Experimental interpretation of adequate weight-metric combination for dynamic user-based collaborative filtering

Savas Okyay [Equal first author, 1, 2] , **Sercan Aygun** [Corresp. Equal first author, 3, 4]

1 Computer Engineering, Eskisehir Osmangazi University, Eskisehir, Turkey

2 Computer Engineering, Eskisehir Technical University, Eskisehir, Turkey

3 Computer Engineering, Yildiz Technical University, Istanbul, Esenler, Turkey

4 Electronics Engineering, Istanbul Technical University, Istanbul, Maslak, Turkey

Corresponding Author: Sercan Aygun
Email address: ayguns@itu.edu.tr

Recommender systems include a broad scope of applications as an appealing research area. A recommender system is associated with subjective preferences. Therefore, proper recommendations that pose a compelling problem vary for each individual. As a field of data science and machine learning, recommender systems require both statistical perspective and sufficient performance monitoring. In this study, we propose diversified similarity measurements by observing performances with generic metrics. In the sense of *user-based* collaborative filtering, it is measured how likely an item is preferred by any user. Having examined the best neighbor counts, we do unveil the test item bias phenomenon for similarity equations. Due to the statistical parameters used to be computed in a global scope beforehand, there is implicit information in the literature, whether those parameters comprise the focal point user data statically. For each dynamic prediction, user-wise parameters are expected to be generated in runtime by leaving the item-of-interest out. This both gives reliable results and is more compatible with real-time systems. Furthermore, we underline the effect of significance weighting by boosting the similarities between a user-of-interest and its neighbors. All in all, this work uniquely combines significance weighting and test item bias mitigation by inspecting the fine-tuned neighborhood. As a consequence, considering the visual comparison of results and elaborated heat-map tables at the end, concluding remarks of adequate *similarity weight* and *performance metric* combinations are interpreted. The source code of our architecture is available at https://codeocean.com/capsule/1427708/tree/v1.

Experimental Interpretation of Adequate Weight-Metric Combination for Dynamic User-Based Collaborative Filtering

Savas Okyay [1,2], Sercan Aygun [3,4]

[1] Computer Engineering Dept., Eskisehir Osmangazi University, Eskisehir, Turkey
[2] Computer Engineering Dept., Eskisehir Technical University, Eskisehir, Turkey
[3] Computer Engineering Dept., Yildiz Technical University, Istanbul, Turkey
[4] Electronics Engineering Dept., Istanbul Technical University, Istanbul, Turkey

*Authors have equal contributions, and names are given alphabetically.

Corresponding Author:
Sercan Aygun [3,4]
Davutpasa, Istanbul, Esenler, 34220, Turkey
Email address: ayguns@itu.edu.tr

Abstract

Recommender systems include a broad scope of applications as an appealing research area. A recommender system is associated with subjective preferences. Therefore, proper recommendations that pose a compelling problem vary for each individual. As a field of data science and machine learning, recommender systems require both statistical perspective and sufficient performance monitoring. In this study, we propose diversified similarity measurements by observing performances with generic metrics. In the sense of *user-based* collaborative filtering, it is measured how likely an item is preferred by any user. Having examined the best neighbor counts, we do unveil the test item bias phenomenon for similarity equations. Due to the statistical parameters used to be computed in a global scope beforehand, there is implicit information in the literature, whether those parameters comprise the focal point user data statically. For each dynamic prediction, user-wise parameters are expected to be generated in runtime by leaving the item-of-interest out. This both gives reliable results and is more compatible with real-time systems. Furthermore, we underline the effect of significance weighting by boosting the similarities between a user-of-interest and its neighbors. All in all, this work uniquely combines significance weighting and test item bias mitigation by inspecting the fine-tuned neighborhood. As a consequence, considering the visual comparison of results and elaborated heat-map tables at the end, concluding remarks of adequate *similarity weight* and *performance metric* combinations are interpreted. The source code of our architecture is available at https://codeocean.com/capsule/1427708/tree/v1.

Introduction

Recommender systems (RS) are utilized in a wide variety of applications as an outstanding research area. Users act on a range of application-specific platforms. Personal data, along with the previous activities, are combined to understand the user's taste. Any kind of recommendation from the items on the platform itself can be supplied. Including both online and offline applications, a stable architecture is essential not only for the machine learning perspective but also for the business aspect of any platform. The implementation of RS has a variety of platforms, including social media [1], healthcare [2], journal [3], music [4], [5] suggestion systems, and last but not least, the movie recommendation frameworks [6]–[8]. In RS, the main motivation is to analyze the preferences and to decide the prospective action of a user. A person has activities on a specific application, such as passing remarks, leaving comments, giving rates, liking or disliking products, as all are logged into a database system. There are two significant reasons that movie-based RS has been the focus of many data scientists. The first is readily available scientific datasets like *MovieLens* [9] and *Netflix* [10] with ease of use. The second is the applicability of an overall RS architecture that is entirely compatible with the additional user and item features; thereby, enabling scientists to measure two well-known phenomena in collaborative filtering (CF) as (i) *user-based* similarity and (ii) *item-based* similarity.

One of the main motivations of this study is to measure the effect of correlation adjustment. In the literature, there are loads of theoretical RS implementations; however, there is blurry information on the inclusion or exclusion of the test item during statistical parameter computations. Similarity calculation between two users requires some analytical computations like *mean* and *median*. Theoretical studies may set statistical arguments as global parameters to encapsulate computations for all upcoming test attempts concerning time complexity and memory management. Even though setting parameters globally is computation friendly, if all those statistical primitives are set in this wide scope, test items become less dependent on the related test attempts. Hence, the expected recommendation might slightly be falsified. In real-time applications, the rating value of the item to be recommended is not known. In this work, the dynamicity effect of the *Item-of-Interest* (*IOI*) is examined to demonstrate how theory and real-time performance could differ. Apart from the *IOI* condition, the *Co-rated Item Count* (*CIC*) between the user-of-interest and its neighbors is utilized to revise the calculated similarity weight for further constant multiplication [11]–[16]. Thus, the correlation between users is linked to the commonly rated item counts. We call this multiplication the *CIC-based significance weighting (SW)* method by showing how it performs. We interpret the efficiency of *IOI* and *SW* conditions based on four different similarity equations, namely *Pearson similarity*, *median-based robust correlation*, *cosine similarity*, and *Jaccard similarity*.

In general, researchers focus on finding a way to increase the efficiency of RS. The closer forecast to the user preference is obtained, the more accurate system design is achieved. However, the performance metrics of a system can be more than a single prediction accuracy. In this study, we address the previously proposed renowned similarity equations and performance

79   metrics in a comparative manner. The research constructs a perspective on how to link user
80   similarity measurements to the enlarged number of performance metrics, including the ones from
81   other disciplines. Schröder et al. propose to utilize relatively less-known metrics such as
82   *informedness*, *markedness*, and *Matthews correlation* since they underline the superiority over
83   *precision*, *recall*, and *F1-measure* [17]. As the aforementioned metrics are acknowledged by
84   Schröder et al. to be suitable for the decision of top-*n* recommendation in *e*-commerce
85   applications, we do answer the question on how those metrics perform compared to the well-
86   knowns.
87   The previous RS implementations have either a relatively small set of metrics to test [18]–[21] or
88   a limited range of specific parameters, like *the best neighborhood* [11], [22]–[26]. Any user is
89   served with a recommendation by looking at the closest neighbors who have the same tendencies
90   for the related *IOI*. Instead of tuning the best neighbor count (*BNC*) at a constant value, the
91   neighborhood should be appropriately decided. For this reason, we parameterize the number of
92   neighbors, s.t. using ε step-size in between the least neighbor count (*LNC*) and the most neighbor
93   count (*MNC*).
94   On the implementation side of this study, a comprehensive back-end software architecture is
95   developed. Our framework[1] is an adaptive tool, which enables the test environment to capture the
96   general behavior of any high-density dataset with a fine-tuned ε. The random outliers in the
97   dataset are suppressed with the aid of the optimized algorithm.
98   To the best of our knowledge, there is no previous RS research that extensively looks up for the
99   adequate combination of similarity measurements and performance metrics. All in all, the
100   following highlights are presented in the scope of this study:
101   •   We draw a vivid picture to construct an RS framework highlighting the possible pitfalls and
102   enhancements on the architecture design. To that end, the following two perspectives, both
103   independently and together, are applied to the similarity equations.
104   o   The first perspective is to underline the *dynamicity* principles of real-time systems by
105   excluding the *IOI*, named as *no Item-of-Interest*, *nIOI*.
106   o   The second perspective is to highlight the results on the utilization of signified
107   weights. With this *SW* method, the more common rating counts from the neighbors
108   are observed, the more signified weights are obtained.
109   •   The *BNC* is analyzed and decided experimentally under a variety of performance metrics.
110   •   Extensive tests are applied to popular *MovieLens* releases with randomized trials of separate
111   runs.
112   •   In the evaluation part, the relatively less-known performance metrics such as *informedness*,
113   *markedness*, *Matthews correlation* are monitored comprehensively. In addition, reputable
114   metrics such as *precision*, *sensitivity*, *specificity*, *F1-measure*, *fallout*, *miss rate*, etc., as well

---

[1] Open-source code information is given in the Acknowledgment. Any dataset can be analyzed
as long as it meets the requirement of *user×item* matrix format.

115   as error metrics are given in a detailed comparison. These prediction-oriented metrics are
116   extensively demonstrated with notable outcomes.
117 • *Prevalence threshold* and *threat score*, which are frequently practiced in other disciplines,
118   are promisingly analyzed in the context of RS.
119 • Finally, it is presented the heat-map tables for the top-performing *BNC*s, linked to the
120   adequate weight-metric combinations.
121 The organization of this paper is as follows. The materials and methods are given in Section II,
122 where the nomenclature and dataset details can be found. Plus, similarity equations and
123 performance metrics used throughout this study take place in the same section. Section III
124 includes the details of the computation environment and the preliminary selection of top-
125 performing neighbors, which is then followed by the extensive results in Section IV. In the last
126 section, the conclusion is presented with the future work.
127

Materials & Methods

129 This section describes the dataset in use and the applied methods. First, the technical details on
130 the *MovieLens* releases are given. Then, the touchstone similarity equations together with the
131 modifications considering the *nIOI* phenomenon and *SW* are discussed. Finally, the performance
132 metrics implemented in our RS framework are presented. The symbols and abbreviations used
133 throughout this paper are listed in Table I to make the rest of the paper easy to follow.
134

A. THE MOVIELENS

136 In the current RS applications, the basic structure of data in practice commonly has the *user×item*
137 matrix format. One of the frequently trained scientific datasets is *MovieLens* [27], which has
138 several releases based on the size and the additional content.
139 In Table II, the main types of *MovieLens* can be reviewed depending on the rating size, e.g., the
140 ML100K has 100,000 clicks. *MovieLens* dataset is upgraded several times, not only for the
141 expanded types but also for the versions of previous releases. For instance, ML100K type has
142 various releases, like the one that includes 1 to 5 ratings only with decimal values. Nonetheless,
143 the other ML100K version consists of 0.5 steps between ratings, namely the *half stars*. However,
144 this version is not recommended for shared research results since it is a development dataset.
145 Most of the previous studies focus on tried-and-trusted original ML100K release, which is a
146 pioneering collection and has considerably efficient runtime performance. In the scope of this
147 study, we utilize this original release that includes only the full-stars. We additionally
148 encapsulate extensive experiments of ML1M to keep full-star rating scaling parallelism along
149 with ML100K. Thereby, we present the results related to the original ML100K and ML1M by
150 holding the interpretation in the same contrast.
151

B. SIMILARITY AND PREDICTION EQUATIONS

153   The four touchstone similarity equations and the prediction formula are taken into account in this
154   section. Before the technical statements, an overview related to the application perspective of the
155   touchstone equations is given first.
156   *Pearson Correlation Coefficient* (PCC) is in the scope of several studies. One of the most applied
157   areas is the music RS [28]. Especially for music genre recommendation, PCC has an attention. In
158   addition, there are other kinds of applications. Mukaka explains the management of medical data
159   on the utilization of PCC [29] like other studies [30], [31]. Apart from these, book
160   recommendations via PCC [32], [33], *e*-commerce application [34], and academic paper RS [35]
161   are intriguing alternatives. Other PCC examples can be found in [36]–[39]. After all, the most
162   applied field is the movie-based RS [40]–[42] as the motivation of our study. Movie genre
163   correlations are calculated using PCC by Kim et al. [43]. Hwang et al. also present the details of
164   PCC on dealing with the movie genre classification [44]. Nonetheless, PCC is said to have some
165   disadvantages in terms of linear procedures of averaging. Tan et al. indicate the underlying
166   limitations of PCC to reinforce the effect of correlation firstly. Then, they propose the resonance
167   similarity between users by parametrizing the median of rating values. They construct a physical
168   analogy between user similarities in the sense of simple harmonic motion on a coordinate system
169   [45]. The mean of rating vectors can be vulnerable to outliers and biases, as Garcin et al. obtain
170   the prediction by aggregating the median stats instead of executing the weighted average [46].
171   Therefore, *Median-Based Robust Correlation* (MRC) is proposed to be utilized within the
172   context of RS suppressing outliers in the user ratings. Besides, *Cosine* (COS) is another frequent
173   method in RS, thanks to its simple calculation, and is encountered in movie-related applications
174   [47], [48], research paper recommendation [49]–[51], cognitive similarity-based design [21],
175   article suggestion system [52], and music RS [53]. Furthermore, *Jaccard* (JAC) is also evaluated
176   by several studies in the literature [19], [26], [54], [55]. JAC has an essential feature in the sense
177   of binary rating analysis [56] and is considered as a measure that does not treat the absolute
178   ratings [57].
179   Together with the proposed *nIOI* and *SW* modifications over similarity equations, different
180   combinations are interpreted by inferring the underlying affinity in between. In the following,
181   first, PCC equation and then, MRC, COS, and JAC similarities are stated technically. In the light
182   of various performance metrics visited in subsection D, adequate weight-metric combinations are
183   to be concluded thereafter.
184
185   1) PEARSON CORRELATION
186   Pearson correlation is an acclaimed phenomenon practiced in many data mining approaches
187   addressing the measurement of data similarity. On the user-based CF side, the PCC is a tool to
188   define in-between user similarity by considering the item ratings. Pearson weighs all connected
189   neighbors and calculates the degree of a linear relationship between two users. Thus, a weight for
190   each correlated neighbor is derived that gives a linear relationship by processing the deviation
191   from mean values, $\bar{r}$ [58]. In Eq. (1), the similarity formula between two users, a and u, is
192   indicated.

$$w^{PCC}_{a,u} = \frac{\Sigma_{i \in (I_a \cap I_u)}((r_{a,i} - \bar{r}_a) \times (r_{u,i} - \bar{r}_u))}{\sqrt{\Sigma_{i \in (I_a \cap I_u)}(r_{a,i} - \bar{r}_a)^2} \times \sqrt{\Sigma_{i \in (I_a \cap I_u)}(r_{u,i} - \bar{r}_u)^2}} \tag{1}$$

193

194   2) MEDIAN-BASED ROBUST CORRELATION

195   Median-based robust correlation is a method that replaces the linear *mean* procedures with the

196   *median* operation [59]–[61]. The utilization of the averages might suffer from the *skewness*

197   problem [62], [63]. Besides, outliers can cause falsified mean values. MRC, which has the

198   median of rating values instead of the averages like in PCC, stands for suppressing outliers in the

199   ratings of each user. In Eq. (2), MRC formula is given.

$$w^{MRC}_{a,u} = \frac{\Sigma_{i \in (I_a \cap I_u)}((r_{a,i} - \tilde{r}_a) \times (r_{u,i} - \tilde{r}_u))}{\sqrt{\Sigma_{i \in (I_a \cap I_u)}(r_{a,i} - \tilde{r}_a)^2} \times \sqrt{\Sigma_{i \in (I_a \cap I_u)}(r_{u,i} - \tilde{r}_u)^2}} \tag{2}$$

200   Unlike the mean values of the user ratings, $\bar{r}_a$ and $\bar{r}_u$ in Eq. (1), the median values, $\tilde{r}_a$ and $\tilde{r}_u$,

201   represent the mid-point of the ratings. The formula is similar to PCC, and the median point of the

202   user ratings is considered a neutral mark.

