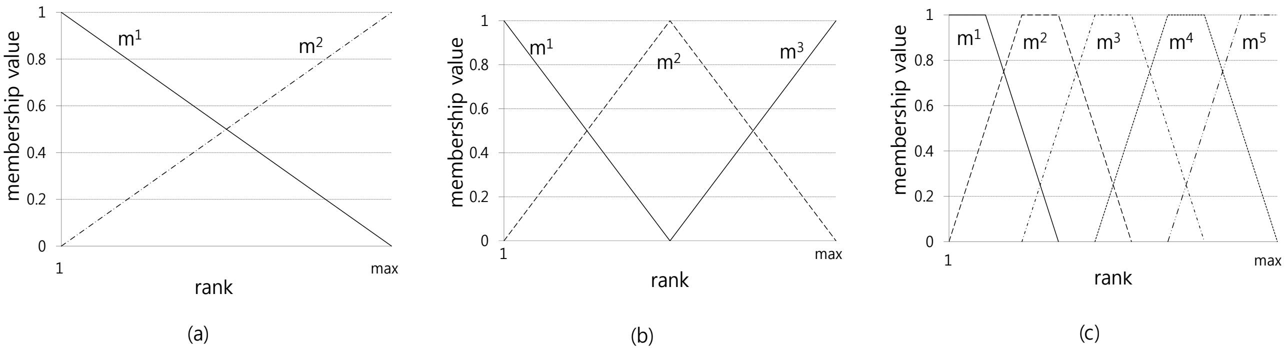

Fig. 1. Fuzzy functions used for experiments

TABLE II
ILLUSTRATION OF USER RATINGS AND FUZZY RANKS

	$u1$	$u2$	$u3$	$u4$	$u5$	$u6$	$u7$	$u8$
ratings for $i1$	2	4	8	3	2	6	9	4
ranks for $i1$	1	4	7	3	1	6	8	4
fuzzy ranks for $i1$	$< 1, 0 >$	$< 0.57, 0.43 >$	$< 0.14, 0.86 >$	$< 0.71, 0.29 >$	$< 1, 0 >$	$< 0.29, 0.71 >$	$< 0, 1 >$	$< 0.57, 0.43 >$
ratings for $i2$	4	6	4	6	6	5	4	5
ranks for $i2$	1	6	1	6	6	4	1	4
fuzzy ranks for $i2$	$< 1, 0 >$	$< 0, 1 >$	$< 1, 0 >$	$< 0, 1 >$	$< 0, 1 >$	$< 0.4, 0.6 >$	$< 1, 0 >$	$< 0.4, 0.6 >$

(COR) and the cosine similarity (COS). Our similarity measures, denoted as COR_{FRank} and COS_{FRank}, between two users u and v, are formally defined below, where $I_{u,v}$ refers to the set of items co-rated by the two users. In the definition, $r_{u,i}$ indicates the rating given by user u to item i and $\bar{r}_u$ the average rating of user u of the items in $I_{u,v}$.

$$COR_{FRank}(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v) w_{u,v}(i)}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \bar{r}_v)^2}}$$

$$COS_{FRank}(u,v) = \frac{\sum_{i \in I_{u,v}} r_{u,i} r_{v,i} w_{u,v}(i)}{\sqrt{\sum_{i \in I_{u,v}} r_{u,i}^2} \sqrt{\sum_{i \in I_{u,v}} r_{v,i}^2}}$$

Furthermore, we incorporate the weight into another popular similarity measure, the mean squared differences (MSD) to suggest a new measure. Specifically,

$$MSD_{FRank}(u,v) = 1 - \frac{1}{|I_{u,v}|} \sum_{i \in I_{u,v}} (r'_{u,i} - r'_{v,i})^2 w_{u,v}^{MSD}(i),$$

where the normalized rating $r'_{u,i}$ is computed as $(r_{u,i} - r_{min})/(r_{max} - r_{min})$. Here, r_{max} and r_{min} are the maximum and minimum ratings allowed by the system, respectively. The weight described above also applies to MSD_{FRank}, but it is modified as follows due to the definition of MSD.

$$w_{u,v}^{MSD}(i) = \frac{1}{n} \sum_{j=1}^{n} |m_{u,i}^j - m_{v,i}^j|.$$

The proposed measures have basically the same principle with the ones suggested by [5] and [8]. The only difference is on the definition of weight and the type of traditional similarity measure to merge with. The weight in the similarity measure presented in [5] utilizes the concept of singularity. Higher weight is assigned to a rare high or low rating. Hence, this method may not be efficient when the ratings data are dense and the range of ratings is small, as demonstrated through some experiments in literature [8]. Moreover, it needs to determine thresholds for high and low ratings, which is critical for performance. On the other hand, [8] takes the entropy of ratings for each item as weight. Thus, it assigns the same weight to the items whose associated entropies are the same, no matter what their ratings are. The proposed measures are different from these measures in that the weight takes each specific rating of an item into account and are not dependent on any performance parameter.

