Computer Science • 23(3) 2022 https://doi.org/10.7494/csci.2022.23.3.4227

Falguni Roy
Mahamudul Hasan

COMPARATIVE ANALYSIS OF DIFFERENT TRUST METRICS OF USER-USER TRUST-BASED RECOMMENDATION SYSTEM

Early bird

Abstract *Information overload is the biggest challenge nowadays for any website – especially e-commerce websites. However, this challenge has arisen due to the fast growth of information on the web (WWW) along with easier access to the internet. A collaborative filtering-based recommender system is the most useful application for solving the information overload problem by filtering relevant information for users according to their interests. However, the current system faces some significant limitations such as data sparsity, low accuracy, cold-start, and malicious attacks. To alleviate the above-mentioned issues, the relationship of trust incorporates in the system where it can be among users or items; such a system is known as a trust-based recommender system (TBRS). From the user perspective, the motive of a TBRS is to utilize the reliability among users to generate more-accurate and trusted recommendations. However, the study aims to present a comparative analysis of different trust metrics in the context of the type of trust definition of TBRS. Also, the study accomplishes 24 trust metrics in terms of the methodology, trust properties & measurements, validation approaches, and the experimented data set.*

Keywords trust-based recommender system, Pearson correlation coefficient, confidence, mean absolute error, precision, recall, coverage

Citation Computer Science 23(3) 2022: 337–375

Copyright © 2022 Author(s). This is an open access publication, which can be used, distributed and reproduced in any medium according to the Creative Commons CC-BY 4.0 License.

1. Introduction

For an extensive evaluation of internet accessibility, information sharing on the World Wide Web (WWW) is becoming an easy job for each user. Currently, most web applications allow millions of users to create, edit, and share information on the WWW in an unbound manner. As a result, system users face information-flooding issues where a user cannot attain the required information in a timely and structured manner for making the right decision. This issue is also known as the information overload problem [16]. To get rid of the issue, a recommender system (RS) is one of the smartest solutions. The primary objective of an RS is to provide useful personalized information for a user by recommending information from an information pool of the WWW [70]. Generally, a recommender system serves its objective in two steps. In the first step, the recommender system analyzes its users' historical data and user-entered data and then predicts a user's personalized data for subsequently recommending information in the last step based on the output of the first step. A recommender system was first inaugurated in 1992 in a project named Tapestry [24]. Initially, the recommender system was applied to e-commerce [8] and amusement-based websites like Amazon [66], Netflix [25], etc.; however, its application domain is not limited nowadays. Different applications of e-tourism [45], e-learning [8,18], e-government [2], and e-resource services [46] have implemented recommender systems to assist their users by receiving faster required information.

Usually, a recommender system (RS) recommends those items to a user that have not yet been experienced by the specific user; the process of recommendation started by deducing a relationship between users or entities [64]. However, a recommender system is broadly categorized into three types based on the information-filtering and recommendation strategy: collaborative filtering (CF), content-based filtering (CBF), and hybrid filtering (HF) [77]. The content-based filtering (CBF) approach needs two attributes and an algorithm for recommending items to a user. However, the attributes are the user's preferences profile and abundant descriptions of items, and the algorithm predicts the user's succeeding preferences to recommend a new set of items by deducing a matching between the attributes [61]. CBF could provide accurate recommendations in the case of a new user and item (known as cold-start), as it recommends items by matching user preferences and item descriptions; however, it is not lucrative for some limitations [51, 68]. CBF cannot provide accurate recommendations if inappropriate information exists in the item descriptions, and it also faces difficulties in retrieving multimedia information like color, texture, etc. [61, 68]. However, CBF also suffers from the overspecialization problem by recommending the same types of items continuously [8, 68]. Furthermore, CBF faces difficulties in measuring the correctness of a recommendation, as it does not contain such user feedback as item ratings [8].

Collaborative filtering-based RS (CF) needs continuous user participation in the system and an algorithm that examines the user item-rating matrix in order to identify similarly tested users or similar types of items for predicting a target user's choices and

Early Bird

then provides recommendations [61, 65]. The whole process of CF executes in three steps: data prepossessing, identifying any neighbors of a target, and recommending items [5]. A target could be any user or item. Generally, CF is categorized as memory- and/or model-based CF according to the way a neighborhood is selected [19, 61, 72, 77]. Model-based CF utilizes different machine-learning algorithms (for example, matrix factorization [31,56], the Bayesian method [52], clustering techniques [42], and genetic algorithms [1, 47]) to inspect a user item-rating matrix for offering new recommendations. On the contrary, memory-based CF statistically analyzes a user item-rating matrix to deduce any uniformity between items or users and offers recommendations based on the similarities [5, 61].