203

204   3) COSINE SIMILARITY

205   The one other similarity is based on the *cosine* function. By performing the Euclidean dot

206   product, the cosine value in between two *n*-element vectors, A and B, can be found out. Thus, the

207   similarity is based on $A . B = \|A\|\|B\|cos\Theta$. For the *user-based* similarity calculation via COS,

208   the similarity weight between a and u can be measured like in Eq. (3).

$$w^{COS}_{a,u} = \frac{r_a \cdot r_u}{\|r_a\| \|r_u\|} \tag{3}$$

209   There are also some other versions of the conventional COS in the literature, such as *adjusted*

210   *cosine similarity* [64] and *asymmetric cosine similarity* [53].

211

212   4) JACCARD SIMILARITY

213   Jaccard similarity is the measure of common elements in two sets. The rating history of a user

214   under test, I_a, and the corresponding neighbor history, I_u, are treated according to Eq. (4). JAC

215   considers the two sets by having the ratio of their *intersection* over the *union*. The range of this

216   similarity coefficient is $0 \leq w_{a,u} \leq 1$, where 0 represents that there are no elements in common,

217   whereas 1 implies that all elements in between two sets are fully joint.

$$w^{JAC}_{a,u} = \frac{|I_a \cap I_u|}{|I_a \cup I_u|} \tag{4}$$

218

219   5) PREDICTION EQUATION

220   After the similarity calculations for all the best neighbor nominees, the obtained weights are

221   sorted by denoting w_{a,u^*}. Then, according to the sorted weights and BNC limits, the best

222   neighbors are determined. After all, the prediction phase has to be completed to achieve the

223 recommendation score. The rating prediction formula, which is named the *mean centering*
224 approach [65]–[69], is given in Eq. (5).

$$p_{a,i} = \bar{r}_a + \frac{\sum\limits_{u^*=1}^{BNC} \left((r_{u^*,i} - \bar{r}_{u^*}) \times w_{a,u^*} \right)}{\sum\limits_{u^*=1}^{BNC} w_{a,u^*}} \tag{5}$$

225

226 ### C. MODIFIED EQUATIONS
227 Using the equations given in the previous section, we do perform several updates on these
228 formulas. Thereby, the efficiency of RS significantly improves in some circumstances.
229 The motivation behind the modifications stems from the two phenomena as (i) making a system
230 model that fits real-time applications and (ii) boosting the similarity weights. The former deals
231 with the dynamicity concept, while the latter is for the user-of-interest and its neighbor by
232 considering their CIC as a constant multiplier, thereby signified weights could be obtained.
233 For the first phenomenon, the already rated test item is discarded from the user-of-interest rating
234 history to predict the actual rating. Thus, during the *mean* or *median* calculations in the formulas
235 like PCC and MRC, the item is excluded as is expected in real-time systems. This case is also
236 valid for the other measurements, such as COS and JAC, where the related item is removed from
237 the vectors in progress. To indicate this phenomenon, we use the *nIOI* subscript by denoting $\hat{a}$ in
238 the equations. As explained in the first section, the negligence of *nIOI* in many other RS
239 applications is thought to be due to runtime concerns on the vast scientific tests.
240 The second phenomenon is the relative weight scaling named *SW*. This is to give priority to a
241 neighbor who has more common ratings for the items. After calculating the co-rated item count,
242 the weights in similarity calculations are signified using this $CIC = |I_a \cap I_u|$ constant multiplier
243 [70]. In the literature, there are some other alternatives [11]–[16], [70]. In [15], Bellogín et al.
244 explain different *user-user* weighting schemes comparatively. The one we set in this study is
245 called *user overlap*, which calculates the common item counts in between user-neighbor [71].
246 There are also *Herlocker's significance weighting* [72] and *McLaughlin's significance weighting*
247 [73], together with *trustworthiness* [74] and *trust deviation* [75]. However, they either include
248 extra parameters or require complex computations. Raeesi and Shajari compare the *SW* strategies
249 by underlining the *user overlap*, which outperforms in terms of the error rates, even though there
250 are fewer arguments to process [71].
251 By considering the modifications, each equation from the previous section is updated with the
252 aforementioned two phenomena. For PCC, recalling Eq. (1) in the previous section, Eq. (6) is
253 obtained by excluding the test item bias. The next, Eq. (7), is the signified version of Eq. (1) by
254 applying *SW* alone.

$$w^{PCC_{nIOI}}_{a,u} = \frac{\Sigma_{i \in (I_{\hat{a}} \cap I_u)}\left((r_{\hat{a},i} - \bar{r}_{\hat{a}}) \times (r_{u,i} - \bar{r}_u)\right)}{\sqrt{\Sigma_{i \in (I_{\hat{a}} \cap I_u)}(r_{\hat{a},i} - \bar{r}_{\hat{a}})^2} \times \sqrt{\Sigma_{i \in (I_{\hat{a}} \cap I_u)}(r_{u,i} - \bar{r}_u)^2}} \tag{6}$$

$$w^{PCC^{SW}}_{a,u} = |I_a \cap I_u| \times w^{PCC}_{a,u} \tag{7}$$

255 The same approach is followed for *MRC* in the following Eq. (8). The *SW* multiplication can be
256 found in Eq. (9).

$$w^{MRC_{nIOI}}_{a,u} = \frac{\Sigma_{i \in (I_{\hat{a}} \cap I_u)}\left((r_{\hat{a},i} - \tilde{r}_{\hat{a}}) \times (r_{u,i} - \tilde{r}_u)\right)}{\sqrt{\Sigma_{i \in (I_{\hat{a}} \cap I_u)}(r_{\hat{a},i} - \tilde{r}_{\hat{a}})^2} \times \sqrt{\Sigma_{i \in (I_{\hat{a}} \cap I_u)}(r_{u,i} - \tilde{r}_u)^2}} \tag{8}$$

$$w^{MRC^{SW}}_{a,u} = |I_a \cap I_u| \times w^{MRC}_{a,u} \tag{9}$$

257 For *COS*, Eq. (10) shows the vector operations of the ratings in which the test item bias is
258 discarded. *SW* approach is followed as in Eq. (11).

$$w^{COS_{nIOI}}_{a,u} = \frac{r_{\hat{a}} \cdot r_u}{\|r_{\hat{a}}\| \, \|r_u\|} \tag{10}$$

$$w^{COS^{SW}}_{a,u} = |I_a \cap I_u| \times w^{COS}_{a,u} \tag{11}$$

259 Finally, *JAC* with the modifications is given in Eq. (12) and Eq. (13).

$$w^{JAC_{nIOI}}_{a,u} = \frac{|I_{\hat{a}} \cap I_u|}{|I_{\hat{a}} \cup I_u|} \tag{12}$$

$$w^{JAC^{SW}}_{a,u} = |I_a \cap I_u| \times w^{JAC}_{a,u} \tag{13}$$

260 Even though previous studies [11]–[15], [71]–[75] present well understanding of *SW* using
261 different perspectives, detailed performance analysis is missing. We do contribute in terms of the
262 relative comparison of similarity equations enhanced with *SW*, including their corresponding
263 performance.
264 The two phenomena, *nIOI* and *SW*, are independently measured first. Then, to monitor the hybrid
265 effect of these approaches, both of them are utilized by obeying the generalized formula in Eq.
266 (14).

$$w^{SIMILARITY^{SW}_{nIOI}}_{a,u} = |I_{\hat{a}} \cap I_u| \times w^{SIMILARITY_{nIOI}}_{a,u} \tag{14}$$

267 The modified rating prediction formula is also given in Eq. (15). Considering the *nIOI*, $\bar{r}_a$ is
268 updated compared to the original equation in Eq. (5).

$$p_{a,i} = \bar{r}_{\hat{a}} + \frac{\displaystyle\sum_{u^* = 1}^{BNC}\left((r_{u^*,i} - \bar{r}_{u^*}) \times w_{a,u^*}\right)}{\displaystyle\sum_{u^* = 1}^{BNC} w_{a,u^*}} \tag{15}$$

269

270 ### D. PERFORMANCE METRICS

271   The final phase of the proposed RS design is the monitoring of the running algorithm. As CF is

272   an intersection of statistics and machine learning, a sound knowledge of a performance check is

273   needed. The application-oriented practice of the metrics is intriguing research to be handled.

274   Particularly, understanding the interrelational achievement of the similarity equations with

275   implied modifications requires a thorough performance monitoring through numerous metrics.

276   Well-known metrics in Table III are applied in the framework to provide insight for further

277   studies as our primary objective. The explanations of listed metrics are briefly summarized as

278   follows.

279   First of all, *exact accuracy* is a metric to measure the exact matches of actual ratings ($r_{a,i}$) and

280   corresponding predictions ($p_{a,i}$). The accuracy computation is handled for the predicted rating,

281   controlling whether $p_{a,i} = r_{a,i}$ or $p_{a,i} \neq r_{a,i}$. For the frameworks that use *N*-scale ratings, *exact*

282   *accuracy* can source a precise observation. Also, there is *threshold accuracy* following the

283   binary decision of *liked* and *disliked* items. By denoting $r_{a,i} \in \mathbb{N}^+$ or $r_{a,i} \in \mathbb{R}^+$ where $argmax($

284    $r_{a,i}) = N$ over *N*-scale ratings, $t \in \mathbb{R}^+$ as a threshold value should be set satisfying $t < N$. Then,

285   the rating value r compared to threshold value t is evaluated to label *liked* or *disliked* items in a

286   binary sense [19].

287   Furthermore, correctly predicted positive values are measured via *sensitivity,* which is also called

288   *recall* or *true positive rate*. Besides, the measure of actual positives is monitored by *precision*,

289   namely *positive predictive value*. *Precision* is referred by Powers [76] as *true positive accuracy,*

290   which implies a confidence score. From *sensitivity* and *precision*, *F1-measure* is calculated via

291   harmonic mean. On the other hand, aside from the positive decisions made, negatives are also

292   considered. Thus, *specificity* (or *inverse sensitivity*) presents the proportion of real negative

293   cases. Moreover, *inverse precision* (or *negative predictive value*), which is also called *true*

294   *negative value* by Powers in [76], shows the predicted negative instances. *False discovery rate*

295   and *false omission rate* are deduced complementarily from the maximum metric score of

296   *precision* and *inverse precision*. Shani and Gunawardana underline that *false discovery rate* can

297   be an alternative control mechanism [77], which is the proportion of *FP* over actual positives. In

298   a similar manner, the *false omission rate* is the ratio of *FN* over all negatives [78].

299   *Sensitivity* and *specificity* attribute *true positive rate* and *true negative rate,* respectively.

300   Likewise, *fallout* and *miss rate* represent *false positive* and *false negative rates*. The irrelevant

301   recommendation ratio is obtained via *fallout*. *Miss rate* stands for the ratio of the items that are

302   not recommended despite being relevant. A recent study from the literature performs *fallout* and

303   *miss rate* by practicing a personalized nutrition recommendation study [79]. Like *F1-measure*,

304   the utilization of *precision* and *recall* by means of geometric mean also appears in *Fowlkes–*

305   *Mallows index*. In another recent study, Panda et al. discuss how to increase *Fowlkes–Mallows*

306   *index* like *F1-measure* as one of their motivation [80]. In the case of an imbalanced confusion

307   matrix, *balanced accuracy* gives a better perspective for performance analyses. *Balanced*

308   *accuracy* is the arithmetic mean of *sensitivity* and *specificity*. In a broad view of understanding

309   the algorithm efficiency, utilizing several metrics like *balanced accuracy* brings considerable
310   feedback.
311   The final metrics are *threat score* and *prevalence threshold*. The former takes the *hits*, *misses*,
312   and *false alarms* in the confusion matrix [81]. The latter emphasizes a sharp change in *positive*
313   *predictive value*. *Prevalence threshold* with a more geometric interpretation of the performance
314   measurement with a focus on *positive* and *negative predictive values* is given in [82], as well as
315   applied recently on the test analyses of the COVID-19 screening [83].
316   Not only the metrics constructed from the confusion matrix but also the error metrics as *mean*
317   *absolute error* (*MAE*), *mean squared error* (*MSE*), and *root mean squared error* (*RMSE*) are
318   visited within the scope of this comprehensive study. Li et al., in their privacy-preserving CF
319   approach, measure the performance with *RMSE* and *MAE* [20] like Nguyen et al. in [21]. *RMSE*
320   has a frequent usage and good reputation for measuring the error performance; for instance, in
321   *Netflix Prize* competition used as a vital indicator of implementation [84].
322   Even though the previous studies give priority typically to *F1-measure*, *precision*, *recall*, and
323   error-based measures [19]–[21], [85], some other performance metrics, which are relatively less-
324   recognizable in RS, also source robust decision-making. According to Chaaya et al., well-known
325   metrics cause significant biases, plus *markedness*, *informedness*, and *Matthews Correlation* are
326   noteworthy alternatives [86]. Since these preeminent metrics have a limited utilization in the
327   literature, we give them a priority throughout this work. They are presented consisting of
328   confusion matrix primitives in Table IV. The definitions of these metrics are briefly reviewed
329   below.
330
331   1) MARKEDNESS
332   The proportion of correct prediction is measured by *markedness*. This metric is free from the
333   imbalanced confusion matrix. The *markedness* scores in the range [-1,+1], and the related
334   formula is given in Eq. (16). *Markedness* can be a substitution for *precision* as a tool that shows
335   how the recommendation is in conjunction with the "chance" [17]. To the best of our knowledge,
336   *markedness* is one of the least visited metrics in the literature in the scope of RS science, even
337   though it supremely supplies information related to *positive* and *negative predictive values*. For
338   instance, this phenomenon is known as *DeltaP* in psychology literature, and Powers underlines
339   that *markedness* is considered a good predictor of human associative judgments [76], [87].

$$Markedness = Precision + Inverse\ Precision - 1 \qquad (16)$$

340
341   2) INFORMEDNESS
342   The second preeminent metric is *informedness*, which includes the *sensitivity* and its *inverse* as
343   in Eq. (17). *Informedness* scores in the same range, [-1,+1], like in *markedness* [17]. This metric
344   is also known as *Youden's Index*, as it differs from the *accuracy* in terms of imbalanced events
345   on the confusion matrix. The returned score defines a perfect prediction by +1 or indicates the
346   opposite by -1 [88]. The efforts of *informedness* in RS science are limited, even though there are
347   intriguing applications which practice this promising metric. Pilloni et al. perform *informedness*

348   in *e*-Health recommendation [89]. For hotel recommendations, *informedness* is also in use for the
349   performance check of the multi-criteria system. Ebadi and Krzyzak set two performance metrics
350   in general as *prediction-based* and *decision-based*, where *informedness* is treated in the scope of
351   decision-based metrics [90]. As another research field, Marciano et al. utilize *informedness* in the
352   context of genetics applications. It is accounted as a relative level of confidence [91]. Besides,
353   Layher et al. measure the performance of neuromorphic applications by assessing *informedness*
354   [92].