IV. PERFORMANCE RESULTS

A. Design of Experiments

We examined performance of the CF system with the proposed similarity measures implemented. Table III lists three popular datasets used in the related field. These datasets have very different characteristics from each other, thus purposely chosen to investigate the performance under various data environment. Sparsity level represents the degree of sparseness of the ratings and is computed as $1 - \frac{total\ number\ of\ ratings}{matrix\ size}$. Due to the limited capacity of the PC for experiments we selected subsets of the users and their associated ratings from the original datasets.

In the dataset, 80% of the ratings data are used for training data, i.e., to find the most similar users (nearest neighbors,

TABLE III
CHARACTERISTICS OF THE DATASETS

	MovieLens 1M	BookCrossing	Jester
Matrix size (users×ratings)	1000×3952	1014×883	998×100
Rating scale	1∼5 (integer)	1∼10 (integer)	$-10 \sim +10$ (real)
Sparsity level	0.9610	0.9775	0.2936
Number of ratings per item	1∼590	4∼168	322∼998

NN). The rest 20% of the data is to evaluate the performance of the collaborative filtering system using the similarity measure of concern. We employed two well-known indices to indicate performance, MAE (Mean Absolute Error) and F1. MAE measures the difference between the predicted rating of an unrated item and its real rating.

To evaluate recommendation quality of the system, we use F1, which is a typical metric combining precision (P) and recall (R) as a harmonic mean [23], i.e., $2PR/(P+R)$. Precision is the ratio of relevant items out of all the recommended items. Recall indicates the ratio of relevant recommended items out of all the relevant items. Considering the range of the rating scale provided by the dataset, we use the relevance threshold of four for MovieLens, eight for BookCrossing, and three for Jester dataset, respectively.

We experimented with three conventional similarity measures, Pearson correlation (COR), the cosine similarity (COS), and the mean squared differences (MSD). In addition, the entropy-weighted method suggested in [8] is implemented into these three, each of which is denoted as COR_E, COS_E, and MSD_E. Likewise, the proposed method is adopted into the three conventional measures and denoted as COR_FR, COS_FR, and MSD_FR, as described in the previous section. Finally, experimentation for Singularity Measure (SM) [5] is conducted.

B. Experimental Results

1) MovieLens Dataset: Fig. 2 presents performance results using MovieLens dataset as the number of nearest neighbors varies. Surprisingly, SM is overall significantly outperformed by all the other measures in both MAE and F1 metrics, even by the original conventional measure. Although performance of SM differs by the singularity threshold, the results indicate that its strategy is not effective with MovieLens.

Notice that the entropy-weighted method turns out to improve its base measure, especially when combined with COR and MSD. For COS, the improvement by COS_E seems insignificant. This in general demonstrates the efficiency of the entropy incorporation for MovieLens, even if the entropy is not expected to fluctuate severely because of its small range of the rating scale of the dataset.

Considering the results of the proposed measure, it is found that it achieves improvement over each of the corresponding base measures. Especially, the improvement is quite noticeable for COS_FR. One of the reasons might be that COS performs far worse than COR and MSD with MovieLens, thus leaving much to be improved, which is well achieved by the proposed measure. Therefore, utilization of fuzzy ranks leads to better improvement than by the entropy-weighted method, particularly in terms of F1. Among the three fuzzy functions in Fig. 1, the one with three sets shown in Fig. 1(b) yielded the best results, which is specified in the legend.

2) BookCrossing Dataset: We conducted experiments with a much sparser dataset, BookCrossing, and obtained the results quite different from those with MovieLens, as shown in Fig. 3. Observe a noticeable performance gap between the similarity measures associated with COR in terms of MAE, where COR_FR is outstandingly the best. This experiment proves that COR is not resilient to data sparseness, since it is worst with this dataset, whereas it performed relatively competitive using MovieLens as seen in Fig. 2. Such drawback of COR is best overcome by COR_FR, which is realized due to its manipulation of fuzzy ranks.