On the other side, hybrid filtering (HF) is the amalgamation of both the CBF and CF approaches in order to enhance both of the approaches' benefits by alleviating each approach's limitations [38, 40]. According to the operations, this is categorized into seven categories: switching, weighted, cascade, mixed, feature-combination, meta-level, and feature-augmented hybrid [38]. Usually, HF needs a vast amount of information for offering recommendations, as it is the integration of different approaches; its computational complexity is high and expensive as compared to others. In spite of the fact that CF suffers some significant flaws (such as data sparsity and cold-start [46]), it is the foremost proficient and widely utilized approach within RS so far [5, 19, 34, 54].

Usually, a CF-based recommender system faces a few problems that affect the system's performance. Usually, RS's performance is determined by the accuracy of users' taste prediction with the coverage of the maximum item of the system. The performance could be degraded due to the presence of data-sparsity [19, 28, 46, 61, 63] and cold-start [7, 19, 46, 61] problems. Data sparsity states a scenario when the number of ratings in a user item-rating matrix is not enough to identify a remarkable overlapping between those items that are rated by a pair of users; this causes the difficulties to create accurate predictions [19]. However, the cold-start issue is further categorized as either a cold-start user or cold-start item. The cold-start user issue emerges when a large number of new users exist or users have rated a low number of items in the system [19,38]. Also, the cold-start item defines the same problem in item prospective [19,61]. However, there has a proportional relationship between the cold-start and data-sparsity problem in the data. RS also suffers reliability issues, as users are generally unaware of the recommendation process and have no monitoring power over it. This creates a reliability issue and decreases user trust in the recommendations of the system.

A trust-based recommender system (TBRS) is one of the modern forms of RS. It includes a trust relationship in the system to ameliorate the system's accuracy and reliability in order to conquer existing issues such as data sparsity and cold-start [34, 72]. In the context of RS, trust usually determines one's faith in others' aptness of providing valuable ratings concerning the preference of the target [26]. However, TBRS also divides as either explicit trust or implicit trust in terms of the

Early bird

methodology of the trust information collection in the system [19,29,59]. Explicit trust in the system is determined by the users directly. Usually, explicit TBRS allows its users to take extra responsibility to assign other users as trusted users [67]. However, explicit trust is defined in the binary format for the privacy concern, which also limits a user from expressing the degree of trust to the trusted users. Conversely, implicit trust is defined in the system by using weighted similarity measures [54,61,78] or applying a probabilistic technique [59] in the user item-rating matrix. It also allows for the manifestation of the degree of trust between users. However, TBRS are further classified as memory-based and model-based approaches that are based on the methodology of the trust integration in RS [32].

In the last few years, several surveys have been done on the trust-based recommender system (TBRS); the surveys have focused on either the properties of trust or the process of the recommendation of TBRS. Also, most of the surveys have performed on the implicit trust [29, 33, 73]. For example, Guo et al. [29] reviewed six implicit-trust metrics according to the trust properties. On the other side, Yadav et al. [73] also surveyed implicit-trust metrics. However, Gupta et al. [33] presented a survey of eight implicit-trust metrics based on trust properties other than the trust establishment type, inferred trust, and network perspective. Selmi et al. [62] reviewed different existing TBRSs and classified them by trust type, relationship, value, propagation, aggregation, context, and techniques. On the other side, Jallouli et al. [39] surveyed the trust metrics in RS by trust propagation, user interactions, and rating vectors' perspectives. Although several TBRS surveys have been previously performed, no survey work has been comprehensively conducted to review TBRS according to trust properties, measurement, evaluation, and data set based on every category of trust (to the best of the authors' knowledge). The articles were selected for this survey by first considering the popularity of the trust metrics and then the publication database and recency.

Early Bird

1.1. Paper contribution

This paper systemically demonstrates a comprehensive review of several trust-based recommender system (TBRS) approaches. The contribution of the paper is four-fold and is described as follows:

- classified trust-based recommender system (TBRS) according to trust definition, subject of trust measurement, and methodology;
- summarized existing TBRS metrics and techniques;
- presented recent studies on TBRS that solve existing issues such as data sparsity, cold start, and error of prediction accuracy;
- provided comparative study in five aspects, such as methodology of trust determination, properties of trust, trust measurement, evaluation metrics, and data set on which experimentation is examined.