$$Informedness = Sensitivity + Inverse\ Sensitivity - 1 \tag{17}$$

355
356   3) MATTHEWS CORRELATION
357   *Matthews correlation* is a promising observation of binary labeling. As in Eq. (18), a wide
358   implicit observation is obtained with a score in [-1,+1] range. The interpretation of this metric
359   holds the three focus points as *perfect prediction*, *random prediction*, and *total disagreement*
360   between actual and predicted values. According to the score range, each corresponding focus
361   point is indicated via +1, 0, and -1 scores, respectively [93].

$$\begin{aligned}&Matthews\ Correlation\\&= \sqrt{\begin{array}{c}Positive\ Predictive\ Value \times True\ Positive\ Rate \times\\ True\ Negative\ Rate \times Negative\ Predictive\ Value\end{array}} - \sqrt{\begin{array}{c}False\ Discovery\ Rate \times False\ Negative\ Rate \times\\ False\ Positive\ Rate \times False\ Omission\ Rate\end{array}}\end{aligned} \tag{18}$$

362   *Matthews correlation* is nothing but the combination of *informedness* and *markedness*. The
363   former holds the understanding of how informed the classifier's decision with knowledge
364   compared to the "chance" [76]. In other words, *informedness* is paraphrased as a probability with
365   respect to a real variable rather than the "chance" [94]. On the other hand, *markedness* carries
366   information on how likely the prediction variable is marked by the *true* variable [92]. All in all,
367   *Matthews correlation* as a geometric mean of *informedness* and *markedness* shows the
368   correlation between the *prediction* and the *true* values. The occurrence of this metric is rare in
369   the literature, but there are still intriguing studies, like the one for diet recommendation [95].
370

Getting Started to Experiments

372   This section describes how we handle the data for a variety of similarity measurements.
373   Including the modifications over the equations, the overall algorithmic flow is introduced. The
374   premising *BNC* values are determined beforehand to be employed in the following section.
375   We first focus on the algorithm which is applied during the simulations. The algorithmic flow
376   can be a guidance for any prospective RS scientist to follow the basic steps. The procedures of
377   the proposed test package are summarized in *Algorithm 1*. The details related to several constant
378   parameters such as *test item count*, *cross-validation fold*, *neighbor counts*, and *threshold of liking*
379   are presented.
380   While running the algorithm for any user, five random items are taken into account. The *k-fold*
381   *cross-validation* technique is integrated with the implementation of repeatedly randomized test
382   attempts. In each independent analysis, the folds are shuffled, and the test items are alternated.
383   For reliability purposes, each test attempt is performed multiple times, then averaged.

384   On utilizing the fine-tuned ε parameter, an increased runtime may occur, especially in the bulk
385   tests. Several previous studies initiate this step size as $\varepsilon = 50$ [96], $\varepsilon = 10$ [23]–[25], or $\varepsilon = 5$
386   [11], [26]. Besides, Feng et al. measure different similarity measures by setting $\varepsilon = 5$; however,
387   they focus only on the error metrics [18]. Bag et al. illustrate the metrics' performance using
388   discrete BNC values as 5, 20, 50, and 100 [19]. In our test package, the fine-tuned neighbor step
389   size, ε, is set distinguishingly compared to the previous studies. $\varepsilon = 1$ is chosen to monitor the
390   sensitiveness of the tests and the neighboring interval sources smooth findings.
391   Furthermore, one of the main perspectives of previous efforts is to mitigate the computation time
392   by generalizing several parameters in the global scope of a development environment. At each
393   iteration of the test package, even though utilizing globally computed arguments reduces the
394   runtime of experiments, it lacks the required dynamic perspective for real-time application
395   imitation. Therefore, we perform *Algorithm 1* for all similarity equations inspecting this fallacy.
396   We apply the algorithm to two separate datasets investigating the overall performance. A fine-
397   tuned neighboring approach is performed for the best combination of *similarity equation* and
398   *performance metrics*. After selecting test items throughout steps 1-3, all the possible similarity
399   equations are recalled in step 4. A clear picture of equations together with corresponding
400   modifications for the dynamicity and weight significance are indicated in Table V. A parametric
401   neighborhood is applied from 1 to 100 users with a single increment, as shown in step 5. At each
402   loop iteration in step 6, the best neighbors are selected based on the similarity score, which is
403   sorted for all neighbor nominees, i.e., users who rated the test item. After the prediction
404   calculation in step 7, the performance is finally evaluated in step 8.
405   During the correlation computations, we warningly emphasize the possible shortcomings of
406   readily available functions in the computing platforms. Correlation methods mostly arrive as
407   inline functions in the development environments. However, we strongly suggest checking the
408   built-in functions for statistical parameter calculation, such as *mean* and *median*. It is advised not
409   to take the statistics of only co-rated items during the computations. It is more accurate to
410   include stats for all items in analyzing general behavior and shared characteristics, especially in
411   the context of dynamic RS [97].
412   For each similarity measure in Table V, the best-performing BNC value of the related
413   performance monitoring is discovered. Having set the BNC precisely, the observations are
414   summarized under various performance metrics presented in Tables VI and VII. Table VI shows
415   ML100K-based BNC values recorded for the dynamicity and weight significance approaches,
416   while Table VII stands for the same analyses of ML1M. These dataset-oriented analyses
417   facilitate guidance for further metric comparisons in the following section, where BNC values
418   inspected beforehand are utilized to interpret the adequate weight-metric combinations.
419   Moreover, the preliminary results in Tables VI and VII underline the effect of the SW method in
420   terms of BNC. For instance, PCC benefits the reduced BNCs having the best performance when
421   SW is applied. Except for *specificity* and *fallout* in ML100K, PCC gets the advantage of the SW
422   method. As presented in Table VI, PCC_{nIOI}^{sw} gives the top performance when $BNC = 17$ for
423   *markedness, Matthews correlation, F1-measure*, threshold-based error metrics and *accuracy*,

424 *Fowlkes–Mallows index*, *threat score*, *inverse precision*, *sensitivity*, *miss rate,* and *false omission*
425 *rate* in ML100K. Similarly, for ML1M, the same observation with $BNC = 31$ is valid for
426 *markedness, Matthews correlation, F1-measure*, threshold-based error metrics, *exact accuracy*,
427 *threshold accuracy, Fowlkes-Mallows index*, *threat score*, *inverse precision*, *sensitivity*, *miss*
428 *rate,* and *false omission rate*.
429

Results and Discussion

431 In this section, numerously repeated randomized tests are analyzed in the light of various
432 performance metrics. The effect of sensitive neighboring interval selection for different
433 similarity equations is performed, thereby measuring the importance of dynamicity and weight
434 significance.
435 In the following, first, we show the comparative performance plots of all similarity equations in
436 Table V using each individual metric. Preeminent metrics as *informedness*, *markedness*, and
437 *Matthews correlation*, together with *F1-measure* are utilized. ML100K and ML1M releases are
438 analyzed separately. For the plots, the *x-axis* stands for the fine-tuned *BNC*s, while the *y-axis* is
439 the related metric output.
440 The statistical approach is delicately depicted in this work to set a dynamic environment, which
441 requires a more adaptive procedure. The specious results based on hypothetical computations, in
442 fact, can only demarcate the maximum achievable top-performance of the dynamicity concept.
443 We prove how the dynamicity deviates from these maximum reachable results. In the following
444 figures, dashed lines present the theoretical perspective by including the global-only stats
445 causing a fallacy. Still, solid lines are for the dynamicity with *nIOI* approach. In addition, lines
446 having diamond marks illustrate the results free from the *SW* approach, while *SW* adjustment can
447 be monitored through the unmarked lines.
448 In the following illustrations, performance plots of ML100K and ML1M are given for the
449 preeminent metrics and *F1-measure*. Each row of the subplots is compared with an equation-
450 dependent perspective. Analyzing ML100K, the similarity equation only with *SW* modification
451 gives the best results for *PCC* lines in black color. In *MRC*, lines having green color, the same
452 dominance for the *SW* methodology can be observed through all metrics. Even though *SW* does
453 not boost the performance of *COS* in all metrics, plots with *SW* in *JAC* show close performance
454 to the ones without *SW*.
455 In a comparative manner, we present the same metric performance throughout the similarity
456 measures in the context of ML1M, as given in Fig. 2. Including only the dynamic *COS*, all other
457 similarities with *SW* increase the *F1-measure* performance. However, the performance in
458 *informedness* diminishes compared to Fig. 1. The top-performing and the least-performing lines
459 in each similarity measure stay the same for *markedness*, *Matthews correlation*, and *F1-measure*
460 like in ML100K. Nonetheless, for *informedness*, the effect of *SW* in *PCC* and *MRC* interchanges
461 the least-performing similarity equation. Furthermore, the dynamicity brings the same expected
462 outcomes in ML1M analysis.

463   Fig. 3 depicts the overall comparison for the compelled dynamicity together with applied weight
464   significance. According to Fig. 3 (a), addressing ML100K, PCC is notably in the leading position
465   compared to other similarity measurements. The ranking for the rest of the similarity measures is
466   hard to be generalized as the lines interchangeably depend on the BNC. For *markedness*, MRC
467   starts with better performance for fewer BNCs; however, the trend reverses for $BNC > 27$.
468   Besides, other measurements than MRC outperform for $BNC > 40$. Similar behavior is valid for
469   *informedness* and *Matthews correlation*. However, BNC threshold of each is relatively greater.
470   Approximately, $BNC > 75$ for *informedness* and $BNC > 60$ for *Matthews correlation* decay the
471   performance of MRC. In *F1-measure*, relatively stable performance is obtained for the measures
472   when $BNC > 10$. In terms of equation rankings, *F1-measure* can be considered as the most
473   stable metric not dependent on BNC for ML100K. This makes it eminent for the similarity
474   measure performance comparison without further BNC consideration.
475   In Fig. 3 (b), the same hybrid monitoring of *nIOI* and *SW* is presented for ML1M. The top-
476   performing lines for the aforementioned preeminent metrics are still obtained via PCC. COS,
477   compared to the others, is in a relatively poor performance for *informedness* metric. The
478   performance ranking of similarity equations stays more stable as a function of BNC when
479   compared to ML100K. Only in *informedness,* there occur slight interchanges in between MRC
480   and JAC. However, in all others, relative performances are independent of BNC. In contrast to
481   the significant interchangings in ML100K, performance metrics keep their relative positions in
482   ML1M. This stability finding can be a general interpretation and be concluded that the more the
483   dataset size is, the more stable performance occurs.
484   After the comparative analysis of different similarity equations together with preeminent metrics,
485   the following figures illustrate the extensive analysis of other metrics frequently evaluated in the
486   literature. First, in Fig. 4, ML100K plots considering the hybrid monitoring are presented.
487   For accuracy-based metrics, both the *exact* (sensitive rating prediction) and *binary* (*liked* or
488   *disliked* labeling) performance are monitored. PCC leads in either case, putting a relatively
489   higher accuracy margin around 0.05 for both metrics. It is also experienced another accuracy
490   calculation extensively in this paper. Considering *balanced accuracy*, PCC outperforms for all
491   BNCs, while MRC diminishes. Similarly, the error metrics are measured both in an *exact* and
492   *binary* sense. As expected, *binary prediction error* rates are lower than the *exact prediction error*
493   rates. Both are plotted using *MAE*, *MSE*, and *RMSE*. In *binary prediction error*, PCC gives the
494   lowest possible error rates. Nonetheless, for the *exact accuracy* metric, the top performance for
495   low error rates is interchangeable based on the neighborhood. Approximately, $BNC > 40$,
496    $BNC > 20$, and $BNC > 20$ respectively for *MSE*, *MAE*, and *RMSE*, yield the lower error rates
497   with JAC measure.
498   Besides, *Fowlkes–Mallows index* shows that PCC and COS outperform, while in general, MRC
499   has poor performance. In *threat score*, the hits of user dislikes are not included; however, the
500   ratio of liking matches concerning the misses is checked. Similarity measurement rankings are
501   relatively stable after $BNC = 10$ and PCC is the adequate measure, while MRC falls behind.

502 This study also discusses different performance checks based on interdisciplinary applications
503 discussed previously. In the context of RS, we propose to apply *prevalence threshold* that
504 sources compound information, including several associated metrics inferred as
505 $(\sqrt{sensitivity \times fallout} - fallout) / Informedness$. Similar to the error metrics, the less
506 *prevalence threshold* value is observed, the more exactitude is accomplished. While higher
507 values are obtained in COS compared to others, PCC has proven its superiority with lower rates.
508 One main concern of some metrics, like *sensitivity* and *specificity*, is to give information on how
509 likely top-*n* items match the user taste or vice versa. Correctly identified positives, i.e., *sensitivity*
510 performs well for the lower BNCs, and COS measure leads with a peak around $BNC = 9$. On the
511 other hand, correctly identified negatives, i.e., *specificity* gets distinctively better as the neighbor
512 count increases for COS and JAC, while PCC and MRC are relatively stable.
513 The last two rows of the whole subplots are the complementary metric couples. The metrics as
514 *sensitivity & miss rate*, *specificity & fallout*, *precision & false discovery rate*, and *inverse*
515 *precision & false omission rate* complete each other. In *specificity*, *precision*, and *inverse*
516 *precision*, PCC performs adequately, which can be cross-checked from *fallout, false discovery*
517 *rate*, and *false omission rate* in turn.
518 The ML1M evaluation is presented comparatively to the previous findings of ML100K. In Fig.
519 5, the analyses are illustrated for the same evaluated metrics. The trend in *exact accuracy* is
520 similar to ML100K, with greater scores. PCC puts a margin, while the others perform closer to
521 each other with slightly lower values compared to PCC. Besides, *exact prediction error* metrics
522 generally result in a reduced numerical range. By a majority, the most erroneous metric is COS,
523 which is valid both for the *exact* and *binary prediction error* metrics. The error performance in
524 ML1M is relatively stable, and PCC is still the less fragile metric for *binary prediction*. In *binary*
525 analyses, similarity measures are homogenously ranked, again with the dominance of PCC.
526 Furthermore, *Fowlkes–Mallows index* shows an increased range of scoring with respect to
527 ML100K findings. In *threat score*, it is observed that MRC significantly improves compared to
528 others. In *prevalence threshold*, COS has a higher margin than the others compared to the
529 performances in the previous analysis. Moreover, even though COS has a good *sensitivity*
530 observation, it performs worse in *specificity* and *precision* as in the previous findings. That is,
531 COS suffer from *true negative rate* and *positive predictive value*.
532 The monitoring of smooth *sensitivity* in ML1M can be important feedback since it is a
533 component of some compound metrics. On the other hand, an indicative finding from the
534 comparison of both releases is the behavior of the *specificity* and *fallout* metric couples. In
535 ML100K, a relatively more stable distribution is monitored for increasing BNC values, while in
536 ML1M, the behavior becomes unstable. Lastly, while JAC for *precision* goes up in the ranking
537 compared to the previous findings, the situation is the opposite for *inverse precision*.
538 Having represented all the plots, we summarize test results using a tabular structure for a
539 compact heat-map presentation. In Tables VIII and IX, the performance metrics over each

