**Similarity measures for collaborative filtering**

Ali A. Amer

TAIZ University, Yemen

Loc Nguyen

Loc Nguyen’s Academic Network, Vietnam

**Abstract**

Collaborative filtering (CF) is an important approach for recommendation system. CF recommends an item to a user if her/his neighbors are interested in such item. One of popular algorithms in CF is nearest neighbors (NN) algorithm. Finding neighbors of a user is main problem of NN algorithm. As usual, similarity measures between two users such as cosine and Pearson are used to determine nearest neighbors of a user. In this research, we test many similarity measures besides cosine and Pearson for NN algorithm. Not like other researches, we split the evaluation process into two sub-processes such as estimation process and recommendation process in order to obtain appropriateness in evaluation. Experiment is designed specially for such two sub-processes, in which we propose a formula to calculate dynamic recommendation count based on both dataset and each rating vector. Moreover, we propose four new similarity measures called SMD, HSMD, quasi-TF-IDF (QTI), and numerical nearby measure (NNMS) with note that QTI is an extension of HSMD and it follows ideology of term frequency (TF) and inverse document frequency (IDF) in information retrieval. NNMS which is a variant of Jaccard measure combines sums of rating values and cardinalities of item sets. The experiment proves that SMD is a preeminent measure which is better than Jaccard.

**Keywords:** collaborating filtering (CF), term frequency (TF), inverse document frequency (IDF), similarity measure, nearest neighbors (NN) algorithm, rating matrix.

**1. Introduction**

Recommendation system is a system which recommends items to users among many existing items in database. Item is anything which users consider, such as product, book, and newspaper. There are two main approaches for recommendation such as content-based filtering (CBF) and collaborative filtering (CF). CF recommends an item to a user if her/his neighbors (other users like her/him) are interested in such item. One of popular algorithms in CF is nearest neighbors (NN) algorithm. The essence of NN algorithm (Torres Júnior, 2004, pp. 16-18) is to find out nearest neighbors of a regarded user (called active user) and then to recommend active user items that these neighbors may like. Let ***U*** = {*u*1, *u*2,…, *um*} be the set of users and let ***V*** = {*v*1, *v*2,…, *vn*} be the set of items. User-based rating matrix is the matrix in which rows indicate users and columns indicate items and each cell is a rating which a user gave to an item. In other words, each row in user-based rating matrix is a rating vector of a specified user. Rating vector of active user is called active user vector. As a convention, rating matrix implies user-based rating matrix if there is no additional explanation. Table 1 is an example of user-based rating matrix in which missing values are denoted by question masks (Do, Nguyen, & Nguyen, 2010, p. 218) and ratings values range from 1 to 5. In Table 1, active vector is *u*4 = (*r*41=1, *r*42=2, *r*43=?, *r*44=?), which is shaded.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Item 1 | Item 2 | Item 3 | Item 4 |
| User 1 | *r*11 = 1 | *r*12 = 2 | *r*13 = 1 | *r*14 = 5 |
| User 2 | *r*21 = 2 | *r*22 = 1 | *r*23 = 2 | *r*24 = 4 |
| User 3 | *r*31 = 4 | *r*32 = 1 | *r*33 = 5 | *r*34 = 5 |
| User 4 | *r*41 = 1 | *r*42= 2 | *r*43 = ? | *r*44 = ? |

**Table 1.** User-based rating matrix

User-based rating matrix can be transposed into item-based rating matrix in which each row is a rating vector of a specified item. Table 2 is the item-based rating matrix which is transposed from the user-based rating matrix shown in Table 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | User 1 | User 2 | User 3 | User 4 |
| Item1 | *r*11 = 1 | *r*21 = 2 | *r*31 = 4 | *r*41 = 1 |
| Item2 | *r*12 = 2 | *r*22 = 1 | *r*32 = 1 | *r*42 = 2 |
| Item3 | *r*13 = 1 | *r*23 = 2 | *r*33 = 5 | *r*43 = ? |
| Item4 | *r*14 = 5 | *r*24 = 4 | *r*34 = 5 | *r*44 = ? |

**Table 2.** Item-based rating matrix

In Table 2, active item vectors are *v*3 = (*r*13=1, *r*23=2, *r*33=5, *r*43=?) and *v*4 = (*r*14=5, *r*24=4, *r*34=5, *r*44=?), which are shaded.

In Table 1, there are four rating vectors *u*1 = (1, 2, 1, 5), *u*2 = (2, 1, 2, 4), *u*3 = (4, 1, 5, 5), and *u*4 = (1, 2, *r*43=?, *r*44=?). Suppose the active rating vector is *u*4, NN algorithm will find out nearest neighbors of *u*4 and then compute the predictive values for *r*43 and *r*44 based on similarities between these neighbors and *u*4. The NN algorithm which acts on user-based rating matrix is called user-based NN algorithm and the NN algorithm which acts on item-based rating matrix is called item-based NN algorithm. Although ideology of user-based NN algorithm and item-based NN algorithm is the same, their implementations are slightly different. User-based NN algorithm is mentioned by default. In general, NN algorithm includes two steps (Torres Júnior, 2004, pp. 17-18):

1. Find out nearest neighbors of the active user by calculating similarities between active vector and other vectors. The more the similarity is, the nearer two users are. Given a threshold, users whose similarities between them and active user are equal to or larger than a threshold are considered as nearest neighbors of active user.
2. Compute predictive values for missing ratings of active vector. The computation is based on ratings of nearest neighbors and similarities calculated in step 1. The items whose predictive values are high enough are recommended to active user.

The essence of NN algorithm is to use similarity measures in order to find out nearest neighbors of an active rating vector. This research focuses on similarity measures for CF. The most popular similarity measures are cosine and Pearson. Given two rating vectors *u*1 = (*r*11, *r*12,…, *r*1*n*) and *u*2 = (*r*21, *r*22,…, *r*2*n*) of user 1 and user 2, in which user 1 is considered as active user and some *rij* can be missing (empty). Let *I*1 and *I*2 be set of indices of items that user 1 and user 2 rated, respectively. Let denote intersection set of *I*1 and *I*2 and let denotes union set of *I*1 and *I*2. All items whose indices belong to are rated by both user 1 and user 2. In other words, all items whose indices belong to co-exist in vectors *u*1 and *u*2. All items whose indices belong to are rated by user 1 or user 2. Notation |x| indicates absolute value of number, length of vector, length of geometric segment, or cardinality of set, which depends on context. Please pay attention to these denotations.

Let sim(*u*1, *u*2) denote the similarity of *u*1 and *u*2. For instance, the cosine measure of *u*1 and *u*2 is defined as follows (Torres Júnior, 2004, p. 17):

|  |  |
| --- | --- |
|  | (1) |

Where |*u*1| and |*u*2| are lengths of *u*1 and *u*2, respectively whereas *u*1•*u*2 is dot product (scalar product) of *u*1 and *u*2, respectively. If all ratings are non-negative, range of cosine measure is from 0 to 1. If it is equal to 0, two users are totally different. If it is equal to 1, two users are identical. Cosine measure will be mentioned more later. The larger the similarity is, the more the user 2 is near to active user 1. Hence, the similarity is used to determine the list of neighbors of active user. Suppose NN algorithm finds out *k* neighbors of *u*1, let *N* be set of indices of *k* neighbors of *u*1. Of course, we have |*N*| = *k*. A missing value *r*1*j* of *u*1 is computed (predicted) based on ratings of nearest neighbors and similarities according to step 2 of NN algorithm (Torres Júnior, 2004, p. 18).

|  |  |
| --- | --- |
|  | (2) |

Where and are mean values of *u*1 and *ui*, respectively. Equation 2 above is called prediction formula or estimation formula.

Where *Ii* is the set of indices of items that user *i* rated. The missing value *r*1*j* of *u*1 can be predicted more simply as follows:

|  |  |
| --- | --- |
|  | (3) |

In general, similarity measure is the heart of NN algorithm because prediction formulas are based on similarity measures. Pearson correlation is another popular similarity measure besides cosine, which is defined as follows (Sarwar, Karypis, Konstan, & Riedl, 2001, p. 290):

|  |  |
| --- | --- |
|  | (4) |

Where and are mean values of *u*1 and *u*2, respectively.

The range of Pearson measure is from –1 to 1. If it is equal to –1, two users are totally opposite. If it is equal to 1, two users are identical. Pearson measure is sample correlation coefficient in statistics. Pearson measure has some variants. Constrained Pearson correlation (CPC) measure considers impact of positive and negative ratings by using median *rm* instead of using the means; for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. CPC measure is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (5) |

The similarity will be significant if both users rated more common items. Weight Pearson correlation (WPC) measure and sigmoid Pearson correlation (SPC) measure concern how much common items are. WPC is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (6) |

SPC is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (7) |

Where *H* is a threshold and it is often set to be 50 (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158).

Jaccard measure is ratio of cardinality of common set to cardinality of union set . It measures how much common items both users rated, which is defined as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (8) |

Another version of Jaccard is (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (9) |

Jaccard does not concerns magnitude of rating values but cosine measure does. By following the ideology of Jaccard measure, cosine measure is modified as follows:

|  |  |
| --- | --- |
|  | (10) |

Let *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. The normalized cosine measure (CON) [4, p. 158] is defined as follows:

Obviously, CON measure is CPC measure (see equation 5).

Let *vj* = (*r*1*j*, *r*2*j*,…, *rmj*) be vector of rating values that item *j* receives from *m* users, for example. The mean of *vj* is:

Adjusted cosine measure (COD) is defined as follows:

|  |  |
| --- | --- |
|  | (11) |

Jaccard can be combined with any measure. For instance, CosineJ is combinations of Jaccard and cosine as follows:

|  |  |
| --- | --- |
|  | (12) |

PearsonJ is combinations of Jaccard and Pearson as follows:

|  |  |
| --- | --- |
|  | (13) |

Mean squared difference (MSD) is defined as inverse of distance between two vectors. Let MAX be maximum value of ratings, MSD is calculated as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (14) |

Another variant of MSD is specified by some authors as follows:

|  |  |
| --- | --- |
|  | (15) |

MSD measure combines with Jaccard measure, which derives MSDJ measure as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 158):

|  |  |
| --- | --- |
|  | (16) |

When rating values are converted into ranks, Spearman’s Rank Correlation (SRC) is defined as follows (Hyung, 2008, p. 39):

|  |  |
| --- | --- |
|  | (17) |

Where *dj* is difference between two ranks on item *j* given by user 1 and user 2.

Note, it is easy to convert ratings values to ranks. For example, suppose rating values (bins) are 5, 6, 7, 8, 9 then, we have rank 1 (for value 9), rank 2 (for value 8) , rank 3 (for value 7), rank 4 (for value 6), and rank 5 (for value 5). If user 1 rates value 9 to item *j*, we have *rank*1*j* = 1. The larger the value is, the smaller (higher) the rank is.