2. Trust-based recommender system

The trust-based recommender system is the next generation of collaborative filtering-based RS. Traditional collaborative filtering-based RS suffers from such issues as data sparsity, cold-start, profile injection attack, etc. Furthermore, collaborative filtering-based RS treats the similarity between a pair of users as symmetric. In real life, however, it is near impossible that two people may like an item in the same context. The trust-based recommender system applies the concept of trust in the traditional techniques to enhance a system's accuracy and reliability [53]. However, trust was initially used in the psychology and sociology disciplines, but it has currently become a valuable attribute in the computer science [23] and recommender system [26] fields as well. In the sociology discipline, trust is determined as a required belief and an oral commitment. However, trust is defined as "a commitment to believe in the smooth running of the future actions of another entity" in the computer science discipline [23]. In RS, trust is defined as one's conviction toward others in giving exact ratings that are relative to the inclinations of that user [26]. Usually, trust is used to scale users' similarities and express the integrity in the relationship between two users to a specific context. The value of the trust can be real or binary numbers, and the range is [-1, 1]. A trust value of "1" denotes the full faith of the target user on his trusted user, and "0" is defined as no trust. A negative trust value indicates the level of distrust that the target user has on other users.

2.1. Properties of trust

Usually, trust is a complex manner for humans; from a sociological perspective, it requires a belief in oral commitments. As a consequence, it is not an easy task to characterize and model the trust between users by using a mathematical equation or computationally. In RS, however, trust is specified based on some properties by utilizing a user's background, context, history of interaction, reputation, similarity, trust statement, etc. [22]. These properties indicate the existence of trust in a system and also define a way of measuring trust. As stated in trust theory, a trust relationship on the web should have the following distinct properties [9, 20, 29, 61]:

- **Asymmetry.** Trust is asymmetric. It is personal and varies with the different users with their own opinions. A user might have distinct faiths on a certain user according to his/her experiences. So, if user u trusts another user v, it is not obvious that user v trusts user u to the same extent.
- **Transitivity.** Calculated trust should be transitive; it is the most important property of trust and is also widely applied in TBRS. This is defined as follows: if user u trusts user v, and user v trusts another user w, then it can be concluded that user u could trust user w to some extent. In a real-life scenario, people tend to believe a companion of a companion more than a stranger. By supporting the transitive property, it is possible to establish an indirect-trust connection between users by identifying more trusted users to elevate the prediction performance of RS. The process of defining trusted users based on the indirect-trust connection

Early Proof

is called trust propagation [34, 57], and propagated trust is known as inferred trust.

- **Dynamicity.** By default, trust is dynamic; it is usually built continuously and changes as time goes on with more experiences. It can be expanded or diminished with positive or negative experiences. For example, the trust of user u for user v is $a\%$ at time t. Also at the time $t+1$, the degree of trust of user u for user v could be $(a \pm 1)\%$ based on their experiences.
- **Context Dependence.** Trust explicitly depends on the context on which it has been shaped. This means that, if user v is trustworthy regarding movie recommendations to user u, he/she may not be trusted in recommendations of IT-related topics to the same extent to the same user. In a recommender system, context refers to which ratings are issued; for example, the location of the users and the items, the time when the user rates an item, and also the items' characteristics which are listed on the users' profiles [29].

Through the asymmetric property, trust-based RS ensures that the degree of the likelihood of a specific item would be different for each user. Furthermore, by the transitivity property of trust, trust-based RS reduces data-sparsity and cold-start issues through trust propagation. Also, this enhances the reliability and accuracy of RS's performance through the dynamicity and context-dependence properties of trust.

Early bird

2.2. Process of recommendation generation

The main job of any recommender system is to generate recommendations efficiently. The trust-based recommender system (TBRS) does the same thing in four phases. The initial phase (trust measurement) is the most vital because a system's performance depends closely on it. A user item-rating matrix and/or user-specified trust values are used as the input of this phase. Furthermore, this phase is divided into two sub-phases: trust calculation, and trust propagation. The output of this phase is a trust matrix that is denoted as $T_{U\times U}$, where U is the set of all users who have a trust relationship in a system. The cell value of the trust matrix (known as the trust value and denoted as tu, v) can be a binary or real number (positive or negative). Usually, a negative trust value defines the degree of distrust between users [17]. The trust calculation sub-phase takes either a user-specified trust value or a user item-rating matrix into account as the input and performs some analysis to generate the trust matrix as the output of the phase (as is presented in Table 1).

