Sādhanā (2018) 43:202
https://doi.org/10.1007/s12046-018-0970-3

A novel similarity measure towards effective recommendation using Matusita coefficient for Collaborative Filtering in a sparse dataset

C SELVI* and E SIVASANKAR

Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli 620015, India
e-mail: selvichandran.it@gmail.com; sivasankarelango@gmail.com

MS received 26 July 2017; revised 20 January 2018; accepted 5 June 2018; published online 3 November 2018

Abstract. Collaborative Filtering (CF) is a prominent approach to ensure personalized recommendations to active online users. An efficient CF is the memory-based strategy that finds nearest neighbours to an active user using conventional similarity measures. Most such measures deal with a co-rated item rated by a pair of users and hence they are not appropriate to provide an effective recommendation to a sparse dataset having less co-rated items. This study proposes a novel similarity measure, Matusita coefficient in CF (MCF), which considers all ratings given by a user to estimate nearest neighbours. MCF considers local and global rating information provided by users on different rating scales. The performance of the proposed measure is examined and checked by comparing it to conventional measures using popular benchmark datasets like MovieLens and Netflix. The recommendation results demonstrate that the proposed measure outperforms conventional similarity measures on various performance metrics like Mean Absolute Error, Root Mean Squared Error, accuracy, precision, recall and coverage.

Keywords. Collaborative Filtering (CF); memory-based CF; similarity measure; Recommender System (RS); co-rated items; no co-rated items; sparse dataset.

1. Introduction

The progress of technology in electronic devices, like cell-phones, tablets, personal computers and other intelligent frameworks, helps people receive services/items from electronic business sites (i.e., Amazon, Flipkart, eBay, etc.) and social networking sites (i.e., Facebook, LinkedIn, Twitter, etc.). People find it difficult to arrive at correct decisions since the WWW is flooded with an enormous amount of varied information. To attain satisfactory search results, people are forced to spend more time and energy. Generally, purchase history and search results of online users, recorded by these sites, help examine and estimate decisions efficiently. The personalized Recommender System (RS) tool summaries the user's history and advices a few amazing services that users may be interested in future. RS doubles the income of many electronic business services, which cover movies [1], products [2], books [3], music [4], videos [5], social networks [6], tourism [7], etc. Filtering algorithms are the backbone of RS. The types of filtering algorithm used by RS includes demographic filtering, content-based filtering and Collaborative Filtering (CF). Demography-based RS works on the premise that users with common personal characteristics will also experience common choices [8]. Content-based RS recommends a new item by matching its characteristics with the characteristics of items already bought by the user [9, 10].

CF is an extensively used RS filtering mechanism, which provides recommendations by analysing the rating information of items or the users [11–13]. Any CF-based model is domain independent and provides improved accuracy over content-based RS. CF is classified into memory-based and model-based methods [14]. In memory-based CF, initially, similarities between an active user and other users are estimated, based on which, closest similar users are identified. Finally, the target item values are predicted using the nearest identified user; in turn, recommendations are provided accordingly. Memory-based CF techniques are further divided into user-based and item-based methods. The user-based method predicts the value of a new item concerning the nearest neighbours of an active user. In an item-based method, prediction is made by connecting the nearest neighbour item to the item of an active user.

In memory-based CF, the user–item rating matrix has to be loaded into memory to estimate the similarity between the active user and other users at every attempt. It results in a scalability issue and consumes additional computational time. To offset this, model-based CF is preferred by researchers as it uses supervised or unsupervised approaches to learn a model from the training user–item rating matrix. The learned model is also used to predict the rating

*For correspondence

of the active user's item. Here, once the model is developed, the user–item matrix is not needed further. Thus, prediction is possible offline and works faster even when the numbers of users and items increase. Memory-based method needs only one parameter, which is the number of nearest neighbours (k), whereas model-based approach requires learning and regularization parameters. Also, memory-based CF approach ensures better performance compared with a model-based approach regarding accuracy [15]. Hence, in this study, memory-based CF approach is considered for providing effective recommendations to online users.

Memory-based CF uses conventional similarity measures to discover active users nearest neighbours or nearest item to the item of an active user. Conventional measures calculate similarity between a pair of users or pair of items comprising Cosine, Pearson and its variants [16]. These measures work on co-rated values, i.e., item rated by a pair of users or a user rates both items. It results in poor prediction if the dataset has fewer co-rated items. Hence, conventional memory-based CF method and its variants are not fit for sparse datasets [17]. Hence, a novel similarity measure is proposed to offset drawbacks of the measures mentioned earlier as it improves RS performance in a sparse dataset.

The contribution of the proposed work is as follows:

- A novel similarity measure is proposed to improve RS performance in a sparse dataset.
- This measure works on
  - * Co-rated and no co-rated items.
  - * Datasets with less co-rated items.
  - * Covers both local and global rating information provided by users.

The remainder of this study is organized as follows. Conventional similarity measures and their variants are discussed in section 2. The proposed similarity measure is described in section 3. The details of the proposed prediction methods are demonstrated in section 4. Section 5 illustrates details of the performance measures and dataset used to analyse the RS; the proposed measure's results are analysed and compared to those of conventional similarity measures. Finally, section 6 concludes the study.

2. Background and related work

Over decades, CF algorithms became a progressive research area that filtered information or recommended services to online users based on their browsing history. Most CF algorithms use a memory-based strategy. This method finds similarity among users and then selects higher similarity users as neighbours of an active user. Then, based on its neighbours, the CF recommends items to active users. Among two types (user-based and item-based) of memory-based methods, user-based method is recognized as a performance improvement technique regarding accuracy. Hence, this study proposes a novel similarity measure for user-based CF. Let $OU = ou_1, ou_2, \ldots, ou_l$ and $I = i_1, i_2, \ldots, i_m$ be a set of online users and items, respectively. The online user–item matrix is represented as $R = (r_{xy})_{l\times m}, x = 1, 2, \ldots, l$ and $y = 1, 2, \ldots, m$. Rating values are expressed in a 1–5 scale. Rating value $r_{ou,i} = \phi$ denotes that the online user has not bought or rated item i. Some popular memory-based CF similarity measures are discussed in this section.

Conventional similarity measures used in memory-based CF are given in table 1. Pearson Correlation Coefficient (PCC), a conventional and standard measure in user-based CF, considers only the value of co-rated items to find similarity [18]. PCC performance is reduced when the number of co-rated items is less and hence it is not suitable for a sparse dataset. PCC is improved as Constrained PCC (CPCC) to address this shortcoming as it uses the median of the rating scale rather than considering user's average rating value as in PCC. Similarity value among online users is highly acceptable when online users rate more similar items [18]. PCC is further improved as Weighted PCC (WPCC) [19] and Sigmoid PCC (SPCC) [20]. PCC variants still suffer from lesser co-rated items and hence fail to find accurate similarity value among users. Also, these measures do not consider the proportion of users with common rating (global) information. Hence, PCC and its variants are not suitable for sparse datasets.

A popular, item-based conventional similarity measure is COSine (COS) similarity, which does not provide preferences for different ranges of a rating scale provided by users to find similarity among a pair of items [21]. This issue is offset by Adjusted COS (ACOS) measure, which subtracts the item's mean value from the user's rating value on the corresponding item [22]. The Spearman Rank Correlation (SRC) is another measure, which considers the rank of the items instead of actual ratings as in PCC. Rankings are made from higher to the lower rated item, i.e., highest rating item ranked as 1 and vice versa [23].

Mean Squared Difference (MSD) considers absolute rating differences among users to compute similarity [18]. It fails to consider the proportion of common ratings, and works only on co-rated items. Hence, the accuracy of similarity value estimated by MSD depends on the number of co-rated items. Thus, it is not a suitable measure for sparse datasets. Unlike MSD, Jaccard measure [24] calculates similarity value using the number of common ratings among online users rather than absolute rating values, i.e., it combines global rating information about users. Similarity value between online users by Jaccard measure is high when there are more common ratings and vice versa. The drawbacks of MSD and Jaccard are addressed by the Jaccard Mean Squared Difference (JMSD) measure, which combines MSD and Jaccard measures to give importance to both absolute rating difference and a proportion of common

Table 1. Conventional similarity measures used in memory-based CF.