There are some other researches related to apply similarity measures into CF. Ahn (Hyung, 2008) proposed a heuristic measure to solve cold-starting problem which relates to missing data in which there is not enough information to calculate similarities between rating vectors (Hyung, 2008, p. 39). The measure called PIP measure based on concept of “agreement” in rating. If both user 1 and user 2 like or dislike the same item, it is called that they have a rating “agreement” on such item. Let *r*1*j* and *r*2*j* be ratings of user 1 and user 2 on item *j*, respectively, the agreement (Hyung, 2008, p. 43) of them is defined as follows:

Note, *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. PIP measure (Hyung, 2008, p. 42) is sum of products of triples Proximity, Impact, and Popularity.

|  |  |
| --- | --- |
|  | (18) |

Proximity (Hyung, 2008, p. 43) indicates similarity of two ratings, based on agreement and distance between them. The distance is increased twice as a penalty if such two ratings are not agreed.

Where *rmin* and *rmax* are minimum rating value and maximum rating value, respectively. If two ratings are agreed, their impact (Hyung, 2008, p. 43) is proportional to difference between them and rating median. If two ratings are disagreed, their impact is inverse of such difference.

Popularity (Hyung, 2008, p. 43) indicates difference between ratings given by active users and the average rating.

Note, *μj* is average rating of item *j*, which is same mean of rating values of item *j*. Experimental results proved that cold-starting problem is solved well by PIP measure (Hyung, 2008, p. 47).

Choi and Suh (Choi & Suh, 2013) proposed a so-called PC measure which is Pearson measure weighted by similarities of items. In other words, PC measure combines similarities of users and items (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 4). The ideology is excellent. PC measure can be applied into any foundation measures. Each factor in PC measure is weighted by a similarity of active item and another item. Suppose it is necessary to estimate rating values of active item *k*, PC measure (Choi & Suh, 2013, p. 148) is defined as follows:

|  |  |
| --- | --- |
|  | (19) |

Where sim(*vk*, *vj*) is similarity of the active item *k* and item *j*. Note, sim(*vk*, *vj*) can be calculated by any measures here. The and are mean values of *u*1 and *u*2, respectively.

Experimental results proved that PC is an effective measure.

Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 156) proposed a new similarity measure called NHMS to improve recommendation task in which only few ratings are available. Their NHMS measure (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 160) is based on sigmoid function and the improved PIP measure as PSS (*Proximity* – *Significance* – *Singularity*). PSS similarity is calculated as follows (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 160):

|  |  |
| --- | --- |
|  | (20) |

Where, is intersection set of *I*1 and *I*2. The proximity factor determines similarity of two ratings, based on distance between them; such distance is as less as better. The significance factor determines similarity of two ratings, based on distance from them to rating median; such distance is as more as better. The significance factor determines similarity of two ratings, based on difference between them and other ratings; such difference is as less as better. Followings are equations of these factors based on sigmoid function (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 161).

Note, *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3 whereas *μj* is rating mean of item *j*. Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 161) also considered the similarity between two users via URP measure as follows:

|  |  |
| --- | --- |
|  | (21) |

Where *μ*1 and *μ*2 are rating means of user 1 and user 2, respectively and *σ*1 and *σ*2 are rating standard deviations of user 1 and user 2, respectively.

Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013, p. 161) proposed a new heuristic similarity model (NHSM) as triple product of PSS measure, URP measure, and Jaccard2 measure.

|  |  |
| --- | --- |
|  | (22) |

In general, Liu et al. (Liu, Hu, Mian, Tian, & Zhu, 2013) aim to alleviate the problem of few rated common items via their NHSM measure. From experimental result, NHSM gave out excellent estimation.

Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 143) proposed a new similarity measure called BCF for CF, which uses all ratings made by a pair of users. Proposed measure finds importance of each pair of rated items by exploiting Bhattacharyya (BC) similarity. The BC similarity, which is core of their own measure, measures the similarity between two distributions. So, these distributions are estimated as the number of uses rated on given item. In general, Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5) combined BC similarity and the local similarity where the local similarity relates to Pearson correlation. It is necessary to survey BC similarity. Bin is a terminology indicating domain of rating values, for example, if rating values range from 1 to 5, we have bins: 1, 2, 3, 4, 5. Let *m* be the number of bins, given items *i* and *j*, item BC coefficient for items is calculated as follows (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5):

|  |  |
| --- | --- |
|  | (23) |

Note, #*i* and #*j* are the numbers of users who rated items *i* and *j*, respectively whereas #*hi* and #*hj* are numbers of users who gave rating value *h* on items *i* and *j*, respectively. So, item BC coefficient concerns two items. In table 1.2, rating vectors of item 3 and item 4 are *v*3 = (1, 2, 5, ?) and *v*4 = (5, 4, 5, ?), respectively with note that rating values range from 1 to 5 and so we have:

According to Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5), user BC similarity is sum of products of item BC coefficients and local similarities as follows:

|  |  |
| --- | --- |
|  | (24) |

The local similarity is calculated as a part of constrained Pearson coefficient (CPC) as follows:

Note, *rm* be median of rating values, for example, if rating values range from 1 to 5, the median is *rm* = (1+5) / 2 = 3. Patra et al. (Patra, Launonen, Ollikainen, & Nandi, 2015, p. 5) proposed Bhattacharyya similarity in CF (BCF) as sum of user BC similarity and Jaccard measure as follows:

|  |  |
| --- | --- |
|  | (25) |

Suryakant and Mahara (Suryakant & Mahara, 2016) proposed a so-called Cosine-Jaccard-Mean Measure of Divergence (CjacMD) based on Mean Measure of Divergence (MMD) to solve the problem of sparse rating matrix. Because MMD measure takes advantages of statistical aspects, it can alleviate sparsity. MMD focuses on personal habits which are ignored by nonstatistical measures (Suryakant & Mahara, 2016, p. 453). Recall that bin is a terminology indicating domain of rating values, for example, if rating values range from 1 to 5, we have bins: 1, 2, 3, 4, 5. Let *X* = (*x*1, *x*2,…, *xb*) and *Y* = (*y*1, *y*2,…, *yb*) be count vectors of user 1 and user 2, respectively where *xj* (*yj*) is the number of items to which user 1 (user 2) gives bin *j* with note that *b* is the number of bins. For example, rating vectors of user 1 and user 2 in table 1.1 are *u*1 = (1, 2, 1, 5) and *u*2 = (2, 1, 2, 4), respectively with note that rating values ranges from 1 to 5. We have:

MMD measure is defined as follows (Suryakant & Mahara, 2016, p. 453), (Harris & Sjøvold, 2018, p. 87):

|  |  |
| --- | --- |
|  | (26) |

Where *θ*1\* and *θ*2\* are Grewal’s transformations (Harris & Sjøvold, 2018, p. 85) of *X* and *Y*, respectively.

In fact, CjacMD (Suryakant & Mahara, 2016, p. 453) combines three other measures such as cosine, Jaccard, and MMD together.

|  |  |
| --- | --- |
|  | (27) |

Experimental result proved that CjacMD model is effective similarity model.

Sun et al. (Sun, et al., 2017) proposed a so-called Triangle similarity measure which considers both angle and lengths of rating vectors. For instance, given two user vectors *u*1 and *u*2 are considered as two vector OA = *u*1 and OB = *u*2 and hence, OAB forms a triangle. TS measure is ratio of the length |AB| to the sum of lengths |OA| + |OB|. Of course, |AB| is always less than or equal to |OA| + |OB| according to triangle inequality. The idea is excellent. TS measure (Sun, et al., 2017, p. 6) is defined as follows:

|  |  |
| --- | --- |
|  | (28) |

Sun et al. also combined Triangle measure and Jaccard measure to form a new measure called Triangle multiplying Jaccard (TMJ) measure. The integrated TMJ (Sun, et al., 2017, p. 6) is defined as follows:

|  |  |
| --- | --- |
|  | (29) |

Experimental result proved that TMJ is effective measure.

To solve the problem of sparse rating matrix, Feng et al. (Feng, Fengs, Zhang, & Peng, 2018) proposed a new model of similarity which includes three parts such as *S*1, *S*2, and *S*3. The *S*1 (Feng, Fengs, Zhang, & Peng, 2018, p. 6) is normal similarity and they choose cosine as *S*1.

Where *ρ* is sparsity threshold which is proposed by Feng et al. The *S*2 (Feng, Fengs, Zhang, & Peng, 2018, p. 6) punishes user pairs whose co-rated items are few.

The *S*3 (Feng, Fengs, Zhang, & Peng, 2018, p. 6) focuses on statistical feature of user ratings, which reflects essential user favorites. *S*3 is aforementioned URP measure.

Where *μ*1 and *μ*2 are rating means of user 1 and user 2, respectively and *σ*1 and *σ*2 are rating standard deviations of user 1 and user 2, respectively. The similarity model of Feng et al. (Feng, Fengs, Zhang, & Peng, 2018, p. 5) is product of *S*1, *S*2, and *S*3 as follows:

|  |  |
| --- | --- |
|  | (30) |

Experimental result proved that Feng model is effective similarity model.

Mu et al. (Mu, Xiao, Tang, Luo, & Yin, 2019) combined local measures (Pearson and Jaccard) with global measure to solve the problem of sparse rating matrix. The global measure is Hellinger (Hg) distance which estimates similarity of two probabilistic distributions. In fact, Hg is inverse of BC coefficient in discrete distributions as follows (Mu, Xiao, Tang, Luo, & Yin, 2019, p. 419):

|  |  |
| --- | --- |
|  | (31) |

Note, #1 and #2 are the numbers of item which are rated by user 1 and user 2, respectively whereas #*h*1 and #*h*2 are numbers of items which receive rating value *h* from user 1 and user 2, respectively. For example, rating vectors of user 1 and user 2 in table 1.1 are *u*1 = (1, 2, 1, 5) and *u*2 = (2, 1, 2, 4), respectively with note that rating values range from 1 to 5 and so we have:

Given weight *α*, the Mu measure (Mu, Xiao, Tang, Luo, & Yin, 2019, p. 419) combines Pearson, Jaccard, and Hg as follows:

|  |  |
| --- | --- |
|  | (32) |

Experimental result proved that Mu measure is effective similarity model.

This research also implements the Similarity Measure for Text Processing (SMTP) for testing. SMTP was developed by Lin, Jiang, and Lee (Lin, Jiang, & Lee, 2013), which as originally used for computing the similarity between two documents in text processing. Here documents are considered as rating vectors. Given two rating vectors *u*1 = (*r*11, *r*12,…, *r*1*n*) and *u*2 = (*r*21, *r*22,…, *r*2*n*), the function *F* of *u*1 and *u*2 is defined as follows (Lin, Jiang, & Lee, 2013, p. 1577):

|  |  |
| --- | --- |
|  | (33) |

Where (Lin, Jiang, & Lee, 2013, p. 1577),

Note that *λ* is the pre-defined number and *σj* is the standard deviation of rating values belonging to field *j* (item *j*). In this research, *λ* is set to be 0.5. Lin, Jiang, and Lee (Lin, Jiang, & Lee, 2013, p. 1577) defined SMTP measure based on function *F* as follows:

|  |  |
| --- | --- |
|  | (34) |

In this research, we test our proposed measures mentioned in section 2 with these measures mentioned in this section.