By using a trust matrix, a trust network can be constructed. Usually, a trust network is a directed graph. Its nodes denote the users of a system, and the connecting edges between nodes define their trust relationships. The weight of each edge determines the extent of trust or distrust that a user has for other users. A sample of a trust network is demonstrated in Figure 1.

Table 1
Sample of Trust Matrix

	User1	**User2**	**User3**	**User4**	**User5**
User1	–	0.52	0	1	0
User2	0.40	–	–0.65	0	0
User3	0	0.25	–	1	0
User4	0.81	0	0.30	–	–0.50
User5	0.10	0	0	1	–

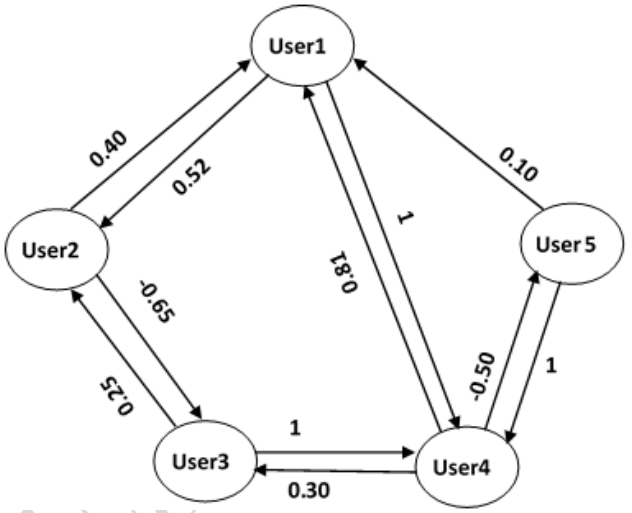

Figure 1. Trust network generated from trust matrix

Usually, an initial trust matrix is sparse, and many cells of the matrix do not contain any direct-trust information between users. For reducing the sparseness of the trust matrix, the trust-propagation method is used by applying the transitive property of the trust and generating an indirect-trust relationship between users based on the calculated trust value of the previous sub-phase. Figure 2 demonstrates an updated trust network by applying trust propagation. The indirect-trust relationships (that is, the result of the trust-propagation sub-phase) are depicted as dashed lines with single arrows in Figure 2, and the bidirectional indirect-trust relationships between users are represented by dashed lines with double arrows. Table 2 represents the output of the trust-measurement phase, where the inferred-trust value is denoted as $\pm x$ (as it depends on the algorithm of the trust calculation and propagation).

The second phrase of TBRS is the neighborhood selection; selected neighbors will play active roles at the time of deducing the target user's preferences.

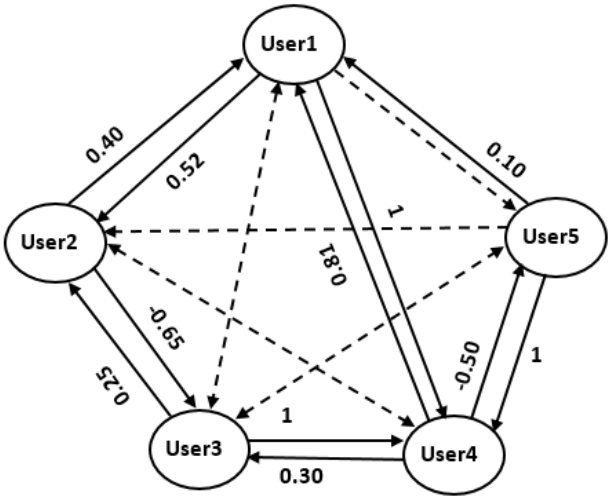

Figure 2. Trust network after propagating trust

Table 2
Final trust matrix after trust-measurement phase

	User1	**User2**	**User3**	**User4**	**User5**
User1	–	0.52	$\pm x$	1	$\pm x$
User2	0.40	–	-0.65	$\pm x$	0
User3	$\pm x$	0.25	–	1	$\pm x$
User4	0.81	$\pm x$	0.30	–	–0.50
User5	0.10	$\pm x$	$\pm x$	1	-

Early bird

The neighborhood-selection process is usually done by filtering the top-trusted users of the target user. This phase takes the final trust matrix as the input (which was the output of the previous phase) and produces a neighbor list as the output of the phase. However, the next phase predicts the users' preferences by aggregating their neighbors' tastes. The input of this phase is the neighbor list of a target user (including their rated items and rating information) and predicting the rating of an unrated item in respect for the target user. One of the popular prediction methods was created by Resnick [60]; this is formulated in Equation 1.

$$p_{u,i} = \bar{r_u} + (r_{v,i} - \bar{r_v}), \tag{1}$$

where $v \in U$ is the trusted user of target user u, $p_{u,i}$ denotes the calculated predicted rating of item i for target user u, $\bar{r_u}$ and $\bar{r_v}$ determine the average ratings

of users u and v, respectively, $r_{v,i}$ indicates the actual rating of item i that is given by user v, and (in TBRS) $W(u, v)$ denotes the amalgamation of trust and similarity.