Sl. no.	Name of the measure	Representation	Expression
1	Pearson Correlation Coefficient (PCC)	$sim(ou_x, ou_y)^{PCC}$	$= \frac{\sum_{d=1}^{n_{co}} (r_{ou_x,i_d} - \bar{r}_{ou_x}) \times (r_{ou_y,i_d} - \bar{r}_{ou_y})}{\sqrt{\sum_{d=1}^{n_{co}} (r_{ou_x,i_d} - \bar{r}_{ou_x})^2} \times \sqrt{\sum_{d=1}^{n_{co}} (r_{ou_y,i_d} - \bar{r}_{ou_y})^2}}$ where $r_{ou,i}$ is the rating of online item i by the online user ou, $\bar{r}_{ou}$ is the average rating of ou for all co-rated items and n_{co} is the number of co-rated items by ous
2	Constrained PCC (CPCC)	$sim(ou_x, ou_y)^{CPCC}$	$= \frac{\sum_{d=1}^{n_{co}} (r_{ou_x,i_d} - r_{med})(r_{ou_y,i_d} - r_{med})}{\sqrt{\sum_{d=1}^{n_{co}} (r_{ou_x,i_d} - r_{med})^2} \sqrt{\sum_{d=1}^{n_{co}} (r_{ou_y,i_d} - r_{med})^2}}$ where r_{med} is the median of value in the rating scale. For example, r_{med} value is 3 in the 1–5 scale ratings
3	Weighted PCC (WPCC)	$sim(ou_x, ou_y)^{WPCC}$	$= \begin{cases} sim(ou_x, ou_y)^{PCC} \times \frac{n_{co}}{O} & n_{co} \leq O \\ sim(ou_x, ou_y)^{PCC} & \text{otherwise} \end{cases}$ where O is an experimental value and set to 50 as in [19]
4	Sigmoid PCC (SPCC)	$sim(ou_x, ou_y)^{SPCC}$	$= sim(ou_x, ou_y)^{PCC} \times \frac{1}{1+exp(\frac{-n_{co}}{2})}$
5	COSine (COS)	$sim(ou_x, ou_y)^{COS}$	$= \frac{\vec{r}_{ou_x} \cdot \vec{r}_{ou_y}}{\|\vec{r}_{ou_x}\| \times \|\vec{r}_{ou_y}\|}$ where $\vec{r}_{ou_x}$ and $\vec{r}_{ou_y}$ are rating vector representation, respectively, of ou_x and ou_y
6	Adjusted COS (ACOS)	$sim(i_x, i_y)^{ACOS}$	$= \frac{\sum_{e=1}^{m_{co}} (r_{ou_e,i_x} - \bar{r}_{i_x})(r_{ou_e,i_y} - \bar{r}_{i_y})}{\sqrt{\sum_{e=1}^{m_{co}} (r_{ou_e,i_x} - \bar{r}_{i_x})^2} \times \sqrt{\sum_{e=1}^{m_{co}} (r_{ou_e,i_y} - \bar{r}_{i_y})^2}}$ where m_{co} represents the number of users who rated both the items and $\bar{r}_i$ is the average rating of an item
7	Mean Squared Difference (MSD)	$sim(ou_x, ou_y)^{MSD}$	$= 1 - \frac{\sum_{d=1}^{n_{co}} (r_{ou_x,i_d} - r_{ou_y,i_d})^2}{n_{co}}$
8	Jaccard	$sim(ou_x, ou_y)^{Jaccard}$	$= \frac{\|I_{ou_x} \cap I_{ou_y}\|}{\|I_{ou_x} \cup I_{ou_y}\|}$ where I_{ou} is the set of items by ou
9	Jaccard MSD (JMSD)	$sim(ou_x, ou_y)^{JMSD}$	$= sim(ou_x, ou_y)^{Jaccard} \times sim(ou_x, ou_y)^{MSD}$
10	Proximity–impact–popularity (*PIP*)	$PIP(r_{ou_x,i}, r_{ou_y,i})$	$= proximity(r_{ou_x,i}, r_{ou_y,i}) \times impact(r_{ou_x,i}, r_{ou_y,i}) \times popularity(r_{ou_x,i}, r_{ou_y,i})$
11	Proximity–significance–singularity (*PSS*)	$PSS(r_{ou_x,i}, r_{ou_y,i})$	$= proximity(r_{ou_x,i}, r_{ou_y,i}) \times significance(r_{ou_x,i}, r_{ou_y,i}) \times singularity(r_{ou_x,i}, r_{ou_y,i})$
12	New Heuristic Similarity Measure (NHSM)	$sim(r_{ou_x,i}, r_{ou_y,i})^{NHSM}$	$= sim(r_{ou_x,i}, r_{ou_y,i})^{JPSS} \times sim(r_{ou_x,i}, r_{ou_y,i})^{URP}$

ratings among a pair of users [25]. Therefore, it covers both local and global rating information of users to some extent. However, it also fails to use no co-rated item values, which are more in sparse datasets.

PIP (proximity–impact–popularity) measure consolidates three factors, namely proximity, impact and popularity, for a pair of online users [26]. Proximity deals with the penalty given to a pair of ratings based on an agreement value. The true and false values of agreements are determined by matching the median rating scale with the pair of user ratings. If the agreement is false, then high penalty is given and vice versa. Impact describes how strongly an online item is preferred or not preferred by the users. Popularity captures a user's global rating information and provides additional preference to an item whose value is far from that item's mean rating value. For the new user, the *PIP* measure outperforms some conventional memory-based similarity measures [26]. *PIP* also suffers from a few co-rated issues and does not suit for sparse datasets.

PSS (proximity–significance–singularity) was introduced to improve RS result compared with *PIP* [26]. Initially, proximity is calculated as *PIP*, which measures the distance between a pair of ratings. Then, a significance factor states that the ratings are significant only when item ratings are apart from the median of a rating scale. Finally, the singularity factor calculates how a pair of ratings differs from another rating [26]. As in *PIP*, *PSS* also suffers from a few co-rated item ratings. Many similarity measures were proposed by modifying and hybridizing conventional similarity measures. The issues faced by *PIP* and *PSS* were reduced by the New Heuristic Similarity Measure (NHSM), which is a combination of the Jaccard PSS (JPSS) and User Rating Preferences (URP) measures [26]. JPSS

consolidates the merits of *PSS* and the modified Jaccard, which uses a significant portion of common ratings. URP measure estimates the normalized rating value of user through mean and variance as URP depends on the ranges of rating scale value [26].

As all these similarity measures fail to consider no co-rated items, they are not suitable when a dataset contains less co-rated items or is sparse. However, the popular Bhattacharyya similarity measure was employed on no co-rated items [27]. These rating values are considered only when the similarity value among a pair of no co-rated items is maximum. Otherwise the values of no co-rated items are neglected. It shows that this measure also fails to consider all the values of no co-rated items. Hence, it is concluded that all the conventional measures are unsuitable for finding similarity among a pair of users if the dataset is sparse. In this study, a novel similarity measure, Matusita coefficient in CF (*MCF*), is proposed, which uses both co-rated and no co-rated item values effectively while eliminating issues faced by conventional measures.

3. Proposed measures

The essential step of the memory-based CF approach is to find the nearest neighbours of an active user using a suitable similarity measure. To maximize the accuracy of a user's similarity value in a sparse dataset, a novel similarity measure *MCF* is proposed here. It utilizes both global and local rating information provided by online users. The global similarity value between a pair of items is estimated using Matusita coefficient (*MC*). Local similarity among users ratings is identified using correlation-based (LOC_{cor}) and median-based (LOC_{med}) measures [28]. Section 3.1 explains the Matusita measure and section 3.2. discusses the local similarity measure and its types.