Exploiting entropy also seems to overcome such drawback of COR as shown in the results of COR_E in Fig. 3, but the strategy employed by SM is found to be more effective especially in terms of MAE. Even so, COR_FR demonstrates that fuzzy ranks are most useful tools for enhancing performance of COR.

For the experiments on COS and MSD, MAE results show almost ignorable differences among the measures except for SM. It is observed that SM is extremely poor in terms of both MAE and F1, compared to all the other measures. SM is originally intended to improve performance of MSD, but its objective is not achieved as shown in the results related to MSD with this sparse dataset. Notice that the proposed measure yields different behavior from that with MovieLens. That is, it is slightly defeated by the entropy-weighted measure, especially when combined with COS. As BookCrossing is sparser and has longer rating range, the deviation of entropy among items should be higher, which is believed to be better reflecting the global rating behavior of users on items on similarity than fuzzy ranks.

3) Jester Dataset: The last experiments are conducted with Jester dataset. In Fig. 4, in case of COR-associated measures, different outcomes are noted, compared with COS- and MSD-associated measures. The main difference comes from the results of COR_E. This measure performs much poorer than the base COR measure, even if the entropy is incorporated. As reported in the literature, COR is reliable with enough data, which is the case with Jester dataset. Thus, combining entropy proves to negatively affect similarity computation of COR. However, this does not apply to the other conventional measures as seen in the figure, where there shows ignorable

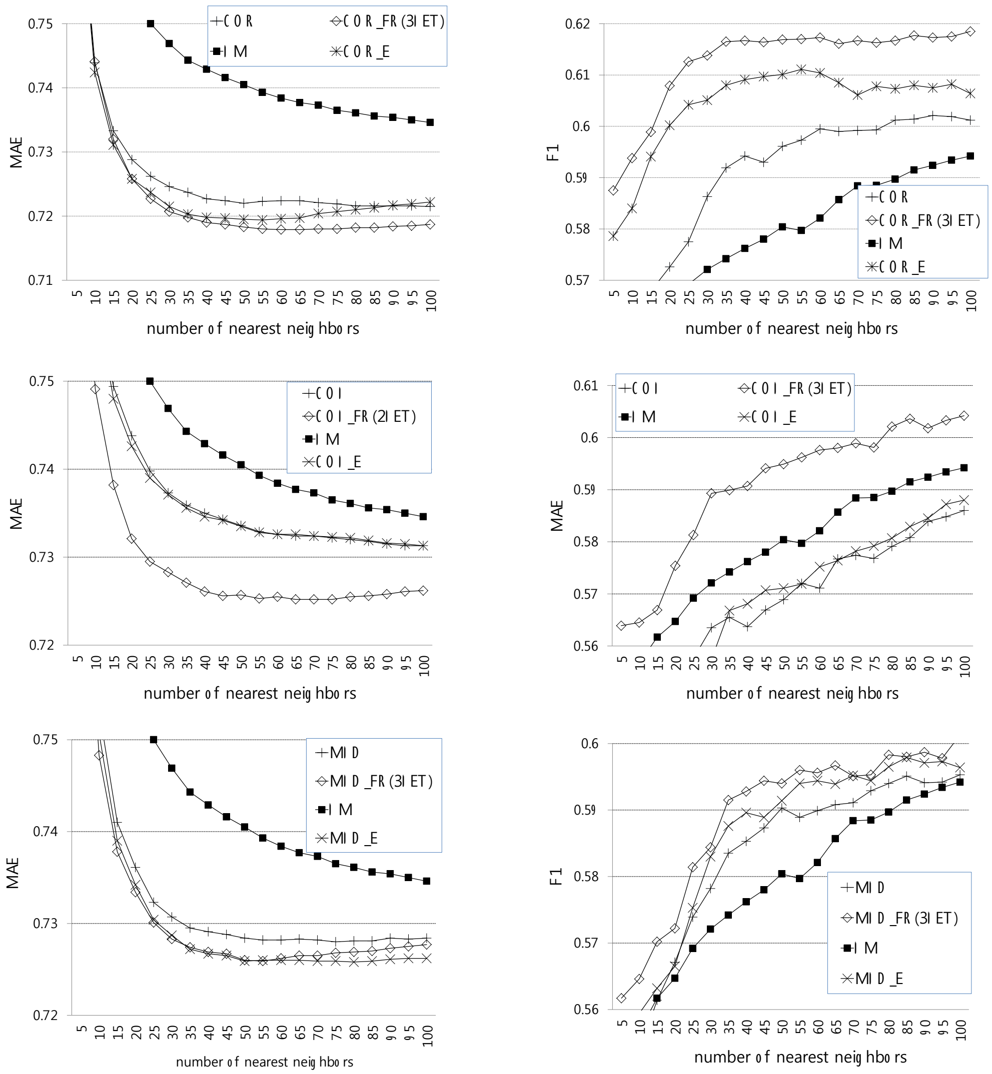

Fig. 2. Performance comparison with MovieLens dataset

differences of results between the measures except SM.