**2. Proposed similarity measures**

Given two rating vectors *u*1 = (*r*11, *r*12,…, *r*1*n*) and *u*2 = (*r*21, *r*22,… , *r*2*n*) of user 1 and user 2, respectively, in which some *rij* can be missing (empty). In binary representation, *rij* is converted into 1 if it is non-missing (rated) and otherwise, *rij* is converted into 0 if it is missing (not rated). Let *N*12 be the number of common values “1” in both *u*1 and *u*2. Let *N* be the total number of all items under consideration; in this case, *N* = *n*. Let *N*1 and *N*2 be the numbers of values “1” of *u*1 and *u*2, respectively. Let *F* be the number of differences between *u*1 and *u*2; for example, the fact that *r*11 = 0 and *r*21 = 1 contributes one difference to *F*. Amer defined a so-called *SMD measure* in binary representation as follows:

|  |  |
| --- | --- |
|  | (35) |

Let *I*1 or *I*2 be sets of indices of items that user 1 or user 2 rates, respectively. Amer also defined another so-called *HSMD measure* in numerical representation in which values *rij* are kept in numerical values as rating values, as follows:

|  |  |
| --- | --- |
|  | (36) |

Where, *R*1 (*R*2) is the sum of non-missing values *r*1*j* (*r*2*j*) of *u*1 (*u*2) such that respective values *r*2*j* (*r*1*j*) are missing.

Note, notation “\” denote complement operator in set theory. *G* is product of two sums of non-missing values for both *r*1 and *r*2.

In general, measures SMD and HSMD are defined firstly for weight vectors of documents in information retrieval, in which every element of a vector is a weight which is product of term frequency (TF) and inverse document frequency (IDF). Here they are applied into CF. For example, given two rating vectors *u*1 = (*r*11=2, *r*12=5, *r*13=7, *r*14=8, *r*15=?, *r*16=9) and *u*2 = (*r*21=9, *r*22=?, *r*23=?, *r*24=6, *r*25=5, *r*26=1). Binary representations of these two vectors are (1, 1, 1, 1, 0, 1) and (1, 0, 0, 1, 1, 1). According to SMD measure, we have *N*12=3, *N*=6, *F*=3, *N*1=5, and *N*2=4. Hence, SMD measure is calculated according to equation 35, as follows:

According to HSMD measure, we have *R*1 = 5+7 = 12, *R*2 = 5, and *G* = (2 + 5 + 7 + 8 + 9) \* (9 + 6 + 5 + 1) = 651. Hence, HSMD measure is calculated according to equation 36, as follows:

When HSMD measure is combined with Jaccard measure, it is called HSMDJ which is specified by equation 37.

|  |  |
| --- | --- |
|  | (37) |

We develop a new measure which is based on HSMD measure and ideology of TF and IDF. Firstly, we research deeply HSMD measure in which the ratio *R*1*R*2/G indicates difference between *u*1 and *u*2. In other words, such ratio implies uniqueness of each vector, which means that the ratio *R*1*R*2/*G* follows ideology of document frequency (DF) in information retrieval. Hence, essentially HSMD measure is a *quasi-IDF*. Here we re-define the quasi-IDF as follows:

Note, notation “\” denotes complement operator in set theory. Similarly, following ideology of term frequency (TF), we define a so-called quasi-TF as follows:

Note, notation “” denotes intersection operator in set theory. The new measure called quasi-TF-IDF (QTI) is product of the quasi-TF and the quasi-IDF, according to equation 38.

|  |  |
| --- | --- |
|  | (38) |

QTI measure combines with Jaccard measure, which derives QTIJ measure according to equation 39.

|  |  |
| --- | --- |
|  | (39) |

Recall that Jaccard measure is mentioned in introduction section. Equation 39 is written in simple way as follows:

Where,

Given two rating vectors *u*1 = (*r*11=2, *r*12=5, *r*13=7, *r*14=8, *r*15=?, *r*16=9) and *u*2 = (*r*21=9, *r*22=?, *r*23=?, *r*24=6, *r*25=5, *r*26=1), we calculate QTIJ as an example. Of course, we have *I*1 = {1, 2, 3, 4, 6} and *I*2 = {1, 4, 5, 6}. We also have:

This implies:

Jaccard measure, which is an effective similarity measures, focuses on whether items are rated but it does not concern magnitude rating values like other measures. We overcome this drawback by proposing a so-called numerical nearby measure (NNMS) which concerns magnitude rating values and keeps strong point of Jaccard measure. In other words, NNMS combines sums of rating values and cardinalities of item sets. Equation 40 specifies NNMS.

|  |  |
| --- | --- |
|  | (40) |

Note that |*I*1 ∩ *I*2| is the number of items rated by both user 1 and user 2, |*I*1| is the number of items rated by only user 1, and |*I*2| is the number of items rated by only user 2. It is easy to recognize that NNMS is an interesting advanced variant of cosine measure with support of Jaccard measure. However, NNMS is totally different from combination of cosine and Jaccard as CosineJ. Experimental section will mention evaluation of NNMS and CosineJ. Anyway, NNMS is simpler than CosineJ.

Given two rating vectors *u*1 = (*r*11=2, *r*12=5, *r*13=7, *r*14=8, *r*15=?, *r*16=9) and *u*2 = (*r*21=9, *r*22=?, *r*23=?, *r*24=6, *r*25=5, *r*26=1), we calculate NNMS as an example. Of course, we have *I*1 = {1, 2, 3, 4, 6} and *I*2 = {1, 4, 5, 6}. We also have:

Shortly, in this research, we propose four measure families such as SMD, HSMD, QTI, and NNMS. SMD family includes measures such as SMD and HSMD. QTI family includes measures such as QTI and QTIJ. NNMS family includes only NNMS measure. In previous research (Nguyen & Amer, 2019), we also proposed measures called TA, TAJ, TAN, and TANJ. In this research, we test our measures such as SMD, HSMD, QTI, QTIJ, NNMS, TA, TAJ, TAN, and TANJ. Here we describe shortly TA, TAJ, TAN, and TANJ.

Cosine measure is effective but it has a drawback that there may be two end points of two vectors which are far from each other according to Euclidean distance, but their cosine is high. This is negative effect of Euclidean distance which decreases accuracy of cosine similarity. Therefore, a so-called triangle area (TA) measure (Nguyen & Amer, 2019) is proposed as an improved version of cosine measure. Figure 1 illustrates TA measure.

A close up of a map

Description automatically generated

**Figure 1.** Triangle area (TA) measure with 0 ≤ *α* ≤ *π*/2

TA measure uses ratio of basic triangle area to whole triangle area as reinforced factor for Euclidean distance so that it can alleviate negative effect of Euclidean distance whereas it keeps simplicity and effectiveness of both cosine measure and Euclidean distance in making similarity of two vectors. TA is considered as an advanced cosine measure. TA is defined by equation 41 (Nguyen & Amer, 2019):

|  |  |
| --- | --- |
|  | (41) |

Where, as usual:

Let TAJ denote the combined measure which combines TA measure and Jaccard measure. TAJ measure is defined as follows:

|  |  |
| --- | --- |
|  | (42) |

Let *rm* be median of rating values, TA measure is normalized as TAN measure as follows:

|  |  |
| --- | --- |
|  | (43) |

By combined with Jaccard measure, TAN measure becomes TANJ measure as follows:

|  |  |
| --- | --- |
|  | (44) |

As a convention, TA family includes TA, TAJ, TAN, and TAJ. Hence, equation 41 is the key of TA family.

By default, all measures are calculated based on user-based rating matrix in which every vector is user rating vector. When user-based rating matrix is transposed into item-based rating matrix in which every vector is item rating vector, equations for these measures are not changed in semantics. NN algorithm for user-based rating matrix becomes user-based NN algorithm and NN algorithm for item-based rating matrix becomes item-based NN algorithm. By default, NN algorithm implies user-based NN algorithm if there is no additional explanation.

**3. Experimental design**

Dataset Movielens (GroupLens, 1998) is used for evaluation, which has 100,000 ratings from 943 users on 1682 movies (items). Every rating ranges from 1 to 5. In the experiments, dataset Movielens is divided into 5 folders and each folder includes training set and testing set. Training set and testing set in the same folder are disjoint sets. The ratio of testing set over the whole dataset depends on the testing parameter *r*. For instance, if *r* = 0.1, the testing set covers 10% the dataset, which means that the testing set has 10,000 = 10%\*100,000 ratings and of course the training set has 90,000 ratings. In our experimental design, parameter *r* has nine values 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. The smaller *r* is, the more accurate measures are because training set gets large if *r* gets small with note that NN algorithm is executed on training set.

Popular metrics to assess CF algorithms are mean absolute error (MAE), recall, and precision. These metrics will be mentioned later. Quality of a CF algorithm like NN algorithm depends on both estimation and recommendation. Estimation ability is ability to estimate or predict exactly missing values. Recommendation is ability to provide list of recommended items which is as suitable as possible to users. Hence, we do not follow previous researches to focus on recommendation tasks with metrics MAE, precision, and recall. Instead we divide our tests into two processes such as estimation and recommendation as follows:

* In estimation process, given tested vector *ut* = (*v*1=1, *v*2=2, *v*3=3) having three items, it is made empty as empty vector *u*’ = (*v*1=?, *v*2=?, *v*3=?) with missing values. Later, NN algorithm is applied with measures above into predicting (estimating) missing values. As a result, predictive vector (estimated vector) is obtained, for example, *up* = (*v*1=2, *v*2=3, *v*3=4) having three estimated items. By comparing *ut* and *up*, MAE metric is used to evaluate estimation process. For instance, MAE metric is (|2-1| + |3-2| + |4-3|) / 3 = 1.
* In recommendation process, given tested vector *ut* = (*v*1=1, *v*2=2, *v*3=3), NN algorithm is asked for providing a recommended list (recommended vector) of items. Supposed the recommended vector is *us* = (*v*2=5, *v*4=4, *v*3=4, *v*5=2) within rating range {1, 2, 3, 4, 5}. By comparing *ut* and *us*, precision metric and recall metric are calculated to evaluate recommendation process. For instance, given *ut* = (*v*1=1, *v*2=2, *v*3=3) and *us* = (*v*2=5, *v*4=4, *v*3=4, *v*5=2), precision is 0.25 and recall is 1. The way to calculate precision and recall will be described later.

Hence, different metrics (MAE, recall, precision) are used for different evaluation processes (estimation and recommendation). This independent evaluation allows use to test measures more objectively, in which estimation process focused on accuracy of CF algorithm and recommendation process focuses on quality of CF algorithm. In general, MAE is used for estimation whereas recall and precision are used for recommendation process.