3.1 *Matusita measure*

The Matusita measure is broadly utilized in various applications like image processing [29], signal and pattern recognition [30], etc. This measure is introduced to compute the significant distance between two probability distributions [31]. This study modifies the Matusita measure to work with RS and is used to eliminate data sparsity issues. Let $p_{i_u}(x)$ and $p_{i_v}(x)$ be the probability density distribution, respectively, of items i_u and i_v in a continuous domain. The *MC* for the continuous domain is characterized in Eq. (1):

$$MC(p_{i_u}, p_{i_v}) = \sqrt{2 - 2\int \sqrt{p_{i_u}(x)p_{i_v}(x)}dx}. \tag{1}$$

For the discrete domain X, *MC* is represented in Eq (2):

$$MC(p_{i_u}, p_{i_v}) = \sqrt{2 - 2\sum_{x\in X} \sqrt{p_{i_u}(x)p_{i_v}(x)}}. \tag{2}$$

The probability density values of $p_{i_u}(x)$ and $p_{i_v}(x)$ are estimated from the given rating values. Let $\hat{p}_{i_u}$ and $\hat{p}_{i_v}$ be the estimated probability density distribution, respectively, of items i_u and i_v. *MC* similarity between items i_u and i_v is calculated as in Eq. (3):

$$\begin{aligned} sim(i_u, i_v)^{MC} &= MC(i_u, i_v) = MC(\hat{p}_{i_u}, \hat{p}_{i_v}) \\ &= \sqrt{2 - 2\sum_{s=1}^{t} \sqrt{\hat{p}_{i_{us}}\hat{p}_{i_{vs}}}} \end{aligned} \tag{3}$$

where t denotes the total number of distinct ratings in the considered rating scale value (s) and s can take value from the set {1, 2, 3, 4, 5}; $\hat{p}_{i_{us}} = \frac{\#s}{\#i_u}$ and $\hat{p}_{i_{vs}} = \frac{\#s}{\#i_v}$ are the estimated probability density distribution, respectively, of items i_u and i_v with respect to rating value s, where $\#i_u$ represents the number of online users who rated item i_u and $\#s$ represents the total number of online users who rated item i_u with rating values s. Here, $\sum_{s=1}^{t} \hat{p}_{i_{us}} = \sum_{s=1}^{t} \hat{p}_{i_{vs}} = 1$ and $sim(i_u, i_v)^{MC}$ lies between 0 and 1.

$sim(i_u, i_v)^{MC}$ is illustrated here with a basic example. Let $i_u = (1, 0, 2, 0, 3, 0, 4, 0, 5, 0)^T$ and $i_v = (0, 2, 0, 3, 0, 4, 0, 3, 0, 5)^T$ be the rating vectors of items. The rating values are in 1–5 scale. The *MC* similarity value is computed utilizing Eq. (3):

$$\begin{aligned} sim(i_u, i_v)^{MC} &= \sqrt{2 - 2\sum_{s=1}^{5} \sqrt{\hat{p}_{i_{us}}\hat{p}_{i_{vs}}}} \\ &= \sqrt{2 - 2\left(\sqrt{\left(\frac{1}{5}\right)\left(\frac{0}{5}\right)} + \sqrt{\left(\frac{1}{5}\right)\left(\frac{1}{5}\right)} + \sqrt{\left(\frac{1}{5}\right)\left(\frac{2}{5}\right)} + \sqrt{\left(\frac{1}{5}\right)\left(\frac{1}{5}\right)} + \sqrt{\left(\frac{1}{5}\right)\left(\frac{1}{5}\right)}\right)} \\ &= 0.4841. \end{aligned}$$

Unlike conventional measures, *MC* similarity measure works even when there is no co-rated item.

3.2 *Local similarity*

Local similarity performs an important role in the proposed *MCF* measure by providing local information on the ratings of the corresponding users on a pair of items. A local similarity value between two ratings may be positive or negative. Positive similarity value indicates that the users rating on item i_u and i_v are highly similar and negative similarity value indicates dissimilarity among them. A local similarity value between ratings on a pair of items is estimated utilizing two techniques: LOC_{cor} and LOC_{med}. The LOC_{cor} considers user's average rating value as the rating scale reference. Let r_{ou_x,i_u} and r_{ou_y,i_v} be the rating values of items i_u and i_v given by users ou_x and ou_y; then the function for LOC_{cor} is given in Eq (4):

$$LOC_{cor}(r_{ou_x,i_u}, r_{ou_y,i_v}) = \frac{(r_{ou_x,i_u} - \overline{r}_{ou_x})(r_{ou_y,i_v} - \overline{r}_{ou_y})}{\sigma_{ou_x}\sigma_{ou_y}} \quad (4)$$

where $\overline{r}_{ou_x}$ and $\overline{r}_{ou_y}$ are the mean value of the ratings given by ou_x and ou_y, respectively; σ_{ou_x} and σ_{ou_y} are the standard deviation of ratings given by online users ou_x and ou_y, respectively. The LOC_{med} technique utilizes median of the rating scale as a rating scale reference. The $LOC_{med}(r_{ou_x,i_u}, r_{ou_y,i_v})$ appears in Eq. (5):

$$LOC_{med}(r_{ou_x,i_u}, r_{ou_y,i_v}) = \frac{(r_{ou_x,i_u} - r_{med})(r_{ou_y,i_v} - r_{med})}{\sqrt{\sum_{i_u \in I_{ou_x}} (r_{ou_x,i_u} - r_{med})^2} \sqrt{\sum_{i_u \in I_{ou_y}} (r_{ou_y,i_v} - r_{med})^2}} \quad (5)$$

Parameter r_{med} is the median of the rating scale and I_{ou_x} and I_{ou_y} are a set of items rated by ou_x and ou_y, respectively.

3.3 *MCF: novel similarity measure for memory-based CF in sparse dataset*

The proposed novel similarity measure *MCF* uses the benefits of *MC* and local similarity. *MCF* considers the ratings on common items and the numerical rating information made by a pair of users, which maximizes the accuracy of the proposed similarity measure. The $I_{ou_x} \cap I_{ou_y} = \phi$ reveals that there are no co-rated items between users ou_x and ou_y. *MCF* similarity measure between users ou_x and ou_y is a function of *MC* similarity between a pair of items and local similarity between ratings on the pair of items of the corresponding users. The function of the *MCF* measure is shown in Eq. (6):

$$sim(r_{ou_x}, r_{ou_y})^{MCF} = \sum_{i_u \in I_{ou_x}} \sum_{i_v \in I_{ouy}} sim(i_u, i_v)^{MC} \times LOC(r_{ou_x,i_u}, r_{ou_y,i_v}). \quad (6)$$

The first parameter $sim(i_u, i_v)^{MC}$ considers global rating information between a pair of items. *MC* similarity value can be measured between a pair of items even when there is no common rated user among them. However, *MC* measure does not consider the actual rating information. The second parameter $LOC(r_{ou_x,i_u}, r_{ou_y,i_v})$ invalidates the shortcoming of *MC* by considering all rating information of items i_u and i_v. The proposed *MCF* measure is used in two situations. The first is when the global similarity value between items i_u and i_v is closer to 0; in this case, $sim(i_u, i_v)^{MC}$ decreases local similarity value between ratings of the corresponding users on the pair of items. In the second situation, if the global similarity value $sim(i_u, i_v)^{MC}$ between items is nearer to 1, the *MCF* increases preference for the local similarity value.