Looking into the results of COR_FR, its achievement is slight compared to COR, as COR exhibits very competitive performance in this dense dataset. This situation also occurs for COS and MSD, indicating that combination of additional information with the traditional measures appears not so useful for a dense dataset with a long rating range. However, incorporation of fuzzy ranks into COR as proposed by COR_FR allows still further improvement in terms of both MAE and F1. Although incorporation of weight is also made by SM where the weight is so-called singularity, SM turns out to perform worst. The reason may be mainly because of the dense dataset where high or low ratings should not be so singular as in a sparser dataset.

V. Conclusion

This study proposed a new similarity measure for user-based collaborative filtering systems. It is intended to reflect all of the users' rating behavior on each common item onto similarity between two users. Such behavior is measured by computing fuzzy ranks of user ratings on an item and incorporating them into traditional similarity measures to develop a new measure.

We investigated performance of the proposed measure in depth using datasets with various characteristics. The experiments result in that the proposed mostly outperforms not only the conventional similarity measures but also the state-of-the-art similarity measures which also attempt to achieve the same objective as ours, i.e., reflecting the users' rating behavior onto similarity. The improvements are found especially significant in all the datasets when the proposed measure incorporates Pearson correlation. The experimental findings reveal that the proposed strategy is promising in comparison to the weight-based similarity measures, although further study should be done to discover proper combinations of previous similarity measures and the weights.