540 similarity measurement are visualized in a colored format[2]. As preliminarily explained in the
541 third section, the selected *BNC*s giving the top performances are added into the tables
542 highlighting the main motivation of our work: *the decision of adequate weight-metric*
543 *combinations*. Each metric is processed through a column-wise coloring to make the comparison
544 easier. The cells in green shades indicate the effectiveness of the proper combination. We present
545 results both using *SW*-induced dynamic equations and the plain dynamicity. Hence, the effect of
546 boosting the weights can be monitored.
547 All other test outcomes can be found in our code repository[3]. We have prepared a fully detailed
548 supplementary material to comprise all the outcomes. Any RS researcher can benefit from the
549 prepared document for such purposes, e.g., the selection of *BNC*s, enhanced similarity measure
550 conditions, etc. Every single iteration in the test package is logged into the aforementioned
551 document.
552 The colorized tables are organized by grouping the column names attributed to the performance
553 metrics. The first group is the *preeminent metrics*, to which we give priority. Then, the *error-*
554 *based metrics* are combined. The third group is the *accuracy-based metrics*, and the final group
555 is the rest, which is frequently used in the literature, including interdisciplinary applications. This
556 way of representation determines the consistency of each similarity equations in the sense of
557 grouping. For instance, *JAC* equation with *SW* generally keeps its stability in each metric group
558 concerning the smooth coloring. Remarkably, the homogeneous scoring highlights the overall
559 performance of any similarity equation.
560 In a general view, if an RS design targets only the recommendations of preferable items, then
561 *COS* may be a suitable similarity measure. Metrics, which do not address *TN* values such as *F1-*
562 *measure*, *Fowlkes–Mallows index*, *threat score*, *sensitivity*, and *miss rate,* feedback the indicative
563 scores when combined with *COS*. On the other hand, if *COS* is decorated with the *SW* approach,
564 it is observed that the homogeneity of the heat-map tones while transiting between metrics
565 deteriorates. Although some certain metrics get the advantage of *SW*, *COS* can be concluded that
566 it is not totally compatible with the *SW* approach. Conversely, the beneficial impact of *SW* can
567 broadly be observed through all other equations. In general, *PCC* is the most appealing metric as
568 it highlights its previous fame in the literature. Even though *PCC* deals with the linear
569 correlation, it fits suitably into the 5-star rating analyses. The one adequate similarity equation
570 can be generalized as *PCC* with *SW* while *MRC* without *SW* becoming the least performing
571 equation as the harsh red background color indicates. *MRC* free from *SW* shows the most
572 inadequate performance in Table VIII. Therefore, the weighting method on the *MRC* utilization
573 is highly recommended for ML100K, while the case is slightly vice versa for ML1M.
574
575 ## Conclusions

---

[2] The fractional values in the table are displayed based on three significant digits. The heat-map coloring is achieved according to full precision.
[3] The supplementary material containing the complete results of the whole test package can be accessed from the repository given in the Acknowledgement.

576   This comprehensive study presents an experimental perspective to interpret the interrelations of
577   similarity equations and performance metrics. The most indicative highlight of this article is to
578   underline the necessity of the dynamic approach by performing independent computations. The
579   misleading effect of the test item bias is emphasized in our analyses. We unveil how this pitfall
580   can demarcate the results. The test item bias in the training phase brings hazardous outcomes,
581   and the upper limit a system can reach is drawn. In addition, another highlight is the impact of
582   the similarity weighting. All the combinations of the modifications are monitored
583   experimentally. Overall evaluation is inferred from multiple simulations to ensure suppressing
584   any outlier and to stabilize the outcome. Besides, we present a fine-tuned neighborhood analysis,
585   which also takes place in weight-metric combinations. It can be deduced the limit of *BNC* in our
586   graphical interpretations. Furthermore, our remarks are profoundly demonstrated in heat-map
587   tables with the best performing neighborhood surveyed by intercrossing similarity equations and
588   specific metrics. All in all, it is finally emphasized that the back-end of any RS design can be
589   developed with the same procedures applied throughout this paper. Any dataset can adaptively
590   be examined via the open-source code of our framework as indicated in the Acknowledgement.
591   Starting from the fine-tuned neighborhood, the test item bias mitigation approach can thoroughly
592   be followed *with* and *without* the *SW* method. We believe that any further studies in RS science
593   can benefit from our indicated remarks. For future work, any other metadata or features, such as
594   user demographics and item details, can be included to enhance our framework.
595

Acknowledgements

597   The reproducible code run can directly be executed on the Code Ocean platform. Moreover, both
598   the open-source code of the whole package and the supplementary material,
599   *supplementaryMaterial_allResults.xlsx,* are attached to the
600   https://github.com/savasokyay/AdequateWeightMetricDynamicCF repository. The
601   supplementary material contains detailed results, including all neighbors from the top-
602   performing to the least-performing. For the easiness of result monitoring, any reader can
603   manipulate this file by filtering and sorting the fields.

604

References

606   [1]   P. Kazienko, K. Musiał, and T. Kajdanowicz, "Multidimensional social network in the
607   social recommender system," *IEEE Trans. Syst. Man, Cybern. Part A: Systems and Humans*, vol.
608   41, no. 4, pp. 746–759, Jul. 2011.
609   [2]   A. Calero Valdez and M. Ziefle, "The users' perspective on the privacy-utility trade-offs
610   in health recommender systems," *Int. J. Hum. Comput. Stud.*, vol. 121, pp. 108–121, Jan. 2019.
611   [3]   D. Wang, Y. Liang, D. Xu, X. Feng, and R. Guan, "A content-based recommender
612   system for computer science publications," *Knowledge-Based Syst.*, vol. 157, pp. 1–9, Oct. 2018.
613   [4]   I. Andjelkovic, D. Parra, and J. O'Donovan, "Moodplay: Interactive music
614   recommendation based on artists' mood similarity," *Int. J. Hum. Comput. Stud.*, vol. 121, pp.
615   142–159, Jan. 2019.

616 [5]   Ò. Celma and P. Herrera, "A new approach to evaluating novel recommendations," *in*
617 *RecSys'08 Proc. 2008 ACM Conf. Recomm. Syst.*, Lausanne, Switzerland, 2008, pp. 179–186.
618 [6]   M. N. Moreno, S. Segrera, V. F. López, M. D. Muñoz, and A. L. Sánchez, "Movie
619 recommendation framework using associative classification and a domain ontology," in *Hybrid*
620 *Artificial Intelligent Systems*, *8th International Conf.*, Salamanca, Spain, 2013, pp. 122–131.
621 [7]   F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles,
622 methods and evaluation," *Egypt. Informatics J.*, vol. 16, no. 3, pp. 261–273, Nov. 2015.
623 [8]   Y. Wang, M. Wang, and W. Xu, "A sentiment-enhanced hybrid recommender system for
624 movie recommendation: A big data analytics framework," *Wirel. Commun. Mob. Comput.*, vol.
625 2018, pp. 1–9, Mar. 2018.
626 [9]   Grouplens, "MovieLens," 1992, [Online]. Available:
627 https://grouplens.org/datasets/movielens/
628 [10]   Netflix, "Netflix price dataset," 2009, [Online]. Available: https://www.netflixprize.com/
629 [11]   M. A. Ghazanfar and A. Prugel-Bennett, "Novel significance weighting schemes for
630 collaborative filtering: Generating improved recommendations in sparse environments," in
631 *International Conference on Data Mining, DMIN 2010*, Las Vegas, USA, 2010.
632 [12]   C. A. Levinas, "An analysis of memory based collaborative filtering recommender
633 systems with improvement proposals," M.S. thesis, Universitat Politècnica de Catalunya, 2014.
634 [13]   B. Zhang and B. Yuan, "Improved collaborative filtering recommendation algorithm of
635 similarity measure," in *AIP Conf. Proc.*, Hangzhou, China, 2017.
636 [14]   M. Gao, Y. Fu, Y. Chen, and F. Jiang, "User-weight model for item-based
637 recommendation systems," *J. Softw.*, vol. 7, no. 9, pp. 2133–2140, Sept. 2012.
638 [15]   A. Bellogín, P. Castells, I. Cantador, "Neighbor selection and weighting in user-based
639 collaborative filtering: A performance prediction approach," *ACM Trans. on the Web*, no. 1,
640 Mar. 2014.
641 [16]   L. Zhang, Q. Wei, L. Zhang, B. Wang, and W. H. Ho, "Diversity balancing for two-stage
642 collaborative filtering in recommender systems," *Appl. Sci.*, vol. 10, no. 4, pp. 1–16, Feb. 2020.
643 [17]   G. Schröder, M. Thiele, and W. Lehner, "Setting goals and choosing metrics for
644 recommender system evaluations," in *UCERSTI2 workshop at the 5th ACM conference on*
645 *recommender systems*, Chicago, USA, 2011.
646 [18]   J. Feng, X. Fengs, N. Zhang, and J. Peng, "An improved collaborative filtering method
647 based on similarity," *PLoS One*, vol. 13, no. 9, pp. 1–18, Sept. 2018.
648 [19]   S. Bag, S. K. Kumar, and M. K. Tiwari, "An efficient recommendation generation using
649 relevant Jaccard similarity," *Inf. Sci.*, vol. 483, pp. 53–64, Jan. 2019.
650 [20]   D. Li, C. Chen, Q. Lv, L. Shang, Y. Zhao, T. Lu, and N. Gu, "An algorithm for efficient
651 privacy-preserving item-based collaborative filtering," *Futur. Gener. Comput. Syst.*, vol. 55, pp.
652 311–320, Dec. 2014.
653 [21]   L. V. Nguyen, M. S. Hong, J. J. Jung, and B. S. Sohn, "Cognitive similarity-based
654 collaborative filtering recommendation system," *Appl. Sci.*, vol. 10, no. 12, pp. 1–14, Jun. 2020.

655 [22]   T. Arsan, E. Koksal, and Z. Bozkus, "Comparison of collaborative filtering algorithms
656 with various similarity measures for movie recommendation," *Int. J. Comput. Sci. Eng. Appl.*,
657 vol. 6, no. 3, pp. 1–20, Jun. 2016.
658 [23]   J. L. Sánchez, F. Serradilla, E. Martínez, and J. Bobadilla, "Choice of metrics used in
659 collaborative filtering and their impact on recommender systems," in *2nd IEEE Int. Conf. Digit.*
660 *Ecosyst. Technol.*, Phitsanuloke, Thailand, 2008, pp. 432–436.
661 [24]   H. Liu, Z. Hu, A. Mian, H. Tian, and X. Zhu, "A new user similarity model to improve
662 the accuracy of collaborative filtering," *Knowledge-Based Syst.,* vol. 56, pp. 156–166, Nov.
663 2013.
664 [25]   B. H. Huang and B. R. Dai, "A weighted distance similarity model to improve the
665 accuracy of collaborative recommender system," in *IEEE Int. Conf. on Mob. Data Manag.,*
666 Pittsburgh, PA, USA, 2015, pp. 104–109.
667 [26]   S. B. Sun, Z. H. Zhang, X. L. Dong, H. R. Zhang, T. J. Li, L. Zhang, and F. Min,
668 "Integrating triangle and Jaccard similarities for recommendation," *PLoS One*, vol. 12, no. 8, pp.
669 1–16, Aug. 2017.
670 [27]   F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM*
671 *Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 1–19, Dec. 2015.
672 [28]   U. Kuzelewska and R. Ducki, "Collaborative filtering recommender systems in music
673 recommendation," *Adv. in Comp. Sci. Res.*, vol. 10, pp. 67–79, 2013.
674 [29]   M. M. Mukaka, "Statistics corner: A guide to appropriate use of correlation coefficient in
675 medical research," *Malawi Med. J.*, vol. 24, no. 3, pp. 69–71, Sept. 2012.
676 [30]   H. Akoglu, "User's guide to correlation coefficients," *Turkish J. Emerg. Med.*, vol. 18,
677 no. 3, pp. 91–93, Sept. 2018.
678 [31]   H. A. Miot, "Correlation analysis in clinical and experimental studies," *J. Vasc. Bras.*,
679 vol. 17, no. 4, pp. 275–279, 2018.
680 [32]   N. Kurmashov, K. Latuta, and A. Nussipbekov, "Online book recommendation system,"
681 in *Int. Conf. Electron. Comput. Computa. ICECCO*, Almaty, Kazakhstan, 2016.
682 [33]   N. Sivaramakrishnan, V. Subramaniyaswamy, S. Arunkumar, A. Renugadevi, and K. K.
683 Ashikamai, "Neighborhood-based approach of collaborative filtering techniques for book
684 recommendation system," *Int. J. Pure Appl. Math.*, vol. 119, no. 12, pp. 13241–13250, 2018.
685 [34]   T. Q. Lee, Y. Park, and Y. T. Park, "A time-based approach to effective recommender
686 systems using implicit feedback," *Expert Syst. Appl.*, vol. 34, no. 4, pp. 3055–3062, May 2008.
687 [35]   J. Lee, K. Lee, and J. G. Kim, "Personalized academic research paper recommendation
688 system," 2013, Accessed on: Oct., 2020, [Online]. Available: http://arXiv:1304.5457
689 [36]   Adiyansjah, A. A. S. Gunawan, and D. Suhartono, "Music recommender system based on
690 genre using convolutional recurrent neural networks," in *4th Int. Conf. on Comp. Sci. Comput.*
691 *Intelli. (ICCSCI)*, vol. 157, 2019, pp. 99–109.
692 [37]   Z. Cataltepe and B. Altinel, "Music recommendation by modeling user's preferred
693 perspectives of content, singer/genre and popularity," in *Collaborative and Social Information*

694 *Retrieval and Access: Techniques for Improved User Modeling*, edited by Max Chevalier, et al.,
695 IGI Global, 2009, pp. 203–221.
696 [38] N. Sigg, "An investigation into the relationship between music preference, personality
697 and psychological wellbeing," M.S. thesis, Auckland University of Technology, 2009.
698 [39] D. Shepherd and N. Sigg, "Music preference, social identity, and self-esteem," *Music*
699 *Perception*, vol. 32, no. 5, pp. 507–514, 2015.
700 [40] S. Dhawan, K. Singh, and Jyoti, "High rating recent preferences based recommendation
701 system," in *4th Int. Conf. on Eco. Comp. and Comm. Sys. (ICECCS)*, vol. 70, 2015, pp. 259–264.
702 [41] K. Madadipouya, "A location-based movie recommender system using collaborative
703 filtering," *Int. J. Found. Comput. Sci. Technol.*, vol. 5, no. 4, pp. 13–19, Jul. 2015.
704 [42] L. Sheugh and S. H. Alizadeh, "A note on Pearson correlation coefficient as a metric of
705 similarity in recommender system," in *AI & Robot. (IRANOPEN)*, Qazvin, Iran, 2015.
706 [43] K. R. Kim, J. H. Lee, J. H. Byeon, and N. M. Moon, "Recommender system using the
707 movie genre similarity in mobile service," in *4th Int. Conf. Multimed. Ubiquitous Eng.*, Cebu,
708 Philippines, 2010.
709 [44] T. G. Hwang, C. S. Park, J. H. Hong, and S. K. Kim, "An algorithm for movie
710 classification and recommendation using genre correlation," *Multimed. Tools Appl.*, vol. 75, pp.
711 12843–12858, Apr. 2016.
712 [45] Z. Tan and L. He, "An efficient similarity measure for user-based collaborative filtering
713 recommender systems inspired by the physical resonance principle," *IEEE Access*, vol. 5, pp.
714 27211–27228, 2017.
715 [46] F. Garcin, B. Faltings, R. Jurca, and N. Joswig, "Rating aggregation in collaborative
716 filtering systems," in *RecSys'09 - Proc. 3rd ACM Conf. Recomm. Syst.*, New York, USA, 2009,
717 pp. 349–352.
718 [47] R. H. Singh, S. Maurya, T. Tripathi, T. Narula, and G. Srivastav, "Movie
719 recommendation system using cosine similarity and KNN," *Int. J. Eng. Adv. Technol.*, vol. 9, no.
720 5, pp. 556–559, Jun. 2020.
721 [48] I. S. Wahyudi, A. Affandi, and M. Hariadi, "Recommender engine using cosine similarity
722 based on alternating least square-weight regularization," in *15th Int. Conf. Qual. Res. Int. Symp.*
723 *Electr. Comput. Eng.*, Nusa Dua, Bali, Indonesia, 2017, pp. 256–261.
724 [49] S. Philip, P. B. Shola, and A. O. John, "Application of content-based approach in
725 research paper recommendation system for a digital library," *Int. J. Adv. Comput. Sci. Appl.*, vol.
726 5, no. 10, pp. 37–40, 2014.
727 [50] S. Ahmad and M. T. Afzal, "Combining metadata and co-citations for recommending
728 related papers," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 28, pp. 1519–1534, 2020.
729 [51] A. Samad, M. A. Islam, M. A. Iqbal, and M. Aleem, "Centrality-based paper citation
730 recommender system," *EAI Endorsed Trans. on Ind. Netw. and Intell. Sys.*, vol. 6, no. 19, pp. 1–
731 10, Jun. 2019.
732 [52] R. LVN, Q. Wang, and J. D. Raj, "Recommending news articles using cosine similarity
733 function," in *Proc. SAS Ins. Global Forum*, 2014, pp. 1–7.