It is necessary to describe metrics MAE, precision, and recall. MAE is calculated by equation 45 (Herlocker, Konstan, Terveen, & Riedl, 2004, pp. 20-21) in which *n* is the total number of estimated items while *vj*’ and *vj* are predictive rating and true rating of item *j*, respectively.

|  |  |
| --- | --- |
|  | (45) |

The smaller MAE is, the more accurate the measures are and so the better the algorithm is.

Precision and recall are quality metrics that measure quality of recommended list – how much the recommendation list reflects user’s preferences. The larger quality metric is, the better the algorithm is. An item is relevant if its rating is larger than average rating. For example, within rating range {1, 2, 3, 4, 5}, the average rating is 3 = (1 + 5)/2. An item is selective if it is recommended to users. Let *Nr* be the number of relevant items and let *Ns* be the number of selective items. Let *Nrs* be the number of items which are relevant and selective. According to equation 46, precision is the ratio of *Nrs* to *Ns* and recall is the ratio of *Nrs* to *Nr* (Herlocker, Konstan, Terveen, & Riedl, 2004, p. 23). In other words, precision is probability that selective item is relevant and recall is probability that relevant item is selective.

|  |  |
| --- | --- |
|  | (46) |

For example, given tested vector *ut* = (*v*1=1, *v*2=2, *v*3=3) and recommended vector *us* = (*v*2=5, *v*4=4, *v*3=4, *v*5=2), we have *Ns* = |*u*s| = 4. Because the vector *ut* has only one relevant item 3 (*v*3=3), we have *Nr* = 1. We also have *Nrs* = 1 because only one relevant item 3 exists in both *ut* and *us*. Shortly, we have Precision = *Nrs*/*Ns* = 1/4 = 0.25 and Recall = *Nrs*/*Nr* = 1/1 = 1.

The problem in recommendation is how to determine the number of recommended items denoted *C* which is the length of recommended vector. As a convention, *C* is called recommendation count. The count *C* cannot be too small or too large. If it is too small, evaluation is inaccurate. Otherwise, if it is too large, evaluation task will run slowly. Some researches set fixed number whereas other researches changed such number over some values such as 10, 20, and 100. We proposed a method to determine *C* based on dataset with purpose that *C* will be more accurate and objective. The proposed method is dynamic and takes advantages of a so-called sparse-relevant ratio. This ratio is the ratio of the count of relevant ratings to the count of cells with note that the count of cells is product of user number and item number, which is size of rating matrix. Recall that a relevant rating is larger than average rating and the count of cells is sum of the count of rating values and the count of missing values. Equation 47 specifies sparse-relevant ratio denoted *sr*.

|  |  |
| --- | --- |
| *sr* = the-count-of-relevant-ratings / (|***U***| \* |***V***|) | (47) |

Note, |***U***| is the number of users and |***V***| is the number of items. We calculate recommendation count *C* dynamically according to both dataset and each rating vector *ui*. Let *C*(*ui*) be the recommendation count for user *i*, which means that NN algorithms will recommend at least *C*(*ui*) items to user *i*. Equation 48 specifies *C*(*ui*).

|  |  |
| --- | --- |
|  | (48) |

Where *T* is the number of items with note that every item included in *T* is rated by at least one user. Of course, *T* is smaller than or equal to the number of users |***U***|. Note, |*Ii*| is the number of items rated by user *i*. The quantity |*Ii*| is not redundant because real recommendation systems always recommend a user items that she/he do not either know or rate yet. If |*Ii*| is too much smaller than *T* (|*Ii*| << T), *C*(*ui*) can be calculated as follows:

Recall that in our experiments, dataset Movielens is divided into 5 folders and each folder has one training set and one testing set. In equation 48, for each folder, *T* and sparse-relevant ratio *sr* are calculated on training set but |*Ii*| is determined on testing set, of course. For example, suppose one among 5 folders divided from Movielens has training set *d*1 and testing set *t*1. The number of users in *d*1 is 943 and the number of items in *d*1 is 1,584. Because every item in *d*1 is rated by at least one user, we have *T* = 1,584. Training set *d*1 has 50,000 rating values but only 27,712 rating values are relevant. So sparse-relevant ratio is *sr* = 27,712 / (943\*1584) ≈ 1.86%. Suppose it is necessary to make recommendation on user rating vector *u*12 (in testing set *t*1) which has 23 rating values. Hence, recommendation count for user 12 is *C*(*u*12) = 1.86% \* (1,584 – 23) ≈ 29.

**4. Results and discussions**

As a convention, we have measures and measure families as follows:

* SMD family includes measures: SMD, HSMD, and HSMDJ.
* QTI family includes measures: QTI and QTIJ.
* NNMS family has only one NNMS measure.
* Cosine family includes measures: cosine, COJ, CON, COD, CosineJ.
* Pearson family includes measures: Pearson, WPC, SPC, PearsonJ.
* MSD family includes measures: MSD and MSDJ.
* TA family includes measures: TA, TAJ, TAN, and TAJ.
* Individual measures are Jaccard, SRC, NHSM, BCF, SMTP, PC, PIP, CjacMD, TMJ, Feng, and Mu.

Our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are tested with all other measures. Item-based NN algorithm is better than user-based NN algorithm because one execution of user-based NN algorithm implies *n* executions of item-based NN algorithm in implementation where *n* is equal to the number of missing values. However, qualities of measures are independent from user-based one or item-based one. Moreover, user-based NN algorithm runs faster than item-based NN algorithm. Hence, we test all measures with user-based NN algorithm given user-based rating matrix. NN algorithm implies user-based NN algorithm and rating matrix implies user-based rating matrix. Recall that dataset Movielens (GroupLens, 1998) is used for evaluation, which is divided into 5 folders and each folder includes training set and testing set. Each folder has own tested measures and so tested measures shown here are made average over 5 folders.

Table 3 shows MAE metric of all tested measures about all *r* = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9 within estimation process. The last column shows average MAE metrics over all values of *r* and shaded cells indicate best values. As a convention, we define that preeminent measures (dominant measures) are ones in top-5 lists.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *r*=0.1 | *r*=0.2 | *r*=0.3 | *r*=0.4 | *r*=0.5 | *r*=0.6 | *r*=0.7 | *r*=0.8 | *r*=0.9 | Average  (MAE) |
| Cosine | 0.7532 | 0.7551 | 0.7560 | 0.7593 | 0.7630 | 0.7654 | 0.7736 | 0.7905 | 0.8255 | 0.7713 |
| COJ | 0.7457 | 0.7483 | 0.7494 | 0.7533 | 0.7572 | 0.7610 | 0.7710 | 0.7937 | 0.8681 | 0.7720 |
| CON | 0.7469 | 0.7503 | 0.7532 | 0.7582 | 0.7657 | 0.7738 | 0.7889 | 0.8227 | 0.8985 | 0.7842 |
| COD | 0.8224 | 0.8271 | 0.8307 | 0.8357 | 0.8441 | 0.8526 | 0.8621 | 0.8821 | 0.9423 | 0.8555 |
| CosineJ | 0.7459 | 0.7485 | 0.7496 | 0.7537 | 0.7577 | 0.7615 | 0.7712 | 0.7921 | 0.8537 | 0.7704 |
| Pearson | 0.7395 | 0.7462 | 0.7519 | 0.7611 | 0.7734 | 0.7882 | 0.8091 | 0.8435 | 0.8473 | 0.7845 |
| WPC | 0.7312 | 0.7365 | 0.7405 | 0.7480 | 0.7581 | 0.7721 | 0.7947 | 0.8337 | 0.9202 | 0.7817 |
| SPC | 0.7388 | 0.7452 | 0.7506 | 0.7592 | 0.7708 | 0.7848 | 0.8051 | 0.8399 | 0.8490 | 0.7826 |
| PearsonJ | 0.7311 | 0.7375 | 0.7427 | 0.7510 | 0.7624 | 0.7766 | 0.7992 | 0.8379 | 0.9173 | 0.7840 |
| Jaccard | 0.7465 | 0.7491 | 0.7502 | 0.7543 | 0.7583 | 0.7620 | 0.7717 | 0.7939 | 0.8651 | 0.7723 |
| MSD | 0.7529 | 0.7549 | 0.7558 | 0.7591 | 0.7627 | 0.7651 | 0.7732 | 0.7901 | 0.8471 | 0.7734 |
| MSDJ | 0.7457 | 0.7484 | 0.7495 | 0.7536 | 0.7575 | 0.7613 | 0.7709 | 0.7919 | 0.8538 | 0.7703 |
| SRC | 0.7429 | 0.7434 | 0.7426 | 0.7453 | 0.7536 | 0.7682 | 0.8127 | 0.9394 | 1.1041 | 0.8169 |
| NHSM | 0.7410 | 0.7441 | 0.7452 | 0.7498 | 0.7545 | 0.7599 | 0.7728 | 0.8006 | 0.8729 | 0.7712 |
| BCF | 0.7984 | 0.8004 | 0.8011 | 0.8035 | 0.8061 | 0.8095 | 0.8159 | 0.8316 | 0.8658 | 0.8147 |
| SMTP | 0.7533 | 0.7551 | 0.7560 | 0.7592 | 0.7629 | 0.7652 | 0.7733 | 0.7903 | 0.8478 | 0.7737 |
| PC | 0.8229 | 0.8279 | 0.8317 | 0.8365 | 0.8446 | 0.8522 | 0.8612 | 0.8805 | 0.9443 | 0.8558 |
| PIP | 0.7424 | 0.7451 | 0.7466 | 0.7510 | 0.7556 | 0.7606 | 0.7726 | 0.7989 | 0.8723 | 0.7717 |
| CjacMD | 0.7804 | 0.7885 | 0.7946 | 0.8054 | 0.8193 | 0.8310 | 0.8554 | 0.8975 | 0.9795 | 0.8391 |
| TMJ | 0.7463 | 0.7489 | 0.7500 | 0.7541 | 0.7581 | 0.7618 | 0.7714 | 0.7935 | 0.8647 | 0.7721 |
| Feng | 0.7454 | 0.7479 | 0.7489 | 0.7528 | 0.7566 | 0.7605 | 0.7704 | 0.7928 | 0.8673 | 0.7714 |
| Mu | 0.7388 | 0.7425 | 0.7456 | 0.7517 | 0.7612 | 0.7725 | 0.7917 | 0.8262 | 0.9027 | 0.7814 |
| TA | 0.7518 | 0.7538 | 0.7547 | 0.7581 | 0.7618 | 0.7643 | 0.7726 | 0.7901 | 0.8487 | 0.7729 |
| TAJ | 0.7449 | 0.7475 | 0.7486 | 0.7527 | 0.7568 | 0.7606 | 0.7704 | 0.7920 | 0.8552 | 0.7699 |
| TAN | 0.7467 | 0.7503 | 0.7527 | 0.7586 | 0.7654 | 0.7713 | 0.7846 | 0.8084 | 0.8723 | 0.7789 |
| TANJ | 0.7379 | 0.7416 | 0.7439 | 0.7501 | 0.7568 | 0.7633 | 0.7779 | 0.8050 | 0.8734 | 0.7722 |
| SMD | 0.7524 | 0.7546 | 0.7557 | 0.7592 | 0.7631 | 0.7657 | 0.7737 | 0.7885 | 0.8253 | 0.7709 |
| HSMD | 0.7481 | 0.7505 | 0.7514 | 0.7550 | 0.7588 | 0.7622 | 0.7719 | 0.7944 | 0.8812 | 0.7748 |
| HSMDJ | 0.7427 | 0.7458 | 0.7470 | 0.7516 | 0.7562 | 0.7614 | 0.7739 | 0.8017 | 0.8791 | 0.7733 |
| QTI | 0.7396 | 0.7433 | 0.7448 | 0.7501 | 0.7557 | 0.7630 | 0.7796 | 0.8148 | 0.8983 | 0.7766 |
| QTIJ | 0.7375 | 0.7422 | 0.7443 | 0.7509 | 0.7581 | 0.7681 | 0.7894 | 0.8306 | 0.9125 | 0.7815 |
| NNMS | 0.7420 | 0.7453 | 0.7466 | 0.7517 | 0.7567 | 0.7626 | 0.7761 | 0.8056 | 0.8837 | 0.7745 |