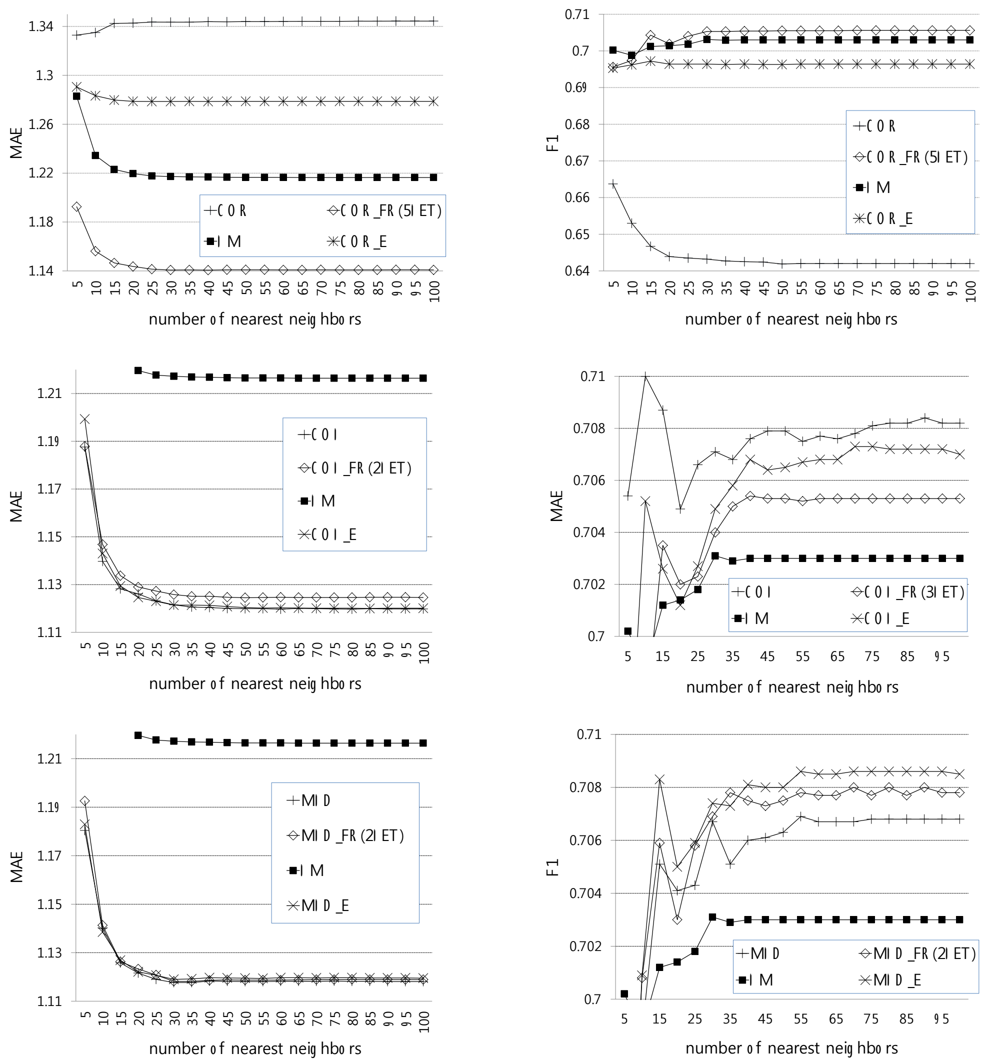

Fig. 3. Performance comparison with BookCrossing dataset

References

[1] M. Aamir and M. Bhusry, "Recommendation system: State of the art approach," *International Journal of Computer Applications*, vol. 120, no. 12, pp. 25–32, June 2015.

[2] F. Cacheda, V. Carneiro, D. Fernandez, and V. Formoso, "Comparison of collaborative filtering algorithms: limitations of current techniques and proposals for scalable, high-performance recommender systems," *ACM Trans Web*, vol. 5, no. 1, pp. 1–33, 2011.

[3] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009, 2009.

[4] M. Y. H. Al-Shamri and N. H. Al-Ashwal, "Fuzzy-weighted similarity measures for memory-based collaborative recommender systems," *Journal of Intelligent Learning Systems and Applications*, vol. 6, pp. 1–10, 2014.

[5] J. Bobadilla, F. Ortega, and A. Hernando, "A collaborative filtering similarity measure based on singularities," *Information Processing and Management*, vol. 48, no. 2, pp. 204–217, 2012.

[6] M. Jamali and M. Ester, "Trustwalker: A random walk model for combining trust-based and item-based recommendation," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2009, pp. 397–406.

[7] J.-H. K. Hyeong-Joon Kwon, Tae-Hoon Lee and K.-S. Hong, "Improving prediction accuracy using entropy weighting in collaborative filtering," in *Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing*, 2009, pp. 40–45.

[8] S. Lee, "Entropy-weighted similarity measures for collaborative recommender systems," in *Int'l Conf. Mathematical Methods & Computational Techniques in Science & Engineering*, Feb. 2018.

[9] K. G. Saranya, G. S. Sadasivam, and M. Chandralekha, "Performance comparison of different similarity measures for collaborative filtering technique," *Indian Journal of Science and Technology*, vol. 9, no. 29, pp. 1–8, Aug. 2016.

[10] L. H. Son, "Hu-fcf: A hybrid user-based fuzzy collaborative filtering method in recommender systems," *Expert Systems with Applications*, vol. 41, pp. 6861–6870, 2014.

[11] P. Resnick, N. Lakovou, M. Sushak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proc. the ACM conference on Computer supported cooperative work*. ACM Press, 1994, pp. 175–186.

[12] P. Xia, L. Zhang, and F. Li, "Learning similarity with cosine similarity ensemble," *Information Sciences*, vol. 307, pp. 39–52, 2015.

[13] C. C. Chen, Y.-H. Wan, M.-C. Chung, and Y.-C. Sun, "An effective recommendation method for cold start new users using trust and distrust networks," *Information Sciences*, vol. 224, pp. 19–36, 2013.

[14] J. L. Herlocker, J. A. Konstan, and J. T. Riedl, "An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms," *Information Retrieval*, vol. 5, no. 4, pp. 287–310, 2002.