In two cases, $sim(i_u, i_v)^{MC} = 1$ and $sim(i_u, i_v)^{MC} = 0$, *MCF* fails to give importance to the actual rating of items (i.e.,) LOC_{cor} and LOC_{med}. To avoid this drawback and to use absolute rating values, the proposed *MCF* measure is combined with the similarity measure JMSD. JMSD considers the absolute rating information of users and the proportion of common ratings. That is, JMSD utilizes local and global rating information to some degree. Hence, the proposed *MCF* measure is re-defined and shown in Eq. (7):

$$sim(r_{ou_x,i}, r_{ou_y,i})^{MCF} = sim(r_{ou_x}, r_{ou_y})^{JMSD} + \sum_{i_u \in I_{ou_x}} \sum_{i_v \in I_{ou_y}} sim(i_u, i_v)^{MC} \times LOC(r_{ou_x,i_u}, r_{ou_y,i_v}) \quad (7)$$

3.4 *Contribution of the proposed similarity measure*

This section discusses the significance of the proposed *MCF* similarity measure over existing measures.

- Parameters that find similarity in conventional measures consider only actual rating information of co-rated items and do not consider the values of no co-rated items. Since the Bhattacharyya measure considers similarity among a pair of no co-rated items, it neglects no co-rated items when similarity value among them is minimum. Thus, this measure fails to consider all no co-rated items' values.
- In a sparse dataset, rating values by individual users are minimum and finding co-rated items is inadequate. Hence, the proposed *MCF* measure deals with a few or no co-rated items. It uses local and global rating information of users.

- Local rating information obtained using LOC_{cor} and LOC_{med} similarity measures, which calculates similarity among absolute ratings of corresponding users on a pair of items.
- Global rating information obtained using *MC* similarity measure, which calculates similarity between a pair of items.
- To use the probability of common rated items and all rating information efficiently, the proposed *MCF* measure adopts the benefits of the JMSD measure, thereby ensuring that the similarity value of *MCF* is not null and insignificant. It shows that *MCF* is independent of the number of co-rated items and highly preferable for sparse datasets.

4. Prediction

The item value is predicted based on the k-nearest neighbours to an active user [32]. Initially, the similarity values are normalized between 0 and 1. Then, the best k-nearest neighbours are chosen based on an ascending order of similarity values. As the proposed measure considers 0 for a high similarity value, and 1 for low similarity value [33], two types of prediction strategies (Type 1 and Type 2) are proposed based on the nearest user who rated/did not rate an active user's item. Initially, prediction starts with Type 1; if it fails then Type 2 prediction is carried out [34]. If these two types of predictions do not succeed, then the default prediction method is used (see Eq. (14)).

4.1 *Type 1 prediction*

In Type 1, active user's item rating is predicted when the neighbours of an active user also rate the same item. The prediction function for average (Avg) [32] is presented in Eq. (8). The proposed prediction functions for *MCF* are Weighted Sum (WS) and Adjusted Weight Aggregation(AWA), which are given in Eqs. (9) and (10), respectively:

$$p^{Avg}_{ou_x,i_u} = \frac{1}{\#E_{ou_x,i_u}} \sum_{n \in E_{ou_x,i_u}} r_{ou_n,i_u} \Leftrightarrow E_{ou_x,i_u} \neq \phi \tag{8}$$

$$p^{WS}_{ou_x,i_u} = \mu_{ou_x,i_u} \sum_{n \in E_{ou_x,i_u}} (1 - sim(ou_x, ou_n)) \times r_{ou_n,i_u} \Leftrightarrow E_{ou_x,i_u} \neq \phi \tag{9}$$

$$p^{AWA}_{ou_x,i_u} = \overline{r}_{ou_x} + \mu_{ou_x,i_u} \sum_{n \in E_{ou_x,i_u}} (1 - sim(ou_x, ou_n)) \times (r_{ou_n,i_u} - \overline{r}_{ou_n}) \Leftrightarrow E_{ou_x,i_u} \neq \phi \tag{10}$$

where p_{ou_x,i_u} is rating prediction on item i_u for active user ou_x, k_{ou} is k-nearest OU to ou_x, n is any ou in k_{ou}, ou_n is n^{th} OU in k_{ou}, r_{ou_n,i_u} is rating of n^{th} k_{ou} on item i_u and $sim(ou_x, ou_n)$ is similarity between ou_x and n^{th} k_{ou} to ou_x, $\mu_{ou_x,i_u} = \frac{1}{\sum_{n \in E_{ou_x,i_u}} (1 - sim(ou_x, ou_n))} \Leftrightarrow E_{ou_x,i_u} \neq \phi$ is a normalization factor, $E_{ou_x,i_u} = \{n \in k_{ou} \mid \exists r_{ou_n,i_u} \neq \phi\}$ is a k_{ou} who rated the item i_u and $\#E_{ou_x,i_u}$ is a count of k_{ou} who rated the item i_u.

4.2 *Type 2 prediction*

The average rating value of an item by every user is taken into consideration to fill the neighbour item value related to an active item [34]. Similar to Type 1, Type 2 also utilizes the prediction functions Avg, proposed WS and AWA. The Type 2 Avg prediction function is given in Eq. (11). The proposed Type 2 functions WS and AWA are given in Eqs. (12) and (13), respectively:

$$p^{Avg}_{ou_x,i_u} = \frac{1}{\#D_{ou_x,i_u}} \sum_{n \in D_{ou_x,i_u}} r_{ou_n,i_u} \Leftrightarrow E_{ou_x,i_u} = \phi \wedge D_{ou_x,i_u} \neq \phi \tag{11}$$

$$p^{WS}_{ou_x,i_u} = \mu_{ou_x,i_u} \sum_{n \in D_{ou_x,i_u}} (1 - sim(ou_x, ou_n)) r_{ou_n,i_u} \Leftrightarrow E_{ou_x,i_u} = \phi \wedge D_{ou_x,i_u} \neq \phi \tag{12}$$

$$p^{AWA}_{ou_x,i_u} = \overline{r}_{ou_x} + \mu_{ou_x,i_u} \sum_{n \in D_{ou_x,i_u}} (1 - sim(ou_x, ou_n))(r_{ou_n,i_u} - \overline{r}_{ou_n}) \Leftrightarrow E_{ou_x,i_u} = \phi \wedge D_{ou_x,i_u} \neq \phi \tag{13}$$

where $\mu_{ou_x,i_u} = \frac{1}{\sum_{n \in D_{ou_x,i_u}} (1 - sim(ou_x, ou_n))} \Leftrightarrow E_{ou_x,i_u} = \phi \wedge D_{ou_x,i_u} \neq \phi$ is a normalization factor, $D_{ou_x,i_u} = \{ou_n \in OU | ou_n \neq ou_x, r_{ou_n,i_u} \neq \phi\}$ is a k_{ou} who did not rate the item i_u and $\#D_{ou_x,i_u}$ is a count of k_{ou} who did not rate the item i_u.

4.3 *Default prediction*

Equation (14) shows the function for default prediction:

$$p_{ou_x,i_u} = \phi \Leftrightarrow E_{ou_x,i_u} = \phi \wedge D_{ou_x,i_u} = \phi. \tag{14}$$

5. Experimental evaluation

5.1 *Performance measures*

The standard performance measures utilized in RS are classified as quantitative and qualitative [32]. Maximum Absolute Error (*MAE*) and Root Mean Squared Error (*RMSE*) are quantitative measures. *Accuracy*, *precision*, *recall* and *coverage* are qualitative measures. These two types of performance measures are described in the following sections.