**Table 3.** MAE metric within estimation process

Top-5 measures according to MAE metric within estimation process are TAJ, MSDJ, CosineJ, SMD, and NHSM whose average MAE metrics are 0.7699, 0.7703, 0.7704, 0.7709, and 0.7712, respectively. Our SMD measure is in the top-5 list given MAE metric. Shortly, dominant orders of our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are 13rd, 1st, 20th, 11st, 4th, 18th, 14th, 19th, 22nd, and 17th among 32 measures given MAE metric respectively.

Table 4 shows precision metric of all tested measures about all *r* = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9 within recommendation process given precision metric. The last column shows average precision metrics over all values of *r* and shaded cells indicate best values.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *r*=0.1 | *r*=0.2 | *r*=0.3 | *r*=0.4 | *r*=0.5 | *r*=0.6 | *r*=0.7 | *r*=0.8 | *r*=0.9 | Average  (Precision) |
| Cosine | 0.0055 | 0.0104 | 0.0154 | 0.0207 | 0.0262 | 0.0324 | 0.0396 | 0.0508 | 0.0836 | 0.0316 |
| COJ | 0.0056 | 0.0105 | 0.0156 | 0.0207 | 0.0262 | 0.0318 | 0.0377 | 0.0439 | 0.0510 | 0.0270 |
| CON | 0.0054 | 0.0100 | 0.0148 | 0.0197 | 0.0250 | 0.0310 | 0.0384 | 0.0512 | 0.0912 | 0.0319 |
| COD | 0.0046 | 0.0086 | 0.0128 | 0.0172 | 0.0220 | 0.0273 | 0.0339 | 0.0443 | 0.0755 | 0.0274 |
| CosineJ | 0.0056 | 0.0105 | 0.0156 | 0.0209 | 0.0265 | 0.0327 | 0.0399 | 0.0510 | 0.0835 | 0.0318 |
| Pearson | 0.0051 | 0.0095 | 0.0141 | 0.0187 | 0.0237 | 0.0291 | 0.0359 | 0.0467 | 0.0803 | 0.0292 |
| WPC | 0.0052 | 0.0097 | 0.0144 | 0.0190 | 0.0241 | 0.0295 | 0.0363 | 0.0471 | 0.0804 | 0.0295 |
| SPC | 0.0051 | 0.0096 | 0.0141 | 0.0187 | 0.0238 | 0.0292 | 0.0359 | 0.0468 | 0.0804 | 0.0293 |
| PearsonJ | 0.0052 | 0.0097 | 0.0143 | 0.0190 | 0.0240 | 0.0295 | 0.0362 | 0.0471 | 0.0804 | 0.0295 |
| Jaccard | 0.0056 | 0.0105 | 0.0155 | 0.0207 | 0.0261 | 0.0317 | 0.0377 | 0.0438 | 0.0511 | 0.0270 |
| MSD | 0.0055 | 0.0104 | 0.0155 | 0.0207 | 0.0262 | 0.0324 | 0.0396 | 0.0508 | 0.0836 | 0.0316 |
| MSDJ | 0.0056 | 0.0105 | 0.0156 | 0.0209 | 0.0265 | 0.0327 | 0.0399 | 0.0510 | 0.0835 | 0.0318 |
| SRC | 0.0061 | 0.0116 | 0.0168 | 0.0213 | 0.0243 | 0.0255 | 0.0234 | 0.0199 | 0.0362 | 0.0206 |
| NHSM | 0.0057 | 0.0107 | 0.0158 | 0.0211 | 0.0268 | 0.0331 | 0.0402 | 0.0512 | 0.0829 | 0.0319 |
| BCF | 0.0047 | 0.0089 | 0.0133 | 0.0179 | 0.0230 | 0.0282 | 0.0341 | 0.0412 | 0.0518 | 0.0248 |
| SMTP | 0.0055 | 0.0104 | 0.0154 | 0.0207 | 0.0262 | 0.0324 | 0.0396 | 0.0508 | 0.0836 | 0.0316 |
| PC | 0.0050 | 0.0094 | 0.0140 | 0.0189 | 0.0244 | 0.0304 | 0.0379 | 0.0498 | 0.0854 | 0.0306 |
| PIP | 0.0057 | 0.0107 | 0.0158 | 0.0211 | 0.0268 | 0.0331 | 0.0402 | 0.0513 | 0.0833 | 0.0320 |
| CjacMD | 0.0054 | 0.0100 | 0.0147 | 0.0194 | 0.0243 | 0.0291 | 0.0339 | 0.0389 | 0.0453 | 0.0246 |
| TMJ | 0.0056 | 0.0105 | 0.0155 | 0.0207 | 0.0261 | 0.0317 | 0.0377 | 0.0438 | 0.0511 | 0.0270 |
| Feng | 0.0056 | 0.0105 | 0.0156 | 0.0207 | 0.0262 | 0.0318 | 0.0377 | 0.0439 | 0.0510 | 0.0270 |
| Mu | 0.0053 | 0.0099 | 0.0146 | 0.0194 | 0.0245 | 0.0300 | 0.0367 | 0.0476 | 0.0814 | 0.0299 |
| TA | 0.0055 | 0.0104 | 0.0155 | 0.0207 | 0.0263 | 0.0325 | 0.0397 | 0.0509 | 0.0836 | 0.0317 |
| TAJ | 0.0056 | 0.0106 | 0.0157 | 0.0209 | 0.0265 | 0.0328 | 0.0400 | 0.0511 | 0.0835 | 0.0319 |
| TAN | 0.0053 | 0.0099 | 0.0146 | 0.0194 | 0.0248 | 0.0307 | 0.0381 | 0.0510 | 0.0909 | 0.0316 |
| TANJ | 0.0054 | 0.0101 | 0.0148 | 0.0197 | 0.0251 | 0.0311 | 0.0385 | 0.0512 | 0.0909 | 0.0319 |
| SMD | 0.0055 | 0.0104 | 0.0154 | 0.0205 | 0.0259 | 0.0316 | 0.0377 | 0.0448 | 0.0546 | 0.0274 |
| HSMD | 0.0056 | 0.0105 | 0.0155 | 0.0207 | 0.0261 | 0.0318 | 0.0376 | 0.0437 | 0.0499 | 0.0268 |
| HSMDJ | 0.0057 | 0.0106 | 0.0157 | 0.0208 | 0.0263 | 0.0320 | 0.0379 | 0.0439 | 0.0507 | 0.0271 |
| QTI | 0.0057 | 0.0107 | 0.0158 | 0.0210 | 0.0265 | 0.0322 | 0.0380 | 0.0438 | 0.0501 | 0.0271 |
| QTIJ | 0.0058 | 0.0108 | 0.0159 | 0.0211 | 0.0266 | 0.0323 | 0.0381 | 0.0436 | 0.0498 | 0.0271 |
| NNMS | 0.0057 | 0.0106 | 0.0157 | 0.0209 | 0.0264 | 0.0321 | 0.0380 | 0.0440 | 0.0506 | 0.0271 |

**Table 4.** Precision metric within recommendation process

Top-5 measures according to precision metric within recommendation process are PIP, NHSM, TANJ, CON, TAJ whose average precision metrics are 0.0320, 0.0319, 0.0319, 0.0319, and 0.0319, respectively. Our measures TANJ and TAJ are in the top-5 list given precision metric. Shortly, dominant orders of our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are 8th, 5th, 10th, 3rd, 19th, 29th, 24th, 23rd, 22nd, and 21st among 32 measures given precision metric, respectively.