734 [53] F. Aiolli, "Efficient top-N recommendation for very large scale binary rated datasets," in
735 *RecSys'13 - Proc. 7th ACM Conf. Recomm. Syst.*, Hong Kong, China, 2013, pp. 273–280.
736 [54] L. Meilian, Q. Zhen, C. Yiming, L. Zhichao, and W. M. State, "Scalable news
737 recommendation using multi-dimensional similarity and Jaccard-Kmeans clustering," *J. Syst.*
738 *Softw.*, vol. 95, pp. 242–251, May 2014.
739 [55] A. Rana and K. Deeba, "Online book recommendation system using collaborative
740 filtering (with Jaccard similarity)," *J. Phys. Conf. Ser.*, vol. 1362, pp. 1–8, 2019.
741 [56] L. Zahrotun, "Comparison Jaccard similarity, cosine similarity and combined both of the
742 data clustering with shared nearest neighbor method," *Comput. Eng. Appl. J.*, vol. 5, no. 1, pp.
743 11–18, Feb. 2016.
744 [57] N. F. AL-Bakri and S. H. Hashim, "A study on the accuracy of prediction in
745 recommendation system based on similarity measures," *Baghdad Sci. J.*, vol. 16, no. 1, pp. 263–
746 269, Mar. 2019.
747 [58] K. Pearson, "Contributions to the mathematical theory of evolution," *Philosophical*
748 *Trans. of the Royal Soc. of London*, vol. (A.)185, pp. 71–110, Nov. 1893.
749 [59] G. L. Shevlyakov, "On robust estimation of a correlation coefficient," *J. Math. Sci.*, vol.
750 83, no. 3, pp. 434–438, 1997.
751 [60] G. Shevlyakov and P. Smirnov, "Robust estimation of the correlation coefficient: An
752 attempt of survey," *Austrian J. of Stat.*, vol. 40, no. 1&2, pp. 147–156, 2011.
753 [61] V. R. Pasman and G. L. Shevlyakov, "Robust methods of estimating the correlation
754 coefficient," *Avtomat. i Telemekh.*, Issue 3, pp. 70–80, 1987.
755 [62] K. Pearson, "Contributions to the mathematical theory of evolution-II. Skew variation in
756 homogeneous material," *Philosophical Trans. of the Royal Soc. of London*, vol. (A.)186, pp.
757 343–414, Jan. 1895.
758 [63] M. Sato, "Some remarks on the mean, median, mode and skewness," *Austrian J. of Stat.*,
759 vol. 39, no. 2, pp. 219–224, Jan. 1997.
760 [64] M. Gao, Z. Wu, and F. Jiang, "Userrank for item-based collaborative filtering
761 recommendation," *Inf. Process. Lett.*, vol. 111, no. 9, pp. 440–446, Apr. 2011.
762 [65] A. Saric, M. Hadzikadic, and D. Wilson, "Alternative formulas for rating prediction using
763 collaborative filtering," in *Int. Symp. on Meth. for Intell. Sys. (ISMIS)*, 2009, pp. 301–310.
764 [66] H. Zeybek and C. Kaleli, "Dynamic k neighbor selection for collaborative filtering,"
765 *Anadolu Univ. J. Sci. Technol. A - Appl. Sci. Eng.*, vol. 19, no. 2, pp. 303–315, 2018.
766 [67] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering
767 recommendation algorithms," in *Proc. of the 10th Int. Conf. on WWW*, Hong Kong, 2001, pp.
768 285–295.
769 [68] J. Wu, L. Chen, Y. Feng, Z. Zheng, M. C. Zhou, and Z. Wu, "Predicting quality of
770 service for selection by neighborhood-based collaborative filtering," *IEEE Trans. Syst. Man,*
771 *Cybern.: Sys.*, vol. 43, no. 2, pp. 428–439, Mar. 2013.

772 [69]   P. K. Singh, M. Sinha, S. Das, and P. Choudhury, "Enhancing recommendation accuracy
773 of item-based collaborative filtering using Bhattacharyya coefficient and most similar item,"
774 *Appl. Intell.*, vol. 50, pp. 4708–4731, Aug. 2020.
775 [70]   S. Okyay and S. Aygün, "A study of static and dynamic significance weighting
776 multipliers on the Pearson correlation for collaborative filtering," *European J. of Sci. and Tech.*,
777 vol. Spec. Iss., pp. 270–275, 2020.
778 [71]   M. Raeesi and M. Shajari, "An enhanced significance weighting approach for
779 collaborative filtering," in *6th Int. Symp. Telecommun. (IST'12)*, Tehran, Iran, 2012, pp. 1165–
780 1169.
781 [72]   J. Herlocker, J. A. Konstan, and J. Riedl, "An empirical analysis of design choices in
782 neighborhood-based collaborative filtering algorithms," *Inf. Retr.*, vol. 5, no. 4, pp. 287–310,
783 Jan. 2002.
784 [73]   M. R. McLaughlin and J. L. Herlocker, "A collaborative filtering algorithm and
785 evaluation metric that accurately model the user experience," in *Proc. of the 27th Annual Int.
786 ACM SIGIR Conf. on Res. and Dev. in Info. Retr.*, Sheffield, United Kingdom, 2004, pp. 329–
787 336.
788 [74]   J. Weng, C. Miao, and A. Goh, "Improving collaborative filtering with trust-based
789 metrics," in *Proc. of ACM Symp on Appl. Comp.*, Dijon, France, 2006, pp. 1860–1864.
790 [75]   C. S. Hwang and Y. P. Chen, "Using trust in collaborative filtering recommendation," in
791 *Int. Conf. on Ind., Eng. and other App.s of Appl. Intell Sys IEA/AIE*, Kyoto, Japan, 2007, pp.
792 1052–1060.
793 [76]   D. M. W. Powers, "Evaluation: From precision, recall and F-factor to ROC,
794 informedness, markedness & correlation," *Technical Report SIE-07-001, School of Info. and
795 Eng.,* Flinders Uni., Adelaide, South Australia, pp. 1–24, Dec. 2007.
796 [77]   G. Shani and A. Gunawardana, "Evaluating recommendation systems," *Recomm. Syst.
797 Handb.*, pp. 257–297, 2011.
798 [78]   M. Mukhtar, S. S. Ali, S. A. Boshara, A. Albertini, S. Monnerat, P. Bessell, Y. Mori, Y.
799 Kubota., J. M. Ndung'u, and I. Cruz, "Sensitive and less invasive confirmatory diagnosis of
800 visceral leishmaniasis in Sudan using loop-mediated isothermal amplification (LAMP)," *PLoS
801 Negl. Trop. Dis.*, vol. 12, no. 2, pp. 1–14, Feb. 2018.
802 [79]   K. R. Devi, J. Bhavithra, and A. Saradha, "Personalized nutrition recommendation for
803 diabetic patients using improved K-means and Krill-Herd optimization," *Int. J. Sci. Tech. Res.*,
804 vol. 9, no. 3, pp. 1076–1083, Mar. 2020.
805 [80]   S. K. Panda, S. K. Bhoi, and M. Singh, "A collaborative filtering recommendation
806 algorithm based on normalization approach," *J. Ambient Intell. Humaniz. Comput.*, vol. 11, no.
807 11, pp. 4643–4665, Jan. 2020.
808 [81]   R. J. Hogan, C. A. T. Ferro, I. T. Jolliffe, and D. B. Stephenson, "Equitability revisited:
809 Why the 'equitable threat score' is not equitable," *Weather and Forecasting*, vol. 25, no. 2, pp.
810 710–726, Apr. 2010.

811 [82] J. Balayla, "Prevalence threshold (φ_e) and the geometry of screening curves," *PLoS One*,
812 vol. 15, no. 10, pp. 1–12, Oct. 2020.
813 [83] J. Balayla, A. Lasry, Y. Gil, and A. V. Perel, "Prevalence threshold and temporal
814 interpretation of screening tests: The example of the SARS-CoV-2 (COVID-19) pandemic,"
815 2020, Accessed on: Oct., 2020, [Online]. Available: https://doi.org/10.1101/2020.05.17.
816 20104927
817 [84] R. M. Bell and Y. Koren, "Lessons from the Netflix prize challenge," *ACM SIGKDD*
818 *Explor. Newsl.*, vol. 9, no. 2, pp. 75–79, Dec. 2007.
819 [85] W. Hong-Xia, "An improved collaborative filtering recommendation algorithm," in *IEEE*
820 *4th Int. Conf. Big Data Anal. (ICBDA)*, Suzhou, China, 2019, pp. 431–435.
821 [86] G. Chaaya, E. Metais, J. B. Abdo, R. Chiky, J. Demerjian, and K. Barbar, "Evaluating
822 non-personalized single-heuristic active learning strategies for collaborative filtering
823 recommender systems," in *16th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Cancun, 2017, pp.
824 593–600.
825 [87] D. R. Shanks, "Is human learning rational?," *Q. J. Exp. Psychol. Sect. A: Human*
826 *Experimental Psychology*, vol. 48, no. 2, pp. 257–279, 1995.
827 [88] R. W. Broadley, J. Klenk, S. B. Thies, L. P. J. Kenney, and M. H. Granat, "Methods for
828 the real-world evaluation of fall detection technology: A scoping review," *Sensors MDPI*, vol.
829 18, no. 7., pp. 1–28, Jun. 2018.
830 [89] P. Pilloni, L. Piras, L. Boratto, S. Carta, G. Fenu, and F. Mulas, "Recommendation in
831 persuasive eHealth systems: An effective strategy to spot users' losing motivation to exercise,"
832 in *2nd Int. Works. on Health Rec. Sys., HealthRecSys*, 2017.
833 [90] A. Ebadi and A. Krzyzak, "A hybrid multi-criteria hotel recommender system using
834 explicit and implicit feedbacks," *Int. J. Sci. Comp. Inf. Eng.*, vol. 10, no. 8, pp. 1450–1458, 2016.
835 [91] M. A. Marciano, V. R. Williamson, and J. D. Adelman, "A hybrid approach to increase
836 the informedness of CE-based data using locus-specific thresholding and machine learning,"
837 *Forensic Sci. Int. Genet.*, vol. 35, no. 1872-4973, pp. 26–37, Jul. 2018.
838 [92] G. Layher, T. Brosch, and H. Neumann, "Real-time biologically inspired action
839 recognition from key poses using a neuromorphic architecture," *Front. Neurorobot.*, vol. 11, no.
840 13, pp. 1–21, Mar. 2017.
841 [93] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data
842 using Matthews correlation coefficient metric," *PLoS One*, vol. 12, no. 6, pp. 1–17, Jun. 2017.
843 [94] D. M. W. Powers, "A computationally and cognitively plausible model of supervised and
844 unsupervised learning," in *Advan. Brain Insp. Cogn. Sys.*, 2103, pp. 145–156.
845 [95] K. R. Devi, J. Bhavithra, and A. Saradha, "Diet recommendation for glycemic patients
846 using improved K-means and Krill-Herd optimization," *ICTACT J. on S. Comp.*, vol. 10, no. 03,
847 pp. 2096–2101, Apr. 2020.
848 [96] J. Wang, A. P. De Vries, and M. J. T. Reinders, "Unifying user-based and item-based
849 collaborative filtering approaches by similarity fusion," in *the 29th Annu. Int. ACM SIGIR Conf.*
850 *Res. Dev. Inf. Retr.*, Seattle, Washington, USA, 2006, pp. 501–508.

851 [97]  S. Okyay and S. Aygun, "A Significant Fallacy of Built-in Correlation Functions in
852 Recommender Systems," in *3rd IEEE International Congress on Human-Computer Interaction,*
853 *Optimization and Robotic Applications. (HORA)*, 2021, pp. xx–xx (accepted).

Figure 1

The evaluation over ML100K: similarity weight and preeminent metric combination to compare the dynamicity and weight significance.

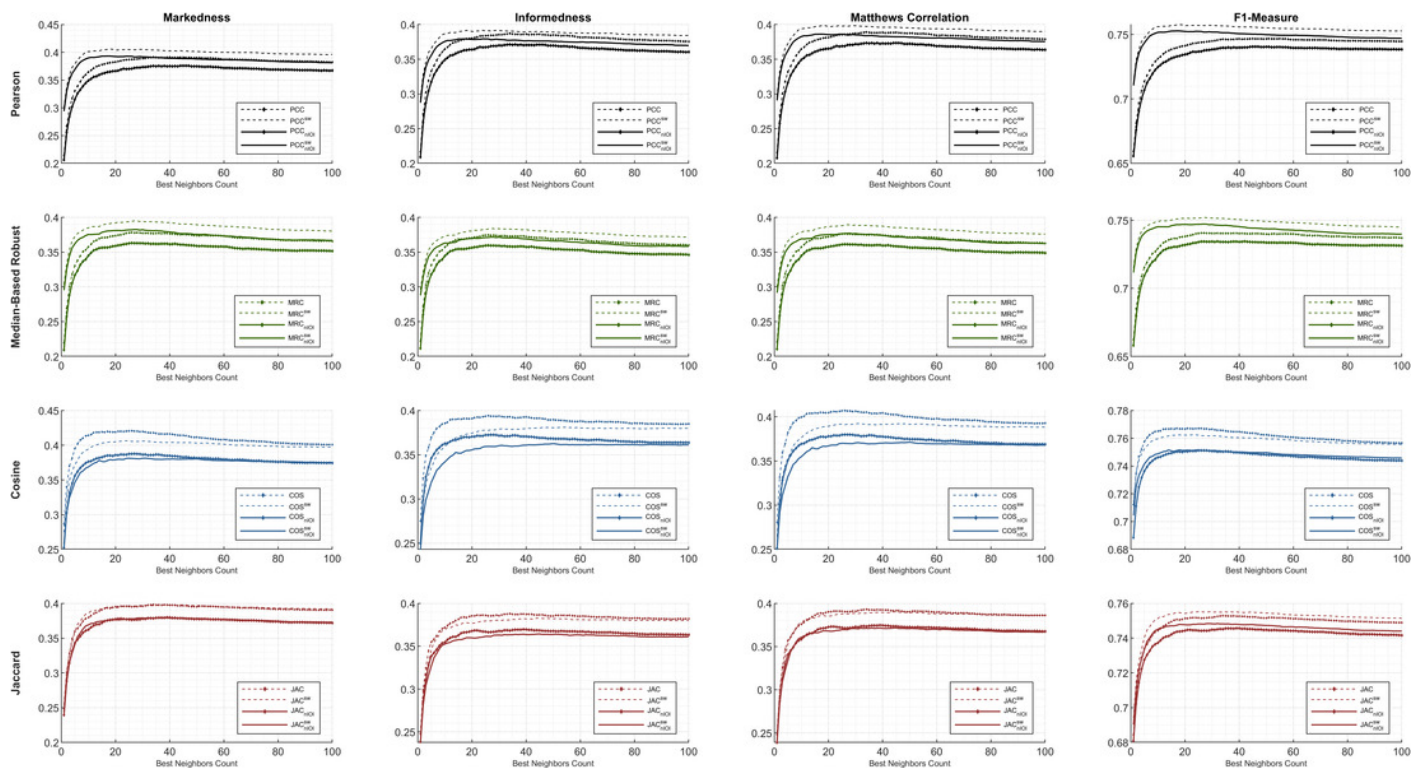



Figure 2

The evaluation over ML1M: similarity weight and preeminent metric combination to compare the dynamicity and weight significance.