5.1a *MAE*: *MAE* is an absolute rating difference between the actual ($r_{ou_x,i}$) and the predicted rating ($pr_{ou_x,i}$) of ou_x for i. Let V_{ou_x} be the validation data of ou_x. The *MAE* for a single user is expressed in Eq. (15):

$$MAE_{ou_x} = \begin{cases} \frac{1}{\#V_{ou_x}} \sum_{i \in V_{ou_x}} | pr_{ou_x,i} - r_{ou_x,i} | & V_{ou_x} \neq \phi \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

If *MAE* is considered for all validations *OU* that are *V*, then the redefined equation (15) is as follows:

$$MAE = \begin{cases} \frac{1}{\#V} \sum_{ou \in V} MAE_{ou} & V \neq \phi \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

5.1b *RMSE*: *RMSE* is a square root of the average of the square of all errors. The error is the difference between the actual and predicted rating values. Equations (17) and (18) represent, respectively, the *RMSE* value for a single user and all users separately:

$$RMSE_{ou_x} = \begin{cases} \sqrt{\frac{1}{\#V_{ou_x}} \sum_{i \in V_{ou_x}} (pr_{ou_x,i} - r_{ou_x,i})^2} & V_{ou_x} \neq \phi \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

$$RMSE = \begin{cases} \frac{1}{\#V} \sum_{ou \in V} RMSE_{ou} & V \neq \phi \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

5.1c *Accuracy*: The *accuracy* of RS is the proportion of a number of correct recommendations made to the total number of recommendations given to a user. It is expressed as follows:

$$accuracy = \frac{mumber\ of\ correct\ recommendations}{total\ Number\ of\ recommendations}. \tag{19}$$

This measure is used to approve prediction quality.

5.1d *Precision*: Precision is the ability to make relevant item recommendations from the total number of recommendations given. Let S_{ou} be a set of N recommendations given to user ou and θ be the threshold value to be considered as a relevant recommendation. The precision value for a single user pre_{ou} is given as follows:

$$pre_{ou} = \frac{\{\#i \in S_{ou} | r_{ou,i} \geq \theta\}}{N}. \tag{20}$$

From Eq. (20), the *precision* for all *OU* is defined as follows:

$$precision = \frac{1}{OU} \sum_{ou \in OU} pre_{ou}. \tag{21}$$

5.1e *Recall*: Recall is the ability to obtain an optimal relevant recommendation from the total number of relevant recommendations made. Equation (22) shows the recall of a single user (re_{ou}):

$$re_{ou} = \frac{\{\#i \in S_{ou} | r_{ou,i} \geq \theta\}}{\{\#i \in S_{ou} | r_{ou,i} \geq \theta\} + \{\#i \in S^c_{ou} | r_{ou,i} \geq \theta \wedge r_{ou,i} \neq \phi\}}. \tag{22}$$

Recall for all *OU* is represented as follows:

$$recall = \frac{1}{OU} \sum_{ou \in OU} re_{ou}. \tag{23}$$

5.1f *Coverage*: Coverage is the percentage of items rated by at least one of the k-nearest neighbours of ou in the total number of no co-rated items of ou. Equation (24) describes the coverage measure of the single user (c_{ou}):

$$c_{ou} = \begin{cases} \frac{\#C_{ou}}{\#D_{ou}} \times 100 & D_{ou} \neq \phi \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

Coverage measure for all users is shown in Eq. (25):

$$coverage = \begin{cases} \frac{1}{\#ou} \sum_{ou \in OU} c_{ou} & D_{ou} \neq \phi \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

where D_{ou} represents the active online user who has not yet rated items and C_{ou} represents the number of items from the non-rated items of ou rated by at least one of the k-neighbours of ou.

The following section describes the characteristics of the dataset considered for analysis and shows the performance evaluation of the proposed similarity measure over conventional similarity measures.

5.2 *Experimental system design*

Implementation of user-based *MCF* measure was done and analysed for two benchmark datasets, namely MovieLens [35] and Netflix [36]. The details of these datasets and their sparsity nature are shown in table 2. Density Index (DI) values [28] of datasets show that the Netflix dataset is more sparse than the MovieLens dataset.

Based on the DI value, subsets of MovieLens(ML_1) and Netflix(NF_1) datasets are extracted randomly and their characteristics are shown in table 3. The DI value is calculated with respect to the number of ratings provided by the randomly chosen users for chosen items.

The sparsity nature of the datasets is also proved in terms of the number of co-rated and no co-rated items, which is shown in table 4. Here, both the subsets of datasets have more no co-rated items than co-rated items.

Table 2. Details of datasets.

Dataset	Description	No. of users (OU)	No. of items (I)	No. of ratings (R)	Density index $DI = \frac{R \times 100}{OU \times I}$	Rating range
MovieLens	Movie	6040	3706	1×10^6	4.467	1–5
Netflix	Movie	480,189	17,770	100×10^6	1.172	1–5

Table 3. Details of subset datasets.

Dataset	Subset	No. of users (OU)	No. of items (I)	No. of ratings (R)	Density index $DI = \frac{R \times 100}{OU \times I}$	$\frac{R}{OU}$	$\frac{R}{I}$
MovieLens	ML_1	4599	2650	18562	0.1523	4.0361	7.0045
Netflix	NF_1	6839	879	7002	0.1165	1.0238	7.9659

5.3 *Experimental results and discussion*

Experimental analysis is performed by considering 20% of users as validation data from the ML_1 and NF_1 datasets. Performance values of *MAE*, *RMSE* and coverage are calculated based on the parameter called the number of neighbours (k), where value of k is defined from 2 to 2074 with an interval of 148. The total number of recommendations (N) parameter determines the value of precision, recall and accuracy measures, whose value ranges from 10 to 100 with an interval of 10 recommendations. The interval value for the parameters k and N is fixed based on the significant difference in the value of performance measures. Another parameter, threshold value (θ), is set as 4 to determine the relevant recommendations for *precision* and *recall* by assuming the irrelevant rating values as 1, 2 and 3 and the relevant rating values as 4 and 5. Table 5 gives details of the parameters considered for evaluating performance measures.

The prediction mechanisms Avg, WS and AWA are used to predict the item's rating of an active user. Among the three, an ideal prediction mechanism is picked and used for experimentation. For all conventional similarity measures (PCC, CPCC, WPCC, SPCC, COS, SRC, MSD, Jaccard, JMSD, *PIP* and *PSS*), performance measure *MAE* is estimated based on all three prediction mechanisms and the results are shown in figure 1. Overall, 742 k-nearest neighbours are considered for *MAE* estimation because the error difference for all three prediction mechanisms is noticeably high for 742 neighbours. Figure 1a presents a comparison of prediction mechanisms for ML_1, whereas figure 1b demonstrates comparison results for the NF_1 dataset.

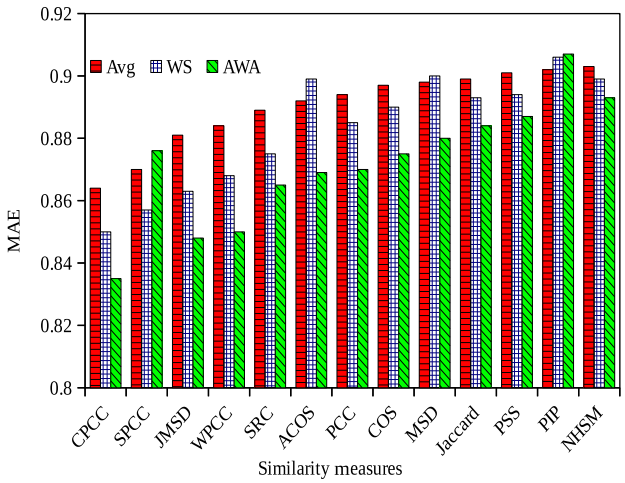
For ML_1 dataset, average *MAE* value of AWA prediction mechanism is reduced by about 1.81% compared with the Avg mechanism and by 1.08% compared with the WS prediction mechanism. Similarly, the reduced average error value of AWA mechanism for NF_1 dataset is 2.14% and 0.67% for Avg and WS prediction mechanisms, respectively. Overall, it is inferred from figure 1 that the proposed AWA prediction mechanism ensures a lower prediction

Table 4. Co-rated and no co-rated item details of ML_1 and NF_1 datasets.