Table 5 shows recall metric of all tested measures about all *r* = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9 within recommendation process given recall metric. The last column shows average recall metrics over all values of *r* and shaded cells indicate best values.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *r*=0.1 | *r*=0.2 | *r*=0.3 | *r*=0.4 | *r*=0.5 | *r*=0.6 | *r*=0.7 | *r*=0.8 | *r*=0.9 | Average  (Recall) |
| Cosine | 0.9241 | 0.9208 | 0.9211 | 0.9177 | 0.9150 | 0.9147 | 0.9066 | 0.8937 | 0.8021 | 0.9018 |
| COJ | 0.9265 | 0.9235 | 0.9230 | 0.9199 | 0.9163 | 0.9157 | 0.9067 | 0.8942 | 0.8462 | 0.9080 |
| CON | 0.9313 | 0.9269 | 0.9269 | 0.9233 | 0.9201 | 0.9184 | 0.9079 | 0.8814 | 0.7327 | 0.8965 |
| COD | 0.9452 | 0.9416 | 0.9394 | 0.9340 | 0.9293 | 0.9269 | 0.9133 | 0.8872 | 0.7690 | 0.9095 |
| CosineJ | 0.9266 | 0.9232 | 0.9223 | 0.9193 | 0.9159 | 0.9153 | 0.9066 | 0.8914 | 0.7970 | 0.9020 |
| Pearson | 0.9439 | 0.9402 | 0.9388 | 0.9359 | 0.9331 | 0.9309 | 0.9190 | 0.8948 | 0.7834 | 0.9133 |
| WPC | 0.9430 | 0.9387 | 0.9373 | 0.9349 | 0.9331 | 0.9315 | 0.9197 | 0.8967 | 0.7836 | 0.9132 |
| SPC | 0.9443 | 0.9402 | 0.9390 | 0.9362 | 0.9335 | 0.9316 | 0.9196 | 0.8960 | 0.7840 | 0.9138 |
| PearsonJ | 0.9429 | 0.9440 | 0.9373 | 0.9351 | 0.9323 | 0.9309 | 0.9186 | 0.8948 | 0.7814 | 0.9130 |
| Jaccard | 0.9266 | 0.9230 | 0.9221 | 0.9191 | 0.9158 | 0.9155 | 0.9073 | 0.8947 | 0.8496 | 0.9082 |
| MSD | 0.9242 | 0.9209 | 0.9210 | 0.9178 | 0.9151 | 0.9147 | 0.9067 | 0.8938 | 0.8020 | 0.9018 |
| MSDJ | 0.9267 | 0.9231 | 0.9223 | 0.9194 | 0.9159 | 0.9154 | 0.9067 | 0.8914 | 0.7969 | 0.9020 |
| SRC | 0.7401 | 0.6827 | 0.6167 | 0.5303 | 0.4208 | 0.3010 | 0.1803 | 0.0825 | 0.0489 | 0.4004 |
| NHSM | 0.9283 | 0.9238 | 0.9234 | 0.9191 | 0.9159 | 0.9142 | 0.9037 | 0.8845 | 0.7842 | 0.8997 |
| BCF | 0.9124 | 0.9106 | 0.9098 | 0.9027 | 0.9032 | 0.9007 | 0.8927 | 0.8790 | 0.8495 | 0.8956 |
| SMTP | 0.9242 | 0.9209 | 0.9209 | 0.9177 | 0.9151 | 0.9148 | 0.9065 | 0.8936 | 0.8016 | 0.9017 |
| PC | 0.9132 | 0.9115 | 0.9117 | 0.9086 | 0.9080 | 0.9096 | 0.8987 | 0.8753 | 0.7456 | 0.8869 |
| PIP | 0.9284 | 0.9259 | 0.9257 | 0.9211 | 0.9183 | 0.9171 | 0.9068 | 0.8887 | 0.7882 | 0.9022 |
| CjacMD | 0.9187 | 0.9130 | 0.9102 | 0.9029 | 0.8971 | 0.8940 | 0.8779 | 0.8551 | 0.8009 | 0.8855 |
| TMJ | 0.9264 | 0.9228 | 0.9221 | 0.9191 | 0.9158 | 0.9155 | 0.9073 | 0.8948 | 0.8498 | 0.9082 |
| Feng | 0.9266 | 0.9237 | 0.9232 | 0.9199 | 0.9164 | 0.9156 | 0.9070 | 0.8944 | 0.8469 | 0.9082 |
| Mu | 0.9354 | 0.9320 | 0.9326 | 0.9298 | 0.9282 | 0.9278 | 0.9184 | 0.8981 | 0.7899 | 0.9102 |
| TA | 0.9242 | 0.9211 | 0.9211 | 0.9177 | 0.9149 | 0.9145 | 0.9060 | 0.8928 | 0.8005 | 0.9014 |
| TAJ | 0.9265 | 0.9229 | 0.9224 | 0.9191 | 0.9154 | 0.9149 | 0.9060 | 0.8907 | 0.7957 | 0.9015 |
| TAN | 0.9303 | 0.9272 | 0.9262 | 0.9216 | 0.9178 | 0.9156 | 0.9025 | 0.8747 | 0.7277 | 0.8937 |
| TANJ | 0.9300 | 0.9267 | 0.9256 | 0.9210 | 0.9176 | 0.9154 | 0.9026 | 0.8744 | 0.7254 | 0.8932 |
| SMD | 0.9244 | 0.9212 | 0.9210 | 0.9180 | 0.9148 | 0.9152 | 0.9078 | 0.9000 | 0.8718 | 0.9105 |
| HSMD | 0.9262 | 0.9228 | 0.9224 | 0.9190 | 0.9161 | 0.9158 | 0.9070 | 0.8948 | 0.8479 | 0.9080 |
| HSMDJ | 0.9284 | 0.9244 | 0.9238 | 0.9199 | 0.9162 | 0.9152 | 0.9049 | 0.8888 | 0.8391 | 0.9067 |
| QTI | 0.9295 | 0.9261 | 0.9253 | 0.9203 | 0.9160 | 0.9135 | 0.9005 | 0.8796 | 0.8255 | 0.9040 |
| QTIJ | 0.9296 | 0.9261 | 0.9244 | 0.9193 | 0.9137 | 0.9098 | 0.8944 | 0.8696 | 0.8171 | 0.9004 |
| NNMS | 0.9272 | 0.9245 | 0.9231 | 0.9189 | 0.9148 | 0.9133 | 0.9024 | 0.8854 | 0.8351 | 0.9050 |

**Table 5.** Recall metric within recommendation process

Top-5 measures according to recall metric within recommendation process are SPC, Pearson, WPC, PearsonJ, and SMD whose average recall metrics are 0.9138, 0.9133, 0.9132, 0.9130, and 0.9105, respectively. Our SMD measure is in the top-5 list given recall metric. It is interesting that three of top-5 list given recall metric are SPC, Pearson, WPC, and PearsonJ which belong to Pearson family. Hence, Pearson family is preeminent over recall metric. Shortly, dominant orders of our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are 23rd, 22nd, 28th, 29th, 5th, 12nd, 13rd, 15th, 24th, and 14th among 32 measures given recall metric, respectively.

From metrics MAE, precision, and recall shown in tables 3, 4, 5, respectively, it is not easy to determine which measures are the best. Remind that estimation process is evaluated by MAE metric and recommendation process is evaluated by both precision metric and recall metric. F1 metric is the way to assembling precision and recall together. Equation 49 (Herlocker, Konstan, Terveen, & Riedl, 2004, p. 25) specifies F1 metric. The larger F1 is, the better measures are.

|  |  |
| --- | --- |
|  | (49) |

For example, given cosine measure from tables 4 and 5, its average precision and recall are 0.0316 and 0.9018 and hence, F1 metric of cosine is:

Shortly, MAE is used to evaluate estimation process and F1 is used to evaluate recommendation process. Table 6 which is derived from tables 3, 4, and 5 shows average MAE values and F1 values of all measures. Shaded cells indicate best values.

|  |  |  |
| --- | --- | --- |
|  | MAE | F1 |
| Cosine | 0.7713 | 0.030551 |
| COJ | 0.7720 | 0.026220 |
| CON | 0.7842 | 0.030763 |
| COD | 0.8555 | 0.026557 |
| CosineJ | 0.7704 | 0.030717 |
| Pearson | 0.7845 | 0.028327 |
| WPC | 0.7817 | 0.028598 |
| SPC | 0.7826 | 0.028379 |
| PearsonJ | 0.7840 | 0.028566 |
| Jaccard | 0.7723 | 0.026189 |
| MSD | 0.7734 | 0.030561 |
| MSDJ | 0.7703 | 0.030717 |
| SRC | 0.8169 | 0.019562 |
| NHSM | 0.7712 | 0.030849 |
| BCF | 0.8147 | 0.024121 |
| SMTP | 0.7737 | 0.030551 |
| PC | 0.8558 | 0.029559 |
| PIP | 0.7717 | 0.030904 |
| CjacMD | 0.8391 | 0.023893 |
| TMJ | 0.7721 | 0.026189 |
| Feng | 0.7714 | 0.026220 |
| Mu | 0.7814 | 0.028980 |
| TA | 0.7729 | 0.030602 |
| TAJ | 0.7699 | 0.030768 |
| TAN | 0.7789 | 0.030552 |
| TANJ | 0.7722 | 0.030769 |
| SMD | 0.7709 | 0.026579 |
| HSMD | 0.7748 | 0.026053 |
| HSMDJ | 0.7733 | 0.026282 |
| QTI | 0.7766 | 0.026301 |
| QTIJ | 0.7815 | 0.026319 |
| NNMS | 0.7745 | 0.026323 |

**Table 6.** General MAE and F1 over all measures

Top-5 measures according to F1 metric within recommendation process are PIP, NHSM, TANJ, TAJ, and CON whose average recall metrics are 0.030904, 0.030849, 0.030769, 0.030768, and 0.030763. Shortly, dominant orders of our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are 8th, 4th, 10th, 3rd, 19th, 29th, 24th, 23rd, 22nd, and 21st among 32 measures given F1 metric, respectively.

Figure 2 which is extracted from table 6 draws comparison over all measures regarding MAE metric.

A screenshot of a cell phone

Description automatically generated

**Figure 2.** Measure comparison with MAE metric

Figure 3 which is extracted from table 6 draws comparison over all measures regarding F1 metric.

A close up of a building

Description automatically generated

**Figure 3.** Measure comparison with F1 metric

It is easy to recognize that table 6 is the general evaluation of all measures regarding estimation process and recommendation process. We cannot unify MAE and F1 as we unify precision and recall because estimation process and recommendation process are not always proportional. Therefore, let *A* be set of top-5 measures regarding MAE and let *B* be set of top-5 measures regarding F1. The intersection of *A* and *B* contains best measures. From table 6, we have *A* = {TAJ, MSDJ, CosineJ, SMD, NHSM} and *B* = {PIP, NHSM, TANJ, TAJ, CON}. Obviously, *best measures in general comparison are TAJ and NHSM*.

Although it is totally possible to evaluate measures with two metrics MAE and F1, it is better to go further with other metrics. For estimation process, some popular metrics which are different from MAE are mean squared error (MSE) and correlation coefficient (R). MSE is calculated by equation 50 (Herlocker, Konstan, Terveen, & Riedl, 2004, p. 21) in which *n* is the total number of estimated items while *vj*’ and *vj* are predictive rating and true rating of item *j*, respectively. Of course, we have predictive vector *v*’ and true vector (tested vector) *v*.

|  |  |
| --- | --- |
|  | (50) |

The smaller MSE is, the more accurate the measures are and so the better the algorithm is. R is used to evaluate correlation between predictive vector *v*’ and true vector *v*. It is really Pearson correlation. The larger R is, the better the measures are. Equation 51 (Montgomery & Runger, 2010, p. 432) specifies R metric.

|  |  |
| --- | --- |
|  | (51) |

Where and are mean values of tested item and predictive item, respectively.