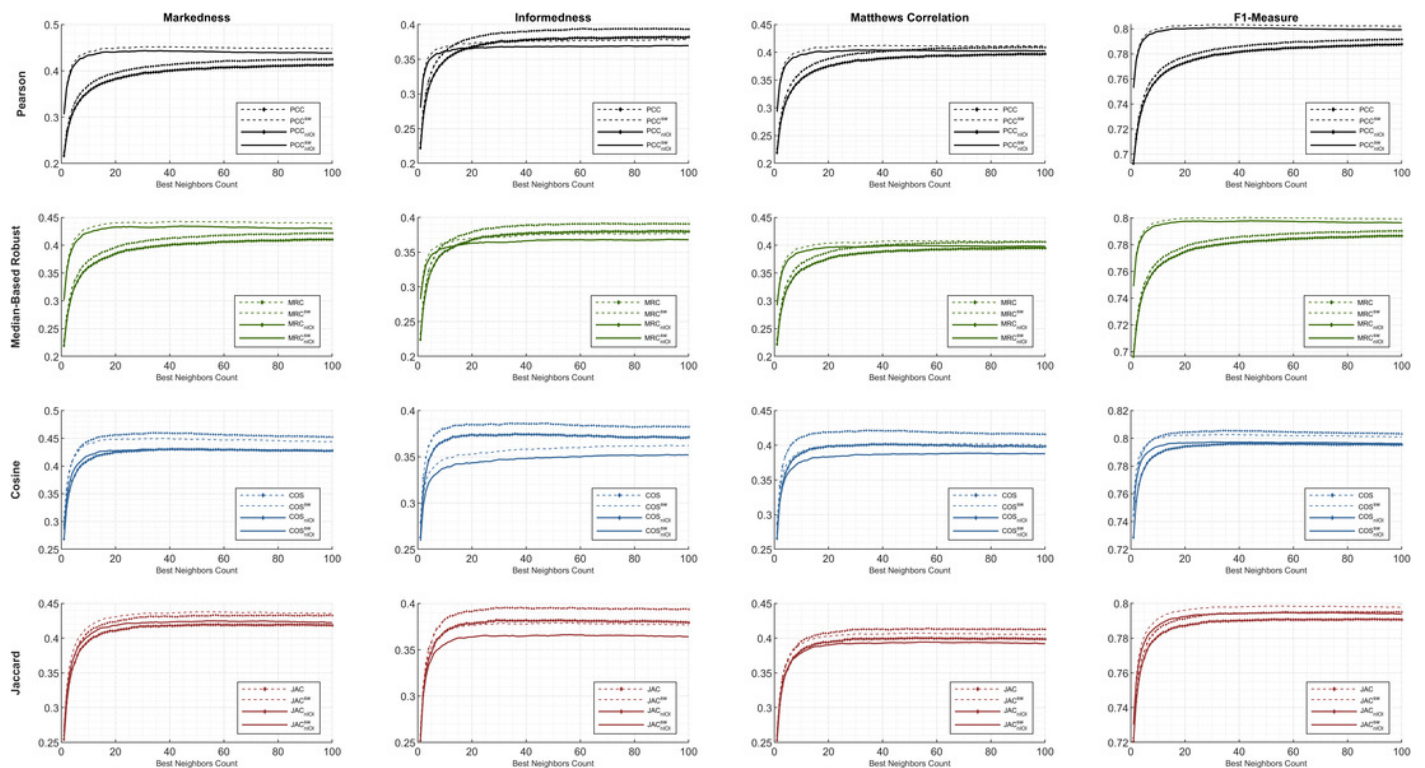



Figure 3

Markedness, informedness, and Matthews correlation as preeminent metrics, plus F1-measure highlighting the hybrid monitoring for (a) ML100K and (b) ML1M.

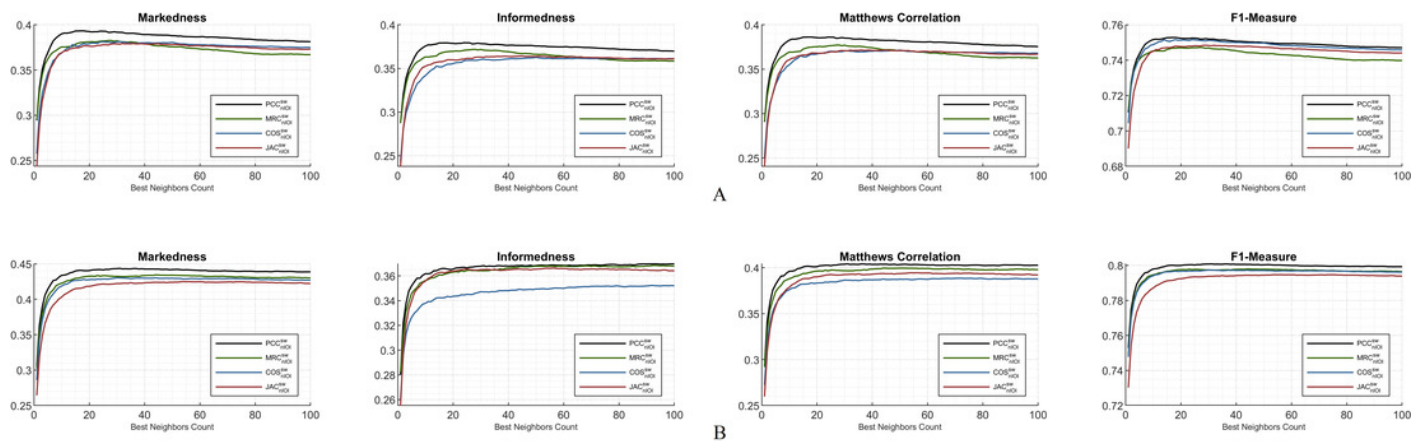



Figure 4

Extended performance evaluation on ML100K considering the hybrid monitoring.

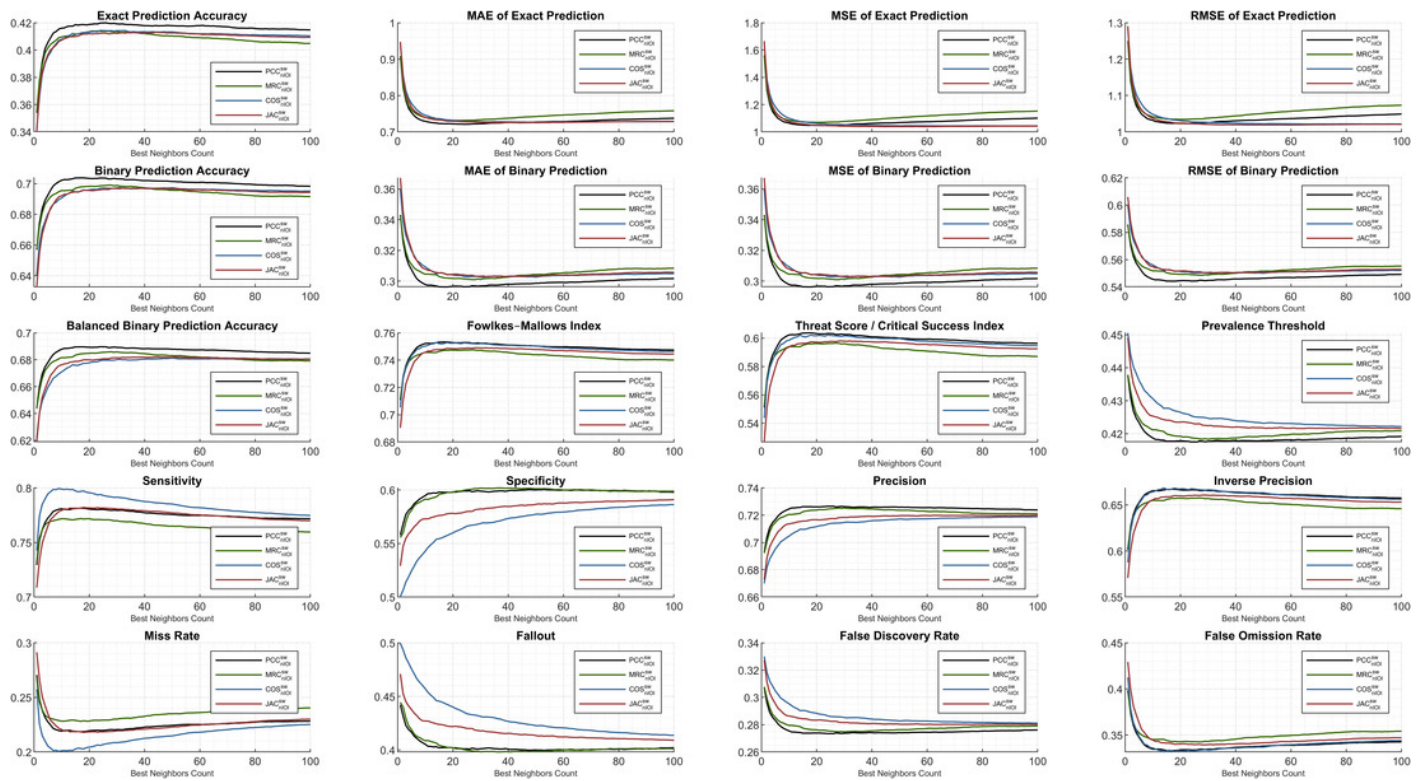



Figure 5

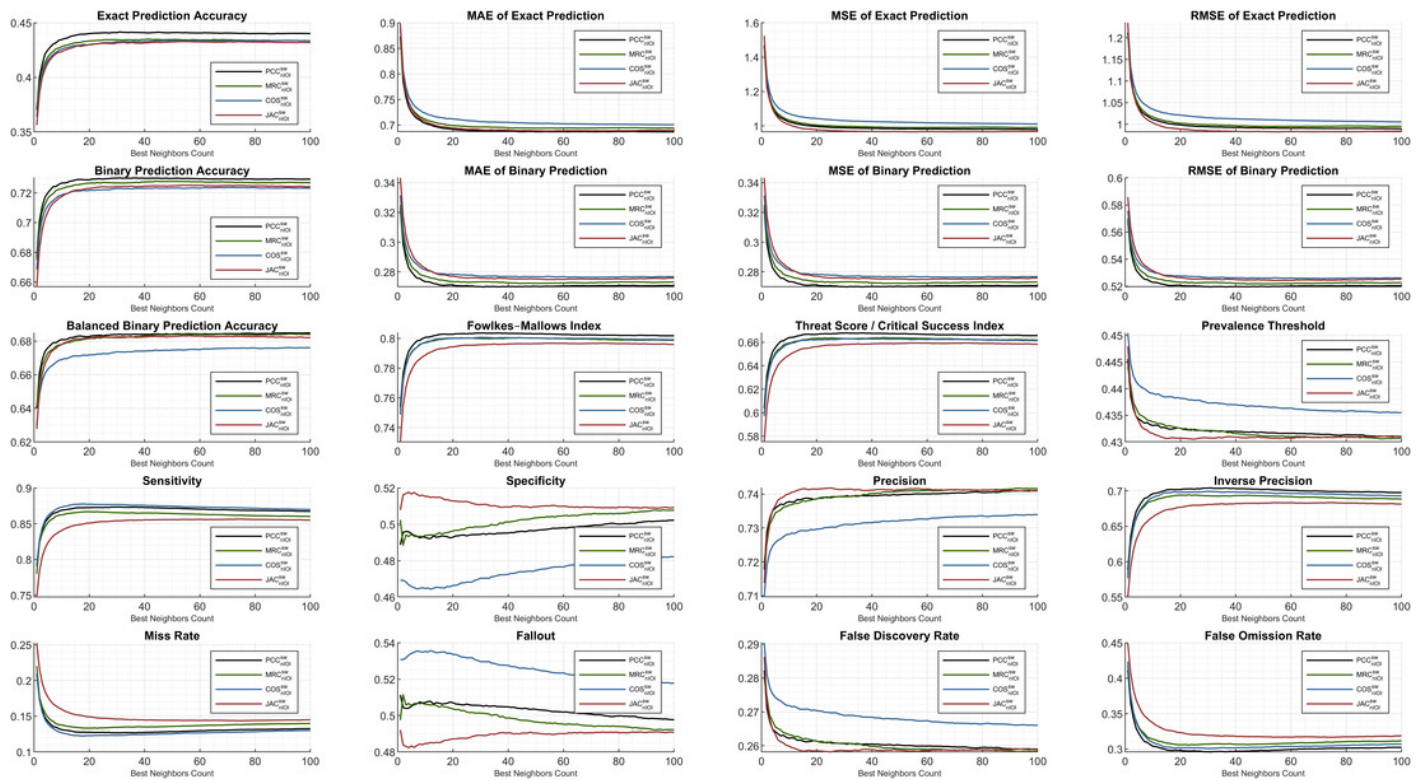
Extended performance evaluation on ML1M considering the hybrid monitoring.

Symbols and abbreviations list.

TABLE I
SYMBOLS AND ABBREVIATIONS LIST

Symbol / Abbraviation	Explanation
a	User-of-interest
$\hat{a}$	User-of-interest where test item bias is discarded
i	Any item-of-interest
U	Possible nominees to be a neighbor
u	Any possible neighbor for collaboration
u^*	Sorted and selected neighbor
$\boldsymbol{r_u}$	Rating vector of u for all items
$r_{u,i}$	Rating of u for i
$\bar{r}_u$	The mean of given ratings for u
$\tilde{r}_u$	The median of given ratings for u
I_u	The rating history of u
PCC	**P**earson **C**orrelation **C**oefficient
MRC	**M**edian-Based **R**obust **C**orrelation Coefficient
COS	**Cos**ine Similarity
JAC	**Jac**card Similarity
$w_{a,u}^S$	Similarity weight between a and u for equation S, where S can be given similarity equations
$p_{a,i}$	The rating prediction of a for i
CIC	**C**o-rated **I**tem **C**ount between two users
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative

MovieLens (ML) release comparison [9], [27].