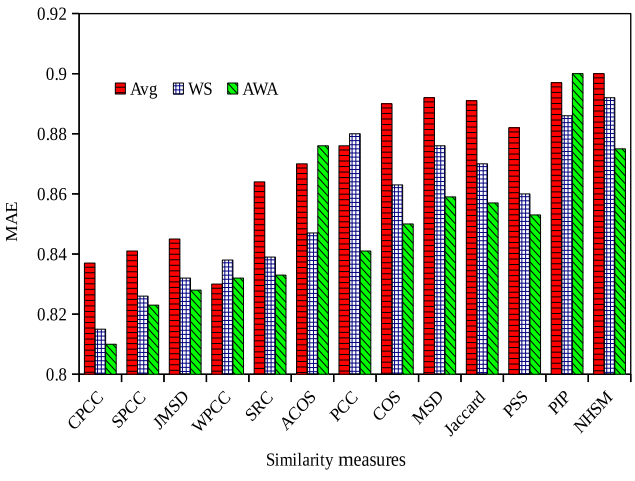
Subset dataset	Number of users	Number of one co-rated items	Number of two co-rated items	Number of three co-rated items	Number of four co-rated items	Number of no co-rated items
ML_1	4599	271180	3626	72	2	168991982
NF_1	6839	248694	–	–	–	95492764

Table 5. Parameters for performance evaluation.

Dataset	Test users	#k-nearest neighbours *MAE*, *RMSE*, Coverage	Step	# Recommendations (N) *precision*, *recall*, *accuracy*	Threshold (θ)
ML_1	20%	Range: 2, ..., 2074	148	Range: 10, 20, ..., 100	4
NF_1	20%	Range: 2, ..., 2074	148	Range: 10, 20, ..., 100	4



(**a**) MovieLens (ML_1)



(**b**) Netflix (NF_1)

Figure 1. Comparison of prediction methods for all conventional similarity measures on *MAE*.

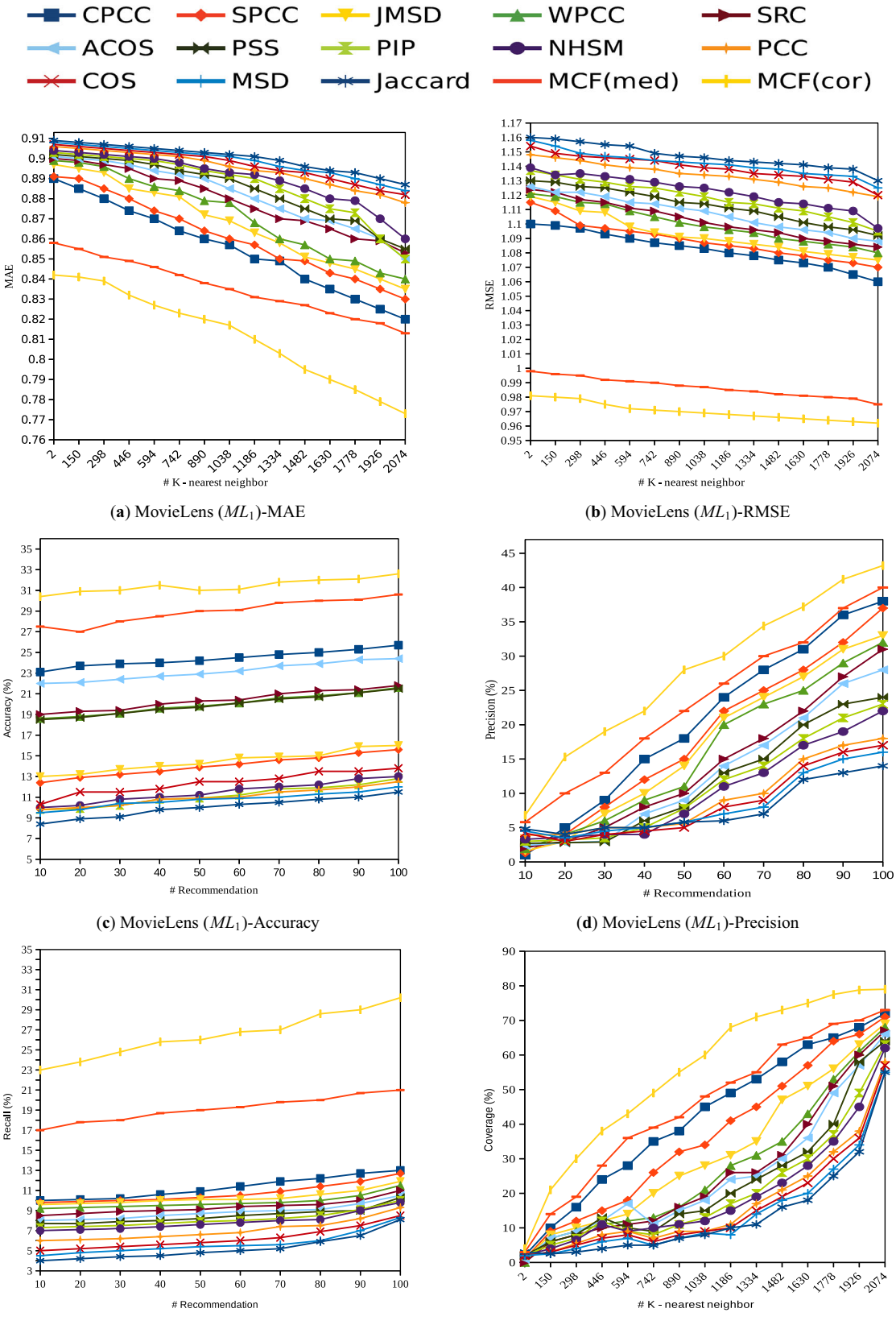
error than other prediction mechanisms. Hence, AWA is considered for further experiments.

Various performance measures such as *MAE*, *RMSE*, *accuracy*, *precision*, *recall* and *coverage* are used to analyse the performance of the proposed *MCF* measure over conventional measures. Figure 2 shows a comparison of similarity measures for the ML_1 dataset. As the proposed *MCF* measure considers both local and global rating information, it is analysed as two types of local similarity estimation techniques, LOC_{cor} and LOC_{med}. Performance results of the proposed *MCF* using LOC_{cor}- and LOC_{med}-based local similarity measures are represented as *MCF*(*cor*) and *MCF*(*med*), respectively. The *MAE* and *RMSE* values obtained by the proposed and conventional similarity measures for the ML_1 dataset are shown in figure 2a and 2b, respectively.

In the sparse dataset, the chance of getting more co-rated items with the k closest neighbours is high when k value is increased, which in turn reduces the error rate of the proposed similarity measures significantly. For a k value with more than 742 users, the error rate of *MAE* and *RMSE*, for both the proposed and conventional similarity measures, is reduced drastically. When the performance difference between the proposed *MCF*(*cor*) and *MCF*(*med*) is considered, *MCF*(*cor*) shows better performance than *MCF*(*med*) since *MCF*(*cor*) considers the average of user's rating instead of a fixed rating scale median (i.e., 3 for 1–5 rating scale). The proposed measures show a lower error rate compared with conventional similarity measures for all k values because they consider both local and global rating information to eliminate the sparsity issue. It is understood from figure 2a and 2b that the conventional similarity measure CPCC has better performance than other conventional measures. The minimum error values of CPCC are 0.82 and 1.065 for *MAE* and *RMSE* measures, respectively. However, the proposed *MCF*(*cor*) shows a minimum error value of 0.773 for *MAE* measure and 0.963 for *RMSE* measure, which is better than the CPCC measure. The average performance improvement of the proposed *MCF*(*med*) over CPCC measure for *MAE* and *RMSE* is 1.96% and 9.546%, respectively. Also, figure 2a and 2b shows that the average performance of the proposed *MCF*(*cor*) measure increased by 4.353% and 11.22% compared with the CPCC measure regarding *MAE* and *RMSE*, respectively.