Table 7 shows MSE metric of all tested measures about all values of *r*. Shaded cells indicate best values.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *r*=0.1 | *r*=0.2 | *r*=0.3 | *r*=0.4 | *r*=0.5 | *r*=0.6 | *r*=0.7 | *r*=0.8 | *r*=0.9 | Average  (MSE) |
| Cosine | 0.9114 | 0.9193 | 0.9235 | 0.9304 | 0.9394 | 0.9476 | 0.9681 | 1.0162 | 1.1135 | 0.9633 |
| COJ | 0.8971 | 0.9060 | 0.9104 | 0.9184 | 0.9285 | 0.9390 | 0.9638 | 1.0255 | 1.2393 | 0.9698 |
| CON | 0.8947 | 0.9050 | 0.9124 | 0.9245 | 0.9427 | 0.9632 | 1.0019 | 1.0939 | 1.3155 | 0.9949 |
| COD | 1.0676 | 1.0800 | 1.0898 | 1.1031 | 1.1233 | 1.1459 | 1.1739 | 1.2349 | 1.4165 | 1.1594 |
| CosineJ | 0.8973 | 0.9064 | 0.9111 | 0.9193 | 0.9294 | 0.9400 | 0.9643 | 1.0209 | 1.1956 | 0.9649 |
| Pearson | 0.8810 | 0.8970 | 0.9107 | 0.9304 | 0.9590 | 0.9930 | 1.0438 | 1.1355 | 1.1640 | 0.9905 |
| WPC | 0.8650 | 0.8786 | 0.8892 | 0.9048 | 0.9289 | 0.9600 | 1.0137 | 1.1146 | 1.3721 | 0.9919 |
| SPC | 0.8792 | 0.8947 | 0.9076 | 0.9261 | 0.9531 | 0.9851 | 1.0344 | 1.1267 | 1.1678 | 0.9861 |
| PearsonJ | 0.8641 | 0.8798 | 0.8922 | 0.9097 | 0.9362 | 0.9680 | 1.0224 | 1.1235 | 1.3499 | 0.9940 |
| Jaccard | 0.8987 | 0.9078 | 0.9125 | 0.9208 | 0.9308 | 0.9415 | 0.9658 | 1.0262 | 1.2313 | 0.9706 |
| MSD | 0.9109 | 0.9190 | 0.9231 | 0.9300 | 0.9388 | 0.9468 | 0.9671 | 1.0150 | 1.1794 | 0.9700 |
| MSDJ | 0.8971 | 0.9062 | 0.9108 | 0.9190 | 0.9291 | 0.9396 | 0.9638 | 1.0206 | 1.1959 | 0.9647 |
| SRC | 0.8871 | 0.8872 | 0.8866 | 0.8933 | 0.9150 | 0.9572 | 1.0799 | 1.4585 | 1.9472 | 1.1013 |
| NHSM | 0.8878 | 0.8977 | 0.9025 | 0.9120 | 0.9239 | 0.9383 | 0.9709 | 1.0444 | 1.2484 | 0.9695 |
| BCF | 1.0126 | 1.0222 | 1.0260 | 1.0332 | 1.0405 | 1.0518 | 1.0709 | 1.1165 | 1.2249 | 1.0665 |
| SMTP | 0.9116 | 0.9195 | 0.9237 | 0.9304 | 0.9394 | 0.9473 | 0.9677 | 1.0158 | 1.1812 | 0.9707 |
| PC | 1.0813 | 1.0936 | 1.1030 | 1.1138 | 1.1324 | 1.1505 | 1.1741 | 1.2295 | 1.4196 | 1.1664 |
| PIP | 0.8940 | 0.9031 | 0.9085 | 0.9174 | 0.9300 | 0.9432 | 0.9740 | 1.0444 | 1.2526 | 0.9741 |
| CjacMD | 0.9645 | 0.9871 | 1.0048 | 1.0299 | 1.0660 | 1.0966 | 1.1603 | 1.2773 | 1.5198 | 1.1229 |
| TMJ | 0.8983 | 0.9073 | 0.9120 | 0.9202 | 0.9303 | 0.9408 | 0.9650 | 1.0250 | 1.2299 | 0.9699 |
| Feng | 0.8962 | 0.9048 | 0.9091 | 0.9171 | 0.9269 | 0.9376 | 0.9626 | 1.0237 | 1.2380 | 0.9684 |
| Mu | 0.8812 | 0.8920 | 0.8998 | 0.9121 | 0.9337 | 0.9596 | 1.0044 | 1.0949 | 1.3152 | 0.9881 |
| TA | 0.9085 | 0.9164 | 0.9205 | 0.9276 | 0.9367 | 0.9449 | 0.9657 | 1.0151 | 1.1833 | 0.9687 |
| TAJ | 0.8952 | 0.9041 | 0.9087 | 0.9170 | 0.9274 | 0.9380 | 0.9627 | 1.0206 | 1.1993 | 0.9637 |
| TAN | 0.8981 | 0.9086 | 0.9168 | 0.9305 | 0.9470 | 0.9634 | 0.9967 | 1.0643 | 1.2514 | 0.9863 |
| TANJ | 0.8811 | 0.8911 | 0.8985 | 0.9127 | 0.9288 | 0.9455 | 0.9819 | 1.0556 | 1.2523 | 0.9719 |
| SMD | 0.9102 | 0.9187 | 0.9231 | 0.9304 | 0.9396 | 0.9475 | 0.9673 | 1.0064 | 1.1131 | 0.9618 |
| HSMD | 0.9023 | 0.9107 | 0.9151 | 0.9225 | 0.9320 | 0.9417 | 0.9663 | 1.0291 | 1.2821 | 0.9780 |
| HSMDJ | 0.8922 | 0.9022 | 0.9072 | 0.9166 | 0.9282 | 0.9423 | 0.9735 | 1.0475 | 1.2688 | 0.9754 |
| QTI | 0.8877 | 0.8987 | 0.9045 | 0.9158 | 0.9296 | 0.9489 | 0.9903 | 1.0833 | 1.3227 | 0.9868 |
| QTIJ | 0.8847 | 0.8979 | 0.9053 | 0.9195 | 0.9375 | 0.9639 | 1.0177 | 1.1276 | 1.3643 | 1.0020 |
| NNMS | 0.8903 | 0.9006 | 0.9058 | 0.9165 | 0.9291 | 0.9447 | 0.9788 | 1.0581 | 1.2820 | 0.9784 |

**Table 7.** MSE metric within estimation process

Top-5 measures according to MSE metric within recommendation process are SMD, Cosine, TAJ, MSDJ, and CosineJ whose average recall metrics are 0.9618, 0.9633, 0.9637, 0.9647, and 0.9649. Our SMD measure is in the top-5 list given MSE metric. Shortly, dominant orders of our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are 7th, 3rd, 20th, 14th, 1st, 17th, 16th, 21st, 27th, and 18th among 32 measures given MSE metric, respectively.

Table 8 shows R metric of all tested measures about all values of *r*. Shaded cells indicate best values.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | *r*=0.1 | *r*=0.2 | *r*=0.3 | *r*=0.4 | *r*=0.5 | *r*=0.6 | *r*=0.7 | *r*=0.8 | *r*=0.9 | Average  (R) |
| Cosine | 0.4185 | 0.3780 | 0.3829 | 0.3806 | 0.3784 | 0.3736 | 0.3580 | 0.3261 | 0.3004 | 0.3663 |
| COJ | 0.3881 | 0.3880 | 0.3915 | 0.3883 | 0.3855 | 0.3771 | 0.3571 | 0.3169 | 0.2051 | 0.3553 |
| CON | 0.4339 | 0.3865 | 0.3851 | 0.3741 | 0.3525 | 0.3265 | 0.2879 | 0.2204 | 0.1403 | 0.3230 |
| COD | 0.1810 | 0.1147 | 0.1087 | 0.0911 | 0.0678 | 0.0537 | 0.0418 | 0.0288 | 0.0178 | 0.0784 |
| CosineJ | 0.3871 | 0.3856 | 0.3885 | 0.3843 | 0.3817 | 0.3733 | 0.3531 | 0.3145 | 0.2228 | 0.3545 |
| Pearson | 0.4355 | 0.3699 | 0.3557 | 0.3260 | 0.2905 | 0.2460 | 0.1931 | 0.1250 | 0.2530 | 0.2883 |
| WPC | 0.4519 | 0.3939 | 0.3821 | 0.3597 | 0.3265 | 0.2828 | 0.2236 | 0.1448 | 0.0569 | 0.2914 |
| SPC | 0.4378 | 0.3749 | 0.3620 | 0.3352 | 0.3013 | 0.2574 | 0.2031 | 0.1314 | 0.2476 | 0.2945 |
| PearsonJ | 0.4109 | 0.3823 | 0.3668 | 0.3424 | 0.3068 | 0.2648 | 0.2070 | 0.1342 | 0.0578 | 0.2748 |
| Jaccard | 0.4273 | 0.3841 | 0.3870 | 0.3828 | 0.3803 | 0.3719 | 0.3516 | 0.3120 | 0.2058 | 0.3559 |
| MSD | 0.4192 | 0.3789 | 0.3837 | 0.3813 | 0.3793 | 0.3750 | 0.3596 | 0.3274 | 0.2335 | 0.3598 |
| MSDJ | 0.4288 | 0.3859 | 0.3891 | 0.3847 | 0.3822 | 0.3741 | 0.3539 | 0.3150 | 0.2248 | 0.3598 |
| SRC | 0.3119 | 0.2985 | 0.2915 | 0.2621 | 0.2143 | 0.1676 | 0.0842 | 0.0078 | -0.0425 | 0.1773 |
| NHSM | 0.4343 | 0.3928 | 0.3951 | 0.3905 | 0.3841 | 0.3711 | 0.3449 | 0.2981 | 0.2121 | 0.3581 |
| BCF | 0.3073 | 0.2636 | 0.2607 | 0.2638 | 0.2530 | 0.2553 | 0.2400 | 0.2224 | 0.1897 | 0.2506 |
| SMTP | 0.4185 | 0.3782 | 0.3833 | 0.3810 | 0.3789 | 0.3745 | 0.3589 | 0.3268 | 0.2328 | 0.3592 |
| PC | 0.1422 | 0.1300 | 0.1192 | 0.1004 | 0.0765 | 0.0609 | 0.0460 | 0.0309 | 0.0128 | 0.0799 |
| PIP | 0.3939 | 0.3913 | 0.3980 | 0.3940 | 0.3870 | 0.3768 | 0.3529 | 0.3089 | 0.2128 | 0.3573 |
| CjacMD | 0.2796 | 0.2561 | 0.2302 | 0.2007 | 0.1547 | 0.1157 | 0.0492 | -0.0346 | -0.1310 | 0.1245 |
| TMJ | 0.3859 | 0.3849 | 0.3881 | 0.3837 | 0.3812 | 0.3730 | 0.3527 | 0.3131 | 0.2063 | 0.3521 |
| Feng | 0.3872 | 0.3887 | 0.3925 | 0.3889 | 0.3862 | 0.3779 | 0.3578 | 0.3169 | 0.2053 | 0.3557 |
| Mu | 0.4062 | 0.3960 | 0.3941 | 0.3780 | 0.3525 | 0.3158 | 0.2667 | 0.1905 | 0.0932 | 0.3103 |
| TA | 0.4221 | 0.3815 | 0.3867 | 0.3839 | 0.3818 | 0.3771 | 0.3613 | 0.3279 | 0.2325 | 0.3616 |
| TAJ | 0.4311 | 0.3879 | 0.3910 | 0.3871 | 0.3843 | 0.3761 | 0.3554 | 0.3157 | 0.2243 | 0.3614 |
| TAN | 0.4338 | 0.3869 | 0.3901 | 0.3793 | 0.3712 | 0.3583 | 0.3366 | 0.2951 | 0.2148 | 0.3518 |
| TANJ | 0.4413 | 0.3964 | 0.4000 | 0.3898 | 0.3810 | 0.3650 | 0.3381 | 0.2913 | 0.2110 | 0.3571 |
| SMD | 0.4180 | 0.3779 | 0.3818 | 0.3792 | 0.3777 | 0.3739 | 0.3619 | 0.3459 | 0.3008 | 0.3686 |
| HSMD | 0.4257 | 0.3846 | 0.3892 | 0.3867 | 0.3835 | 0.3763 | 0.3563 | 0.3138 | 0.1617 | 0.3531 |
| HSMDJ | 0.3880 | 0.3876 | 0.3897 | 0.3855 | 0.3799 | 0.3674 | 0.3414 | 0.2971 | 0.1955 | 0.3480 |
| QTI | 0.3912 | 0.3879 | 0.3908 | 0.3854 | 0.3757 | 0.3584 | 0.3270 | 0.2782 | 0.1832 | 0.3420 |
| QTIJ | 0.3941 | 0.3846 | 0.3835 | 0.3754 | 0.3610 | 0.3384 | 0.3022 | 0.2546 | 0.1732 | 0.3297 |
| NNMS | 0.3892 | 0.3862 | 0.3862 | 0.3806 | 0.3746 | 0.3595 | 0.3318 | 0.2862 | 0.1894 | 0.3426 |