TABLE II
MovieLens (ML) Release Comparison [9], [27]

Releases	ML100K	ML1M	ML10M	ML20M	ML25M
Number of Ratings	100,000	1,000,209	10,000,054	20,000,263	25,000,095
Number of Users	943	6,040	69,868	138,493	162,541
Number of Movies	1,682	3,706	10,681	27,278	62,423
Timespan	09/1997 – 04/1998	04/2000 – 02/2003	01/1995 – 01/2009	01/1995 – 03/2015	01/1995 – 11/2019
Miscellaneous Information	At least 20 ratings by each user, simple demographic information for users (age, gender, occupation, zip-code). 5-star rating.	At least 20 ratings by each user, simple demographic information for users (age, gender, occupation, zip-code). 5-star rating.	At least 20 ratings by each user. No demographic information. Each user is represented by only an ID. 5-star rating with half-stars.	At least 20 ratings by each user. No demographic information. Each user is represented by only an ID. 5-star rating with half-stars.	At least 20 ratings by each user. No demographic information. Each user is represented by only an ID. 5-star rating with half-stars.

Well-known performance metrics.

TABLE III
WELL-KNOWN PERFORMANCE METRICS

Metric Name	Formula
Exact Accuracy	$\dfrac{Exact\ Prediction\ Count}{TP\ +\ TN\ +\ FP\ +\ FN}$
Threshold Accuracy	$\dfrac{TP\ +\ TN}{TP\ +\ TN\ +\ FP\ +\ FN}$
Sensitivity / Recall / True Positive Rate	$\dfrac{TP}{TP\ +\ FN}$
Precision / Positive Pred. Value	$\dfrac{TP}{TP\ +\ FP}$
F1-Measure	$\dfrac{2\ \times\ TP}{2\ \times\ TP\ +\ FP\ +\ FN}$
Specificity / Inverse Sensitivity / True Negative Rate	$\dfrac{TN}{FP\ +\ TN}$
Inverse Precision / Negative Pred. Value	$\dfrac{TN}{TN\ +\ FN}$
False Discovery Rate	$1 - Precision$
False Omission Rate	$1 - Inverse\ Precision$
Fallout / False Positive Rate	$1 - Specificity$
Miss Rate / False Negative Rate	$1 - Sensitivity$
Fowlkes–Mallows Index	$\sqrt{Precision\ \times\ Sensitivity}$
Balanced Accuracy	$\dfrac{(Sensitivity + Specificity)}{2}$
Threat Score / Critical Success Index	$\dfrac{TP}{TP\ +\ FN + FP}$
Prevalence Threshold	$\dfrac{\sqrt{Sensitivity \times (1 - Specificity)} + Specificity - 1}{Sensitivity + Specificity - 1}$

Preeminent performance metrics.

TABLE IV
PREEMINENT PERFORMANCE METRICS

Metric Name	Formula
Markedness	$\dfrac{TP}{TP+FP}+\dfrac{TN}{TN+FN}-1$
Informedness	$\dfrac{TP}{TP+FN}+\dfrac{TN}{TN+FP}-1$
Matthews Correlation	$\dfrac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)\times(TP+FN)\times(TN+FP)\times(TN+FN)}}$

All test configurations considering *nIOI* and *SW* similarity measurements.

TABLE V
ALL TEST CONFIGURATIONS CONSIDERING $nIOI$ AND SW SIMILARITY MEASUREMENTS

Abbreviation of Similarity Equation	Dynamic	Significance Weighting	Related Equation	BNC $LNC : \varepsilon : MNC$
PCC			Eq. (1)	
PCC^{sw}		✓	Eq. (7)	
PCC_{nIOI}	✓		Eq. (6)	
PCC_{nIOI}^{sw}	✓	✓	Eq. (14)	
MRC			Eq. (2)	
MRC^{sw}		✓	Eq. (9)	
MRC_{nIOI}	✓		Eq. (8)	
MRC_{nIOI}^{sw}	✓	✓	Eq. (14)	
COS			Eq. (3)	1:1:100
COS^{sw}		✓	Eq. (11)	
COS_{nIOI}	✓		Eq. (10)	
COS_{nIOI}^{sw}	✓	✓	Eq. (14)	
JAC			Eq. (4)	
JAC^{sw}		✓	Eq. (13)	
JAC_{nIOI}	✓		Eq. (12)	
JAC_{nIOI}^{sw}	✓	✓	Eq. (14)	

The best-performing *BNC* values under a variety of performance metrics: ML100K.

TABLE VI
THE BEST-PERFORMING BNC VALUES UNDER A VARIETY OF PERFORMANCE METRICS: ML100K

Equation	Performance Metrics																							
	Markedness	Informedness	Matthews Correlation	F1-Measure	MAE Exact	MSE Exact	RMSE Exact	MAE Threshold	MSE Threshold	RMSE Threshold	Accuracy Exact	Accuracy Threshold	Accuracy Balanced	Fowlkes-Mallows Index	Prevalence Threshold	Threat Score	Precision	Inverse Precision	Sensitivity/Recall	Specificity	Fallout	Miss Rate	False Discovery Rate	False Omission Rate
PCC_{nIOI}	45	34	45	47	34	23	23	45	45	45	34	45	34	47	34	47	34	47	100	17	17	100	34	47
PCC_{nIOI}^{sw}	17	24	17	17	24	22	22	17	17	17	26	17	24	17	26	17	26	17	17	53	53	17	26	17
MRC_{nIOI}	26	26	26	42	27	19	19	26	26	26	27	26	26	42	25	42	25	42	59	12	12	59	25	42
MRC_{nIOI}^{sw}	27	28	28	27	23	20	20	27	27	27	23	27	28	27	30	27	30	27	17	37	37	17	30	27
COS_{nIOI}	27	27	27	25	28	35	35	27	27	27	27	27	27	25	36	25	36	25	25	100	100	25	36	25
COS_{nIOI}^{sw}	31	50	50	18	50	67	67	31	31	31	24	31	50	14	99	18	99	14	9	100	100	9	99	14
JAC_{nIOI}	39	39	39	39	40	40	40	39	39	39	36	39	39	39	39	39	39	39	39	35	20	20	35	39
JAC_{nIOI}^{sw}	32	45	45	29	45	59	59	36	36	36	44	36	45	29	100	29	100	29	18	100	100	18	100	29

The best-performing *BNC* values under a variety of performance metrics: ML1M.

TABLE VII

THE BEST-PERFORMING BNC VALUES UNDER A VARIETY OF PERFORMANCE METRICS: ML1M

Equation	Performance Metrics																							
	Markedness	Informedness	Matthews Correlation	F1-Measure	MAE Exact	MSE Exact	RMSE Exact	MAE Threshold	MSE Threshold	RMSE Threshold	Accuracy Exact	Accuracy Threshold	Accuracy Balanced	Fowlkes-Mallows Index	Prevalence Threshold	Threat Score	Precision	Inverse Precision	Sensitivity/Recall	Specificity	Fallout	Miss Rate	False Discovery Rate	False Omission Rate
PCC_{nIOI}	100	91	91	100	99	91	91	100	100	100	96	100	91	100	31	100	31	100	100	4	4	100	31	100
PCC_{nIOI}^{sw}	31	93	31	31	100	100	100	31	31	31	31	31	93	31	100	31	100	31	31	100	100	31	100	31
MRC_{nIOI}	97	92	92	97	97	60	60	92	92	92	92	92	92	97	48	97	48	97	100	8	8	100	48	97
MRC_{nIOI}^{sw}	44	93	44	44	58	94	94	44	44	44	41	44	93	44	93	44	93	23	23	96	96	23	93	23
COS_{nIOI}	41	36	41	41	45	45	45	41	41	41	72	41	36	50	13	41	13	50	83	4	4	83	13	50
COS_{nIOI}^{sw}	30	92	72	30	97	100	100	72	72	72	91	72	92	30	99	30	99	30	18	99	99	18	99	30
JAC_{nIOI}	58	29	58	84	58	53	53	58	58	58	58	58	29	84	29	84	29	90	99	10	10	99	29	90
JAC_{nIOI}^{sw}	54	56	56	54	55	55	55	56	56	56	54	56	56	75	25	54	25	75	75	4	4	75	25	75

Adequate weight-metric combination of the top-performing *BNC*s: ML100K.

TABLE VIII
ADEQUATE WEIGHT-METRIC COMBINATION OF THE TOP-PERFORMING BNCS: ML100K

Equation	BNC	Markedness	Informedness	Matthews Correlation	F1-Measure	MAE Exact	MSE Exact	RMSE Exact	MAE Threshold	MSE Threshold	RMSE Threshold	Accuracy Exact	Accuracy Threshold	Accuracy Balanced	Fowlkes-Mallows Index	Prevalence Threshold	Threat Score	Precision	Inverse Precision	Sensitivity/Recall	Specificity	Fallout	Miss Rate	False Discovery Rate	False Omission Rate
PCC_{nlOI}	17	.364	.363	.364	.733	.744	1.081	1.040	.310	.310	.557	.403	.690	.681	.733	.416	.579	.729	.635	.737	.626	.374	.263	.271	.365
	23	.371	.368	.369	.737	.740	1.075	1.037	.307	.307	.554	.405	.693	.684	.737	.416	.583	.730	.641	.744	.624	.376	.256	.270	.359
	34	.375	.371	.373	.740	.739	1.078	1.038	.304	.304	.552	.407	.696	.686	.740	.415	.587	.730	.645	.750	.622	.378	.250	.270	.355
	45	.376	.371	.374	.741	.742	1.091	1.044	.304	.304	.551	.406	.696	.686	.741	.416	.588	.729	.647	.752	.619	.381	.248	.271	.353
	47	.376	.371	.373	.741	.742	1.092	1.045	.304	.304	.552	.406	.696	.685	.741	.416	.588	.729	.647	.753	.618	.382	.247	.271	.353
	100	.367	.360	.364	.739	.756	1.141	1.068	.308	.308	.555	.403	.692	.680	.739	.419	.586	.723	.644	.754	.606	.394	.246	.277	.356
PCC_{nlOI}^{sw}	17	.393	.379	.386	.753	.722	1.048	1.023	.296	.296	.544	.419	.704	.690	.753	.418	.604	.726	.667	.781	.598	.402	.219	.274	.333
	22	.392	.379	.385	.752	.721	1.046	1.023	.297	.297	.545	.419	.703	.689	.753	.418	.603	.726	.666	.780	.599	.402	.220	.274	.334
	24	.393	.379	.386	.753	.721	1.047	1.023	.296	.296	.544	.420	.704	.690	.753	.417	.603	.727	.666	.780	.599	.401	.220	.273	.334
	26	.393	.379	.386	.752	.721	1.049	1.024	.296	.296	.544	.420	.704	.690	.753	.417	.603	.727	.666	.780	.599	.401	.220	.273	.334
	53	.388	.376	.382	.750	.728	1.070	1.034	.299	.299	.547	.418	.701	.688	.750	.418	.600	.726	.662	.775	.600	.400	.225	.274	.338
MRC_{nlOI}	12	.353	.351	.352	.728	.755	1.105	1.051	.316	.316	.562	.397	.684	.676	.728	.419	.572	.724	.628	.732	.620	.380	.268	.276	.372
	19	.358	.356	.357	.732	.750	1.096	1.047	.313	.313	.559	.400	.687	.678	.732	.419	.577	.725	.634	.738	.617	.383	.262	.275	.366
	25	.363	.360	.361	.734	.748	1.098	1.048	.311	.311	.557	.402	.689	.680	.734	.418	.580	.726	.637	.743	.617	.383	.257	.274	.363
	26	.363	.360	.361	.734	.748	1.099	1.048	.310	.310	.557	.402	.690	.680	.734	.418	.580	.726	.637	.743	.617	.383	.257	.274	.363
	27	.363	.360	.361	.734	.748	1.099	1.048	.310	.310	.557	.402	.690	.680	.734	.418	.580	.726	.638	.744	.616	.384	.256	.274	.362
	42	.362	.358	.360	.735	.754	1.121	1.058	.311	.311	.557	.401	.689	.679	.735	.419	.581	.724	.638	.745	.613	.387	.255	.276	.362
	59	.358	.353	.356	.734	.761	1.147	1.071	.313	.313	.559	.400	.687	.677	.734	.420	.579	.722	.636	.746	.607	.393	.254	.278	.364
MRC_{nlOI}^{sw}	17	.380	.369	.375	.747	.733	1.071	1.035	.302	.302	.550	.412	.698	.684	.747	.419	.596	.723	.657	.772	.597	.403	.228	.277	.343
	20	.381	.370	.375	.747	.732	1.069	1.034	.302	.302	.549	.413	.698	.685	.747	.419	.596	.724	.657	.772	.598	.402	.228	.276	.343
	23	.381	.370	.376	.747	.731	1.070	1.035	.302	.302	.549	.414	.698	.685	.747	.419	.596	.724	.657	.771	.599	.401	.229	.276	.343
	27	.383	.372	.377	.747	.732	1.072	1.035	.301	.301	.549	.414	.699	.686	.748	.418	.597	.725	.658	.771	.601	.399	.229	.275	.342
	28	.383	.372	.377	.747	.732	1.073	1.036	.301	.301	.549	.414	.699	.686	.747	.418	.596	.725	.657	.770	.601	.399	.230	.275	.343
	30	.382	.372	.377	.747	.733	1.077	1.038	.301	.301	.549	.414	.699	.686	.747	.418	.596	.725	.657	.770	.602	.398	.230	.275	.343
	37	.380	.370	.375	.746	.736	1.085	1.042	.302	.302	.550	.413	.698	.685	.746	.419	.595	.725	.656	.768	.602	.398	.232	.275	.344
COS_{nlOI}	25	.388	.372	.380	.751	.720	1.027	1.013	.299	.299	.547	.416	.701	.686	.752	.420	.602	.723	.665	.782	.590	.410	.218	.277	.335
	27	.388	.373	.380	.751	.719	1.025	1.012	.299	.299	.547	.416	.701	.686	.752	.420	.602	.723	.665	.782	.591	.409	.218	.277	.335
	28	.388	.372	.380	.751	.719	1.025	1.012	.299	.299	.547	.416	.701	.686	.752	.420	.602	.723	.665	.782	.591	.409	.218	.277	.335
	35	.387	.372	.379	.751	.720	1.024	1.012	.299	.299	.547	.415	.701	.686	.751	.420	.601	.723	.664	.781	.591	.409	.219	.277	.336
	36	.386	.372	.379	.750	.720	1.025	1.012	.299	.299	.547	.415	.701	.686	.751	.420	.601	.723	.663	.780	.592	.408	.220	.277	.337
	100	.374	.364	.369	.744	.727	1.035	1.017	.305	.305	.552	.409	.695	.682	.744	.420	.592	.722	.652	.767	.597	.403	.233	.278	.348
COS_{nlOI}^{sw}	9	.367	.339	.353	.748	.749	1.108	1.053	.310	.310	.557	.407	.690	.670	.750	.431	.598	.703	.663	.799	.540	.460	.201	.297	.337
	14	.378	.353	.365	.752	.736	1.076	1.037	.305	.305	.552	.412	.695	.676	.753	.428	.602	.710	.668	.799	.554	.446	.201	.290	.332
	18	.379	.355	.367	.752	.733	1.066	1.032	.304	.304	.551	.413	.696	.677	.753	.427	.602	.711	.668	.797	.558	.442	.203	.289	.332
	24	.381	.359	.370	.752	.729	1.056	1.028	.303	.303	.550	.414	.697	.679	.753	.425	.602	.714	.668	.794	.565	.435	.206	.286	.332
	31	.381	.361	.371	.751	.728	1.052	1.026	.303	.303	.550	.414	.697	.680	.752	.425	.602	.715	.666	.791	.569	.431	.209	.285	.334
	50	.381	.363	.371	.750	.726	1.045	1.022	.303	.303	.550	.413	.697	.681	.751	.423	.600	.717	.663	.785	.577	.423	.215	.283	.337
	67	.377	.361	.369	.748	.727	1.043	1.021	.304	.304	.551	.412	.696	.681	.748	.423	.597	.718	.659	.780	.581	.419	.220	.282	.341
	99	.375	.361	.368	.746	.728	1.044	1.022	.305	.305	.552	.410	.695	.681	.746	.422	.595	.719	.656	.775	.586	.414	.225	.281	.344
	100	.375	.361	.368	.746	.728	1.044	1.022	.305	.305	.552	.410	.695	.681	.746	.422	.595	.719	.656	.775	.586	.414	.225	.281	.344
JAC_{nlOI}	20	.377	.368	.373	.744	.727	1.034	1.017	.303	.303	.551	.409	.697	.684	.745	.419	.593	.724	.653	.766	.602	.398	.234	.276	.347
	35	.379	.369	.374	.746	.724	1.026	1.013	.303	.303	.550	.410	.697	.684	.746	.419	.594	.724	.655	.768	.601	.399	.232	.276	.345
	36	.380	.369	.374	.746	.724	1.026	1.013	.302	.302	.550	.411	.698	.685	.746	.419	.594	.724	.655	.768	.601	.399	.232	.276	.345
	39	.380	.370	.375	.746	.723	1.025	1.012	.302	.302	.550	.410	.698	.685	.746	.419	.595	.725	.655	.768	.602	.398	.232	.275	.345
	40	.380	.370	.375	.746	.723	1.025	1.012	.302	.302	.550	.410	.698	.685	.746	.419	.595	.725	.655	.768	.601	.399	.232	.275	.345
JAC_{nlOI}^{sw}	18	.377	.360	.368	.748	.729	1.050	1.025	.304	.304	.552	.412	.696	.680	.749	.424	.597	.717	.660	.782	.578	.422	.218	.283	.340
	29	.379	.363	.371	.748	.726	1.042	1.021	.303	.303	.551	.412	.697	.681	.749	.422	.598	.718	.661	.781	.582	.418	.219	.282	.339
	32	.379	.364	.371	.748	.726	1.041	1.020	.303	.303	.550	.413	.697	.682	.749	.422	.598	.719	.661	.781	.583	.417	.219	.281	.339
	36	.379	.364	.372	.748	.725	1.039	1.020	.303	.303	.550	.413	.697	.682	.749	.422	.598	.719	.660	.780	.584	.416	.220	.281	.340
	44	.379	.364	.372	.748	.725	1.038	1.019	.303	.303	.550	.413	.697	.682	.748	.422	.597	.720	.660	.779	.586	.414	.221	.280	.340
	45	.379	.364	.372	.748	.725	1.038	1.019	.303	.303	.550	.413	.697	.682	.748	.422	.597	.720	.659	.778	.586	.414	.222	.280	.341
	59	.377	.363	.370	.747	.726	1.038	1.019	.304	.304	.551	.412	.696	.682	.747	.422	.596	.720	.657	.775	.588	.412	.225	.280	.343
	100	.373	.361	.367	.744	.729	1.042	1.021	.306	.306	.553	.409	.694	.681	.745	.422	.593	.720	.653	.770	.591	.409	.230	.280	.347