The percentage changes in performance measures *accuracy*, *precision* and *recall* with respect to the number of recommendations made to active users are shown in figure 2c, 2d and 2e, respectively. When the number of recommendations is above 25, the proposed *MCF* measure achieves significant improvement in performance measures like *accuracy*, *precision* and *recall*, whereas the compared measures attain a decent achievement for recommendations of more than 40. When the proposed and conventional measures are compared for 100 recommendations, the proposed *MCF*(*cor*) measure achieves a maximum performance value of 32.6%, 43.2% and 30.2% for *accuracy*, *precision* and *recall*, respectively. On the other hand, the competitive measure CPCC reveals a performance value for *accuracy*, *precision* and *recall* as 25.7%, 38% and 13%, respectively. The average performance value of the proposed *MCF*(*med*) over CPCC measure for *accuracy*, *precision* and *recall* is 4.54%, 2.88% and 7.83%, respectively. Also, figure 2c, 2d and 2e proves that the proposed *MCF*(*cor*) performs an average of 7.02%, 7.21% and 15.2% for *accuracy*, *precision* and *recall*, respectively, which are better than those of conventional similarity measures.

Figure 2f presents the change of percentage in coverage measure concerning the number of k-nearest neighbours. The proposed measure achieves 79% coverage, which is a 7% improvement compared with the CPCC measure. An average 3.33% improvement is seen in the proposed



(**a**) MovieLens (ML_1)-MAE

(**b**) MovieLens (ML_1)-RMSE

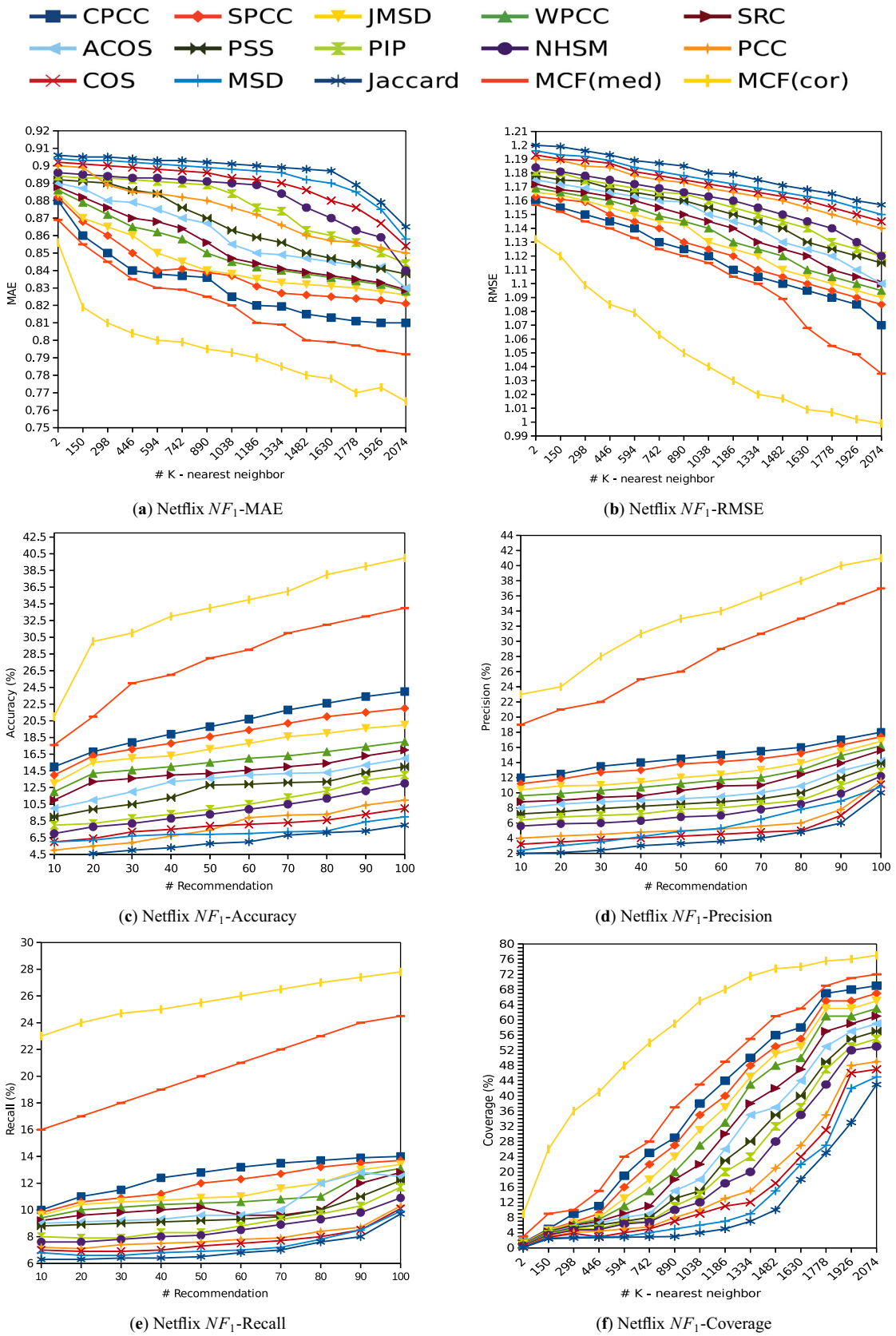
(**c**) MovieLens (ML_1)-Accuracy

(**d**) MovieLens (ML_1)-Precision

(**e**) MovieLens (ML_1)-Recall

(**f**) MovieLens (ML_1)-Coverage

Figure 2. Results of the performance measures in ML_1 sparse dataset.



(**a**) Netflix NF_1-MAE

(**b**) Netflix NF_1-RMSE

(**c**) Netflix NF_1-Accuracy

(**d**) Netflix NF_1-Precision

(**e**) Netflix NF_1-Recall

(**f**) Netflix NF_1-Coverage

Figure 3. Results of the performance measures in NF_1 sparse dataset.

MCF(*med*) over the CPCC measure. The proposed *MCF*(*cor*) improves 13.09% for coverage on average compared with the CPCC measure. This discussion proves that the proposed similarity measures *MCF*(*med*) and *MCF*(*cor*) achieve better performance than those of the compared similarity measures. The proposed measure *MCF*(*med*) shows slightly lower performance than that of *MCF*(*cor*) and better performance than those of conventional measures.

The performance results of NF_1 dataset is shown in figure 3 through the same experiment as in ML_1 dataset. To show the effectiveness of the proposed measures *MCF*(*med*) and *MCF*(*cor*), NF_1 dataset is chosen with a slightly higher sparsity than that of ML_1 (see table 4). Figure 3a and 3b presents performance measures of *MAE* and *RMSE*, respectively, with respect to various k values. Even NF_1 has more sparsity, but the proposed *MCF*(*cor*) and *MCF*(*med*) show minimum errors compared with the similarity measures. From figure 3a and 3b, the minimum error values of CPCC for *MAE* and *RMSE* are 0.81 and 1.07, respectively. However, the proposed *MCF*(*cor*) shows a minimum error value of 0.77 for *MAE* measure and 0.99 for *RMSE* measure, which is lesser than that of CPCC. If CPCC is considered as the immediate performer, then the proposed *MCF*(*med*) measure improves the average performance of *MAE* and *MAE* over CPCC by 1.04% and 1.28%, respectively. The proposed *MCF*(*cor*) improves performance on an average by 3.65% and 6.85% for *MAE* and *RMSE* measures, respectively.