**Table 8.** R metric within estimation process

Top-5 measures according to R metric within recommendation process are SMD, Cosine, TA, TAJ, and MSDJ whose average recall metrics are 0.3686, 0.3663, 0.3616, 0.3614, and 0.3598. Our SMD measure is in the top-5 list given R metric. Shortly, dominant orders of our measures TA, TAJ, TAN, TANJ, SMD, HSMD, HSMDJ, QTI, QTIJ, and NNMS are 3rd, 4th, 17th, 10th, 1st, 15th, 18th, 20th, 21st, and 19th among 32 measures given R metric, respectively.

The best measure TAJ drawn from tables 6 are still in lists of top-5 measures with regard to MSE and R shown in tables 7 and 8. This implies the same semantics of MAE, MSE, and R within estimation process. It is possible to conclude that the important problem is to split the evaluation process into two sub-processes such as estimation and recommendation. For each sub-process, we only need to choose one representative metric. In this research, we choose MAE and F1 as representative metrics for estimation process and recommendation process, respectively.

Although the best measures are TAJ and NHSM with representative metrics MAE and F1, two our measures TAJ and SMD are also preeminent measures. TAJ is dominant over metrics MAE, precision, F1, MSE, and R whereas SMD is dominant over metrics MAE, recall, MSE, and R. Note that NHSM is not preeminent measure with metrics MSE and R. As usual, we define that preeminent measures (dominant measures) are ones in top-5 lists. It is useful to compare NHSM, SMD, and TAJ but it is impossible to unify metrics MAE, MSE, and R together. However, we can compare them by radar chart but some transformations are necessary. Let I-R be inverse of R metric. Let I-Precision be inverse of precision metric and let I-Recall be inverse of recall metric. The smaller I-R, I-Precision, and I-Recall are, the better the measures are. Equation 51 specifies I-R, I-Precision, and I-Recall. Hence, I-R, I-Precision, and I-Recall are replacers of R, Precision, and Recall.

|  |  |
| --- | --- |
|  | (51) |

Table 9 lists metrics MAE, MSE, I-R, I-Precision, and I-Recall of preeminent measures NHSM, SMD, and TAJ.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | MAE | MSE | I-R | I-Precision | I-Recall |
| NHSM | 0.7712 | 0.9695 | 0.6419 | 0.9681 | 0.1003 |
| TAJ | 0.7699 | 0.9637 | 0.6386 | 0.9681 | 0.0985 |
| SMD | 0.7709 | 0.9618 | 0.6314 | 0.9726 | 0.0895 |

**Table 9.** Comparison of NHSM, SMD, and TAJ with MAE, MSE, I-R, I-Precision, and I-Recall

From table 9, TAJ is the best with MAE. SMD is the best with MSE, I-R, and I-Recall. NHSM and TAJ are the best with I-Precision.

Figure 4 shows radar chart of preeminent measures NHSM, SMD, and TAJ regarding MAE, MSE, I-R, I-Precision, and I-Recall.

A close up of a map

Description automatically generated

**Figure 4.** Comparison of NHSM, SMD, and TAJ with MAE, MSE, I-R, I-Precision, and I-Recall

As seen in figure 4, lines of NHSM, SMD, and TAJ are nearly overlapped. Hence, it is not necessary to determine which one is the best over all metrics.

Speed is also a metric to evaluate measures but the bias to calculate speed metric in evaluation process is high due to casual factors of hardware and software. Therefore, speed is not important metric in this research. From experiments, speed values of NHSM, SMD, and TAJ are 0.3288, 0.3116, 0.3411 in millisecond, respectively. Hence, SMD is fastest.

**5. Conclusions**

Besides splitting the evaluation processes into estimation process and recommendation and proposing four new measures such as SMD, HSMD, QTI, and NNMS, this research can be considered as a short summary of similarity measures for CF. We evaluate and compare many measures by appropriate and succinct way. For evaluating similarity-based CF algorithm, choosing how many tested metrics is not important. The most important thing is to choose right representative metrics for estimation process and recommendation process with note that such two processes are different. For instance, it is not accurate if we calculate MAE metric for recommendation process or F1 metric for estimation process. Of course, it is good if we calculate many right representative metrics for each process but it is easier to draw best measures with small set of right representative metrics.

From experiments, our SMD measure is a preeminent measure which is in top-5 lists with metrics MAE, recall, MSE, and R. The common feature of SMD and Jaccard is that they only concern existence of ratings and so they do not concern magnitude of ratings. Jaccard is itself not a dominant measure but it is an important factor to improve any measure (Nguyen & Amer, 2019). In fact, good measures such as TANJ, NHSM, TAJ, MSDJ combine themselves with Jaccard. From experiments, SMD is obviously better than Jaccard. Therefore, it is potential to combine SMD with other measures. In the future, we focus on combining SMD with other measures.

**References**

Choi, K., & Suh, Y. (2013, January). A new similarity function for selecting neighbors for each target item in collaborative filtering. (H. Fujita, & J. Lu, Eds.) *Knowledge-Based Systems, 37*(2013), 146-153. doi:10.1016/j.knosys.2012.07.019

Do, M.-P. T., Nguyen, D. V., & Nguyen, L. (2010). Model-based Approach for Collaborative Filtering. *Proceedings of The 6th International Conference on Information Technology for Education (IT@EDU2010)* (pp. 217-225). Ho Chi Minh: Ho Chi Minh University of Information Technology. Retrieved from https://goo.gl/BHu7ge

Feng, J., Fengs, X., Zhang, N., & Peng, J. (2018, September 24). An improved collaborative filtering method based on similarity. (H. Wang, Ed.) *PLos ONE, 13*(10), 1-18. doi:10.1371/journal.pone.0204003

GroupLens. (1998, April 22). *MovieLens datasets*. (GroupLens Research Project, University of Minnesota, USA) Retrieved August 3, 2012, from GroupLens Research website: http://grouplens.org/datasets/movielens

Harris, E. F., & Sjøvold, T. (2018, September 7). Calculation of Smith's Mean Measure of Divergence for Intergroup Comparisons Using Nonmetric Data. (E. F. Harris, Ed.) *Dental Anthropology: A Publication of the Dental Anthropology Association, 17*(3), 83-93. doi:10.26575/daj.v17i3.152

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems (TOIS), 22*(1), 5-53. doi:10.1145/963770.963772

Hyung, J. A. (2008, January 2). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. (W. Pedrycz, Ed.) *Information Sciences, 178*(1), 37-51. doi:10.1016/j.ins.2007.07.024

Lin, Y.-S., Jiang, J.-Y., & Lee, S.-J. (2013, January 25). A Similarity Measure for Text Classification and Clustering. (J. Pei, Ed.) *IEEE Transactions on Knowledge and Data Engineering, 26*(7), 1575-1590. doi:10.1109/TKDE.2013.19

Liu, H., Hu, Z., Mian, A., Tian, H., & Zhu, X. (2013, November 20). A new user similarity model to improve the accuracy of collaborative filtering. (H. Fujita, & J. Lu, Eds.) *Knowledge-Based Systems, 56*(2014), 156-166. doi:10.1016/j.knosys.2013.11.006

Montgomery, D. C., & Runger, G. C. (2010). *Applied Statistics and Probability for Engineers* (5th ed.). Hoboken, New Jersey, USA: John Wiley & Sons. Retrieved September 6, 2016, from https://books.google.com.vn/books?id=\_f4KrEcNAfEC

Mu, Y., Xiao, N., Tang, R., Luo, L., & Yin, X. (2019, January). An Efficient Similarity Measure for Collaborative Filtering. (R. Bie, Y. Sun, & J. Yu, Eds.) *Procedia Computer Science, 147*(2019), 416-421. doi:10.1016/j.procs.2019.01.258

Nguyen, L., & Amer, A. A. (2019, October 17). Advanced Cosine Measures for Collaborative Filtering. (ITS, Ed.) *Adaptation and Personalization (ADP), 1*(1), 21-41. doi:10.31058/j.adp.2019.11002

Patra, B. K., Launonen, R., Ollikainen, V., & Nandi, S. (2015, March 1). A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data. (H. Fujita, & J. Lu, Eds.) *Knowledge-Based Systems, 82*, 163-177. doi:10.1016/j.knosys.2015.03.001

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based Collaborative Filtering Recommendation Algorithms. *Proceedings of the 10th international conference on World Wide Web* (pp. 285-295). Hong Kong: ACM. doi:10.1145/371920.372071

Sun, S.-B., Zhang, Z.-H., Dong, X.-L., Zhang, H.-R., Li, T.-J., Zhang, L., & Min, F. (2017, August 17). Integrating Triangle and Jaccard similarities for recommendation. (Q. Zou, Ed.) *PLoS ONE, 12*(8), 1-16. doi:10.1371/journal.pone.0183570

Suryakant, & Mahara, T. (2016, January). A New Similarity Measure Based on Mean Measure of Divergence for Collaborative Filtering in Sparse Environment. (R. K. Venugopal, R. Buyya, & M. L. Patnaik, Eds.) *Procedia Computer Science, 89*(2016), 450-456. doi:10.1016/j.procs.2016.06.099

Torres Júnior, R. (2004). *Combining Collaborative and Content-based Filtering to Recommend Research Paper.* Universidade Federal do Rio Grande do Sul, Programa de Pós Graduação em Educação. Porto Alegre: Universidade Federal do Rio Grande do Sul. Retrieved from http://www.lume.ufrgs.br/bitstream/handle/10183/5887/000432990.pdf;sequence=1