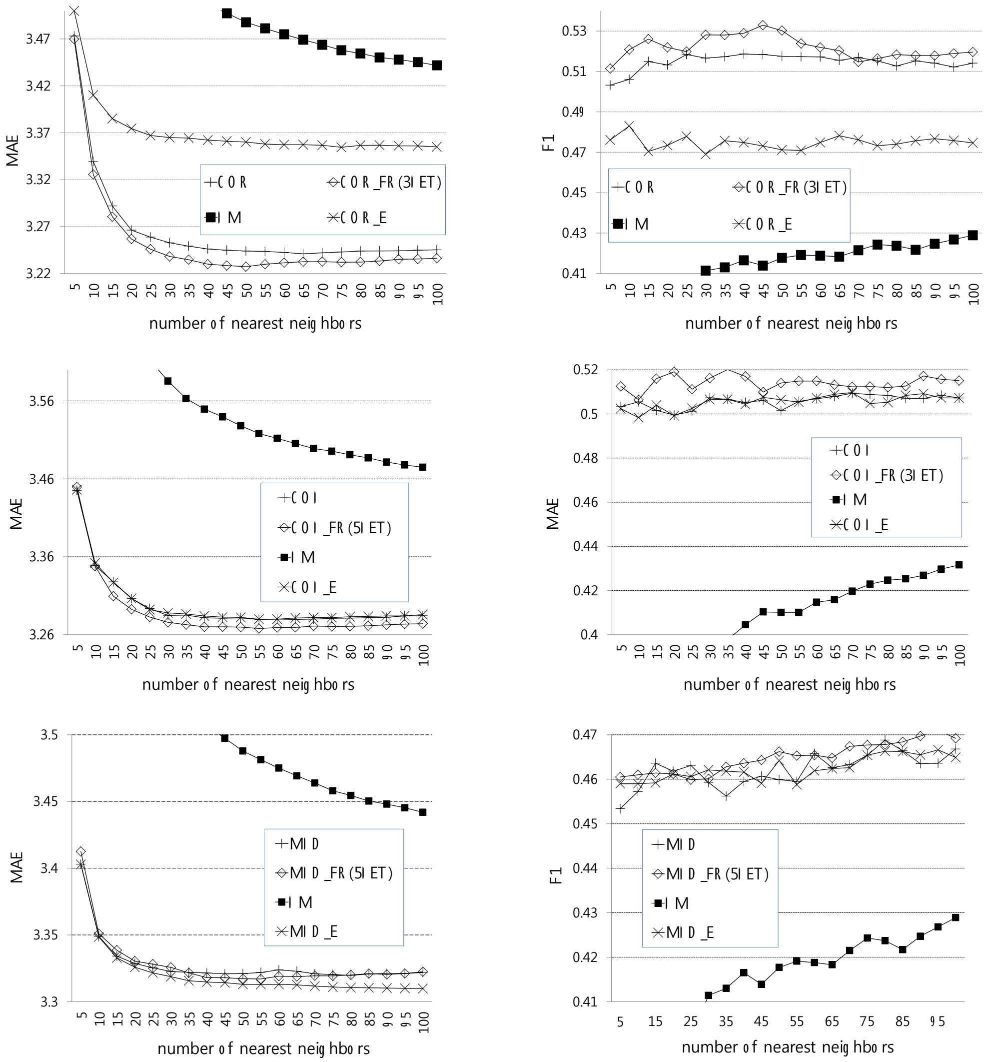

Fig. 4. Performance comparison with Jester dataset

[15] G. Koutrica, B. Bercovitz, and H. Garcia, "FlexRecs: expressing and combining flexible recommendations," in *SIGMOD Conference*, 2009, pp. 745–758.

[16] S. Lee, "Improving jaccard index for measuring similarity in collaborative filtering," *Lecture Notes in Electrical Engineering*, vol. 424, pp. 799–806, 2017.

[17] H. Liu, Z. Hu, A. Mian, H. Tian, and X. Zhu, "A new user similarity model to improve the accuracy of collaborative filtering," *Knowledge-Based Systems*, vol. 56, pp. 156–166, Jan. 2014.

[18] L. Ren, J. Gu, and W. Xia, "A weighted similarity-boosted collaborative filtering approach," *Energy Procedia*, vol. 13, pp. 9060–9067, 2011.

[19] C. E. Shannon, "Prediction and entropy of printed english," *The Bell System Technical Journal*, vol. 30, pp. 50–64, 1951.

[20] H. Chandrashekhar and B. Bhasker, "Personalized recommender system using entropy based collaborative filtering technique," *Journal of Electronic Commerce Research*, vol. 12, no. 3, pp. 214–237, 2011.

[21] M. Li and K. Zheng, "A collaborative filtering algorithm combined with user habits for rating," in *International Conference on Logistics Engineering, Management and Computer Science*, 2015, pp. 1279–1282.

[22] C. H. Piao, J. Zhao, and L. J. Zheng, "Research on entropy-based collaborative filtering algorithm and personalized recommendation in e-commerce," *Service Oriented Computing and Applications*, vol. 3, no. 2, pp. 147–157, 2009.

[23] W. Wang, G. Zhang, and J. Lu, "Collaborative filtering with entropy-driven user similarity in recommender systems," *International Journal of Intelligent Systems*, vol. 30, no. 8, pp. 854–870, Aug. 2015.