The fractional values in the table are displayed based on three significant digits. The heat-map coloring is achieved according to full precision.

Adequate weight-metric combination of the top-performing *BNC*s: ML1M.

TABLE IX
ADEQUATE WEIGHT-METRIC COMBINATION OF THE TOP-PERFORMING BNCs: ML1M

Equation	BNC	Markedness	Informedness	Matthews Correlation	F1-Measure	MAE Exact	MSE Exact	RMSE Exact	MAE Threshold	MSE Threshold	RMSE Threshold	Accuracy Exact	Accuracy Threshold	Accuracy Balanced	Fowlkes-Mallows Index	Prevalence Threshold	Threat Score	Precision	Inverse Precision	Sensitivity/Recall	Specificity	Fallout	Miss Rate	False Discovery Rate	False Omission Rate
PCC_{nlOI}	4	.305	.308	.307	.735	.797	1.200	1.096	.328	.328	.572	.378	.672	.654	.735	.432	.581	.740	.566	.730	.578	.422	.270	.260	.434
	31	.396	.376	.386	.779	.707	.981	.991	.283	.283	.532	.415	.717	.688	.780	.422	.639	.755	.641	.805	.571	.429	.195	.245	.359
	91	.413	.382	.397	.787	.696	.966	.983	.277	.277	.526	.422	.723	.691	.788	.423	.649	.754	.659	.824	.559	.441	.176	.246	.341
	96	.412	.382	.397	.787	.696	.966	.983	.277	.277	.526	.423	.723	.691	.788	.423	.649	.754	.659	.824	.557	.443	.176	.246	.341
	99	.413	.382	.397	.787	.696	.966	.983	.277	.277	.526	.422	.723	.691	.788	.423	.649	.754	.659	.825	.557	.443	.175	.246	.341
	100	.413	.382	.397	.788	.696	.966	.983	.277	.277	.526	.423	.723	.691	.788	.423	.650	.754	.659	.825	.557	.443	.175	.246	.341
PCC_{nlOI}^{sw}	31	.444	.368	.404	.801	.688	.988	.994	.270	.270	.520	.442	.730	.684	.804	.432	.668	.740	.704	.873	.495	.505	.127	.260	.296
	93	.439	.370	.403	.799	.687	.981	.990	.271	.271	.520	.440	.729	.685	.802	.431	.666	.741	.698	.868	.502	.498	.132	.259	.302
	100	.439	.370	.403	.799	.687	.979	.990	.271	.271	.520	.440	.729	.685	.802	.431	.666	.741	.698	.867	.502	.498	.133	.259	.302
MRC_{nlOI}	8	.351	.344	.347	.759	.747	1.075	1.037	.304	.304	.552	.398	.696	.672	.759	.426	.611	.748	.603	.769	.575	.425	.231	.252	.397
	48	.403	.379	.391	.783	.704	.980	.990	.280	.280	.530	.417	.720	.689	.784	.422	.643	.754	.649	.814	.564	.436	.186	.246	.351
	60	.406	.379	.393	.785	.702	.978	.989	.279	.279	.528	.419	.721	.690	.785	.423	.646	.754	.652	.818	.562	.438	.182	.246	.348
	92	.411	.380	.395	.787	.701	.980	.990	.278	.278	.527	.421	.722	.690	.787	.423	.648	.753	.657	.823	.557	.443	.177	.247	.343
	97	.411	.380	.395	.787	.701	.979	.990	.278	.278	.527	.420	.722	.690	.787	.423	.648	.753	.657	.823	.557	.443	.177	.247	.343
	100	.410	.380	.395	.787	.701	.980	.990	.278	.278	.527	.420	.722	.690	.787	.423	.648	.753	.657	.823	.556	.444	.177	.247	.343
MRC_{nlOI}^{sw}	23	.434	.364	.397	.798	.698	1.001	1.001	.273	.273	.523	.434	.727	.682	.800	.432	.664	.739	.695	.867	.497	.503	.133	.261	.305
	41	.434	.366	.399	.798	.694	.992	.996	.273	.273	.522	.435	.727	.683	.800	.432	.664	.740	.693	.865	.501	.499	.135	.260	.307
	44	.434	.367	.400	.798	.694	.992	.996	.272	.272	.522	.435	.728	.684	.800	.431	.664	.741	.694	.865	.502	.498	.135	.259	.306
	58	.433	.368	.399	.798	.694	.990	.995	.272	.272	.522	.435	.728	.684	.800	.431	.663	.741	.692	.863	.505	.495	.137	.259	.308
	93	.431	.368	.398	.797	.694	.989	.995	.273	.273	.522	.434	.727	.684	.799	.431	.662	.742	.689	.861	.508	.492	.139	.258	.311
	94	.431	.368	.398	.797	.694	.989	.995	.273	.273	.522	.434	.727	.684	.799	.431	.662	.742	.689	.861	.508	.492	.139	.258	.311
	96	.431	.368	.398	.797	.694	.990	.995	.273	.273	.522	.434	.727	.684	.799	.431	.662	.742	.689	.860	.508	.492	.140	.258	.311
COS_{nlOI}	4	.373	.346	.359	.774	.736	1.070	1.034	.294	.294	.542	.410	.706	.673	.775	.431	.631	.742	.631	.809	.537	.463	.191	.258	.369
	13	.418	.372	.394	.792	.692	.969	.984	.276	.276	.525	.430	.724	.686	.793	.428	.655	.746	.673	.843	.528	.472	.157	.254	.327
	36	.430	.375	.402	.796	.684	.951	.975	.272	.272	.522	.434	.728	.687	.798	.428	.661	.745	.685	.854	.520	.480	.146	.255	.315
	41	.431	.374	.402	.796	.683	.950	.975	.272	.272	.522	.434	.728	.687	.798	.428	.661	.745	.686	.855	.520	.480	.145	.255	.314
	45	.430	.374	.401	.796	.683	.949	.974	.272	.272	.522	.434	.728	.687	.798	.429	.661	.745	.685	.855	.519	.481	.145	.255	.315
	50	.430	.374	.401	.796	.683	.951	.975	.272	.272	.522	.434	.728	.687	.798	.429	.661	.745	.686	.855	.518	.482	.145	.255	.314
	72	.429	.372	.399	.796	.684	.954	.977	.273	.273	.522	.435	.727	.686	.798	.429	.661	.744	.685	.855	.517	.483	.145	.256	.315
	83	.429	.371	.399	.796	.684	.954	.977	.273	.273	.523	.434	.727	.686	.798	.429	.661	.744	.685	.855	.516	.484	.145	.256	.315
COS_{nlOI}^{sw}	18	.428	.343	.383	.797	.712	1.044	1.022	.278	.278	.527	.430	.722	.672	.800	.438	.662	.730	.699	.878	.466	.534	.122	.270	.301
	30	.430	.347	.387	.797	.706	1.030	1.015	.277	.277	.526	.433	.723	.674	.801	.437	.663	.731	.699	.877	.470	.530	.123	.269	.301
	72	.429	.352	.389	.797	.701	1.015	1.007	.276	.276	.526	.434	.724	.676	.800	.436	.662	.733	.696	.873	.479	.521	.127	.267	.304
	91	.428	.352	.388	.796	.701	1.012	1.006	.277	.277	.526	.434	.723	.676	.799	.436	.662	.734	.694	.871	.481	.519	.129	.266	.306
	92	.428	.352	.388	.796	.701	1.012	1.006	.277	.277	.526	.434	.723	.676	.799	.436	.662	.734	.694	.871	.482	.518	.129	.266	.306
	97	.427	.352	.388	.796	.701	1.011	1.006	.277	.277	.526	.434	.723	.676	.799	.436	.661	.734	.693	.870	.482	.518	.130	.266	.307
	99	.427	.352	.388	.796	.701	1.011	1.005	.277	.277	.526	.433	.723	.676	.799	.435	.661	.734	.693	.870	.482	.518	.130	.266	.307
	100	.427	.352	.388	.796	.701	1.011	1.005	.277	.277	.526	.433	.723	.676	.799	.436	.661	.734	.693	.870	.482	.518	.130	.266	.307
JAC_{nlOI}	10	.396	.369	.382	.781	.705	.983	.991	.284	.284	.533	.419	.716	.685	.782	.425	.641	.750	.646	.814	.555	.445	.186	.250	.354
	29	.417	.382	.399	.790	.689	.947	.973	.275	.275	.524	.425	.725	.691	.791	.424	.652	.752	.665	.831	.551	.449	.169	.248	.335
	53	.420	.382	.400	.791	.687	.942	.971	.274	.274	.524	.426	.726	.691	.792	.424	.654	.752	.668	.834	.548	.452	.166	.248	.332
	58	.420	.382	.400	.791	.687	.943	.971	.274	.274	.524	.426	.726	.691	.792	.424	.654	.752	.668	.834	.547	.453	.166	.248	.332
	84	.420	.381	.400	.791	.688	.944	.972	.274	.274	.524	.426	.726	.690	.792	.424	.654	.751	.668	.835	.546	.454	.165	.249	.332
	90	.419	.380	.399	.791	.688	.945	.972	.275	.275	.524	.426	.725	.690	.792	.425	.654	.751	.668	.835	.545	.455	.165	.249	.332
	99	.419	.380	.399	.791	.688	.945	.972	.275	.275	.524	.426	.725	.690	.792	.425	.654	.751	.668	.836	.544	.456	.164	.249	.332
JAC_{nlOI}^{sw}	4	.366	.333	.349	.773	.745	1.098	1.048	.297	.297	.545	.407	.703	.667	.774	.435	.630	.735	.631	.815	.518	.482	.185	.265	.369
	25	.422	.366	.393	.794	.692	.972	.986	.276	.276	.525	.431	.724	.683	.795	.431	.658	.742	.680	.853	.513	.487	.147	.258	.320
	54	.425	.366	.395	.795	.688	.965	.982	.275	.275	.524	.433	.725	.683	.797	.431	.659	.742	.683	.856	.510	.490	.144	.258	.317
	55	.425	.366	.394	.795	.688	.965	.982	.275	.275	.524	.433	.725	.683	.797	.431	.659	.742	.683	.856	.510	.490	.144	.258	.317
	56	.425	.366	.395	.795	.688	.965	.982	.275	.275	.524	.433	.725	.683	.797	.431	.659	.742	.683	.856	.510	.490	.144	.258	.317
	75	.425	.366	.394	.795	.689	.966	.983	.275	.275	.524	.433	.725	.683	.797	.431	.659	.741	.683	.856	.509	.491	.144	.259	.317

The fractional values in the table are displayed based on three significant digits. The heat-map coloring is achieved according to full precision.

Box 1(on next page)

Algorithm 1: Pseudocode of the experimental process for an individual test package.

Algorithm 1: Pseudocode of the experimental process for an individual test package

A×I	The size of the dataset, A is for users (row count), I is for items (column count).
R=5	Randomly selected test items, $\binom{I_a}{5}$ for each user.
k=10	*10*-fold cross-validation.
LNC=1	*BNC* minimum value parameter.
MNC=100	*BNC* maximum value parameter.
ε=1	fine-tuned *BNC* increment parameter.
t=3.5	Binary prediction (*liked* or *disliked*) rating threshold (for 5-star scale).

1.   Create test *ItemSet* (A×R) randomly and set *k*-fold parameters
2.   **for** all users a =1:A associated with *k*-folds
  3.   **for** all items i =1:R in the corresponding row of *ItemSet*
    4.   **for each** *SimEq*
      5.   **for** all $bnc = LNC : \varepsilon : MNC$
        6. $BN \leftarrow$ getBestNeighbors(*SimEq, a,i,bnc*);
          for (a,i) pair, using the Train Set of corresponding folds
        7. $p_{a,i}^{SimEq,bnc} \leftarrow$ calculatePrediction(BN);
      endfor
    endfor
  endfor
endfor
for all a, i, *SimEq*, *bnc*
  8.   evaluatePerformance($p_{a,i}^{SimEq,bnc}, t$);
      exact and threshold perf. analysis for all $p_{a,i}^{SimEq,bnc}$
  endfor