Figure 3c, 3d and 3e shows the percentage change in performance measures *accuracy*, *precision* and *recall* with respect to the number of recommendations made to active users, respectively. The proposed measure *MCF* achieves significant improvement in *accuracy*, *precision* and *recall* when the number of recommendations is more than 20, whereas the compared measures attain a decent achievement over 40 recommendations. When the proposed and conventional measures are compared for 100 recommendations, the proposed measure *MCF*(*cor*) achieves the maximum performance value of 40%, 41% and 27.8% for *accuracy*, *precision* and *recall*, respectively. On the other hand, the competitive measure CPCC has a performance value of 24%, 18% and 14% for *accuracy*, *precision* and *recall*, respectively. The average performance improvements of *MCF*(*med*) over CPCC measure for *accuracy*, *precision* and *recall* are 7.57%, 13% and 7.85%, respectively. This proves that the proposed *MCF*(*cor*) performs on an average 13.61%, 18% and 13.09%, respectively, for *accuracy*, *precision* and *recall*, which are better than those of the conventional similarity measures.

The percentage change in *coverage* measure concerning the number of k-nearest neighbours is shown in figure 3f. The proposed measure achieves 77% coverage, which is more than 8% improvement when compared with the CPCC measure. The average improvement of the performance measure *coverage* for the proposed *MCF*(*med*) measure over CPCC measure is 3.97%. The proposed *MCF*(*cor*) improves coverage value by 20.27% on average over the conventional CPCC similarity measure.

Similar to ML_1, it can be concluded that the proposed similarity measures *MCF*(*med*) and *MCF*(*cor*) for NF_1 achieve better performance than conventional similarity measures. Also, the proposed measure *MCF*(*med*) shows slightly lower performance than proposed *MCF*(*cor*) measure. Overall, it can be concluded that the proposed *MCF*(*cor*) measure is considered as an efficient method for RS over conventional similarity measures in sparse datasets.

6. Conclusions

Conventional similarity measures cannot provide effective recommendations to an active user in a sparse dataset. As the sparse dataset contains less co-rated items, the conventional measures fail to consider the no co-rated item values. The proposed *MCF* measure efficiently utilizes all rating information without considering only user-provided co-rated item values. As a result, *MCF* measure offers an efficient recommendation to an active user by finding reliable neighbours and outperforms conventional similarity measures. Experimental analysis on benchmark datasets MovieLens and Netflix proves that the proposed *MCF* measure removes the sparsity issue and provides effective recommendations with fewer user and item ratings.

References

[1] Basu Chumki, Hirsh Haym, Cohen William, et al 1998 Recommendation as classification: Using social and content-based information in recommendation. In: *Aaai/iaai*, pages 714–720

[2] Senecal Sylvain and Nantel Jacques 2004 The influence of online product recommendations on consumers online choices. *Journal of Retailing*, 80(2): 159–169

[3] Mooney Raymond J and Roy Loriene 2000 Content-based book recommending using learning for text categorization. In: *Proceedings of the fifth ACM conference on Digital libraries*, pages 195–204. ACM

[4] Hicken Wendell, Holm Frode, Clune James and Campbell Marc 2004 Music recommendation system and method, August 13. US Patent App. 10/917,865

[5] Davidson James, Liebald Benjamin, Liu Junning, Nandy Palash, Van Vleet Taylor, Gargi Ullas, Gupta Sujoy, He Yu, Lambert Mike, Livingston Blake, et al 2010 The youtube video recommendation system. In: *Proceedings of the fourth ACM conference on Recommender systems*, pages 293–296. ACM

[6] Sinha Rashmi R and Swearingen Kirsten 2001 Comparing recommendations made by online systems and friends. In: *DELOS workshop: personalisation and recommender systems in digital libraries*, volume 106

[7] Kabassi Katerina 2010 Personalizing recommendations for tourists. *Telematics and Informatics* 27(1): 51–66

[8] Burke Robin 2002 Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4): 331–370

[9] Lang Ken 1995 Newsweeder: Learning to filter netnews. In: *Proceedings of the 12th international conference on machine learning*, pages 331–339

[10] Lops Pasquale, De Gemmis Marco and Semeraro Giovanni 2011 Content-based recommender systems: State of the art and trends. In: *Recommender systems handbook*, pages 73–105. Springer

[11] Adomavicius Gediminas and Tuzhilin Alexander 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

[12] Bobadilla Jesús, Ortega Fernando, Hernando Antonio and Gutiérrez Abraham 2013 Recommender systems survey. *Knowledge-based Systems* 46: 109–132

[13] Breese John S, Heckerman David and Kadie Carl 1998 Empirical analysis of predictive algorithms for collaborative filtering. In: *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 43–52. Morgan Kaufmann Publishers Inc

[14] Cacheda Fidel, Carneiro Víctor, Fernández Diego and Formoso Vreixo 2011 Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. *ACM Transactions on the Web (TWEB)*, 5(1): 2

[15] Koren Yehuda 2010 Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(1): 1

[16] Ning Xia, Desrosiers Christian and Karypis George 2015 A comprehensive survey of neighborhood-based recommendation methods. In: *Recommender systems handbook*, pages 37–76. Springer

[17] Yildirim Hilmi and Krishnamoorthy Mukkai S 2008 A random walk method for alleviating the sparsity problem in collaborative filtering. In: *Proceedings of the 2008 ACM conference on Recommender systems*, pages 131–138. ACM

[18] Shardanand Upendra and Maes Pattie 1995 Social information filtering: algorithms for automating word of mouth. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217. ACM Press/Addison-Wesley Publishing Co

[19] Herlocker Jonathan L, Konstan Joseph A, Borchers Al and Riedl John 1999 An algorithmic framework for performing collaborative filtering. In: *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237. ACM

[20] Jamali Mohsen and Ester Martin 2009 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*, pages 397–406. ACM

[21] Salton Gerard and McGill Michael J 1986 Introduction to modern information retrieval

[22] Sarwar Badrul, Karypis George, Konstan Joseph and Riedl John 2001 Item-based collaborative filtering recommendation algorithms. In: *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM

[23] Herlocker Jonathan L, Konstan Joseph A, Terveen Loren G and Riedl John T 2004 Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1): 5–53

[24] Koutrika Georgia, Bercovitz Benjamin and Garcia-Molina Hector 2009 Flexrecs: expressing and combining flexible recommendations. In: *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 745–758. ACM

[25] Bobadilla Jesús, Serradilla Francisco and Bernal Jesus 2010 A new collaborative filtering metric that improves the behavior of recommender systems. *Knowledge-Based Systems* 23(6): 520–528

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

[27] Patra Bidyut Kr, Launonen Raimo, Ollikainen Ville and Nandi Sukumar 2014 Exploiting bhattacharyya similarity measure to diminish user cold-start problem in sparse data. In: *International Conference on Discovery Science*, pages 252–263. Springer

[28] Patra Bidyut Kr, Launonen Raimo, Ollikainen Ville and Nandi Sukumar 2015 A new similarity measure using bhattacharyya coefficient for collaborative filtering in sparse data. *Knowledge-Based Systems* 82: 163–177

[29] Fu King-Sun et al 1976 Pattern recognition and image processing. *IEEE Transactions on Computers* 100(12): 1336–1346

[30] Basseville Michele 1989 Distance measures for signal processing and pattern recognition. *Signal processing* 18(4): 349–369

[31] Nikulin M S 2001 Hellinger distance. hazewinkel, michiel, encyclopedia of mathematics. *Springer, Berlin. doi*, 10: 1361684–1361686

[32] Bobadilla Jesús, Ortega Fernando and Hernando Antonio 2012 A collaborative filtering similarity measure based on singularities. *Information Processing & Management* 48(2): 204–217

[33] Aherne Frank J, Thacker Neil A and Rockett Peter I 1998 The bhattacharyya metric as an absolute similarity measure for frequency coded data. *Kybernetika* 34(4): 363–368

[34] Bobadilla Jesus, Hernando Antonio, Ortega Fernando and Bernal Jesus 2011 A framework for collaborative filtering recommender systems. *Expert Systems with Applications* 38(12): 14609–14623

[35] Movielens dataset. http://www.grouplens.org

[36] Netflix dataset. http://www.netflixprize.com