Finally, the last phase presents the output that could be a prediction, a recommendation, or a ranked list; the output of the previous phase is considered to be the input of this phase. However, in the case of providing predictions of unrated items, the system should predict user's preferences about an item by taking active users and items into account. In the case of recommendations, however, the system should provide a list of items that are the opposite of the user's by taking only the active user as an input. On the other side, the ranked list denotes a set of items that are more related to the active user preferences and collected for a recommendation that is based on predicting the user's interest. Usually, a threshold sets for defining the ranked item list based on the minimum user's interest value.

2.3. Classification of trust-based recommender system

Usually, a trust-based recommender system (TBRS) utilizes a trust relationship of items or users for providing an accurate recommendation. However, the existence of a trust relationship is defined by a numeric value (known as a trust value) that could be binary or any real number (positive or negative). Furthermore, TBRS is divided into user-user trust and item-item trust based on the subject of the trust measurement [8,55]. To calculate trustworthiness, user-user TBRS utilizes either the explicit-trust information of the users [44,75] or gathers the implicit-trust information of the users from a social network [49,58]. On the other side, the reliance of items is measured by applying users' feedback on the items [48] or studying users' activity with these items [12,34,41] in item-item TBRS. However, TBRS can be categorized as memory-based [21, 28, 32] and model-based [30–32, 74] approaches according to the methodology of the trust integration. Furthermore, TBRS can be classified as explicit [15,32,44,72], implicit [5,19,61,77], or hybrid trust-based recommender systems [3,14,59] based on the trust definition. Figure 3 shows the classification of TBRS in a row.

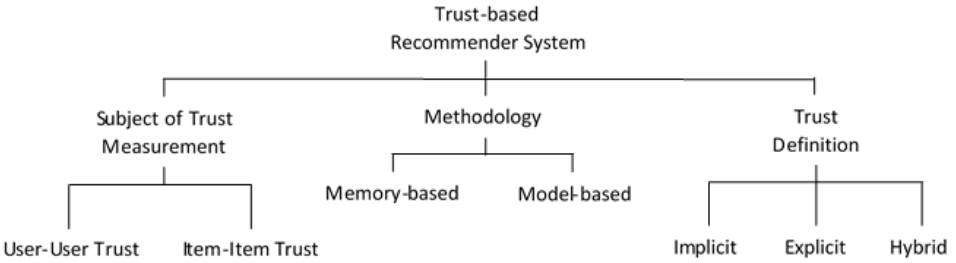

Figure 3. Classification of trust-based recommender system

Early Bird

3. Trust-based recommender system based on trust-definition classification

3.1. Commonly used notations

This section presents a list of frequently used notations in TBRS.

U:	set of all users of RS
I:	set of whole items of RS
R:	set of entire item ratings that are rated by users U
u:	individual user of system (where $u \in U$)
i:	individual item that exists in system (where $i \in I$)
$r_{u,i}$:	rating of item i by users u and $r_{u,i} \in R$
I_u :	set of each item that is rated by user u
$I_{u,v}$:	set of items that are commonly rated by users u and v
$\bar{r_u}$:	average rating of user u
$t_{u,v}$:	degree of trust between users u and v
$sim_{u,v}$:	intensity of similarity between users u and v
$p_{u,i}$:	predicted rating for user u on item i
θ :	threshold for defining trust or similarity
$R_{u \times i}$:	user item-rating matrix (where $u \in U$ and $i \in I$)
$C_{u,v}$:	comprehensive trust between users u and v
$T_{u \times v}$:	trust matrix (where users u & $v \in U$)
r_{max} :	maximum rating of RS (value is 5 in five-scale rating)
r_{min} :	minimum rating of system (value is 1 in five-scale rating)

Early bird

However, we used ETM to denote the explicit-trust metric that distinguishes each trust metric from the others according to the definition of trust. ITM was applied to address the implicit-trust metric, and HTM was also used as the hybrid-trust metric.

3.2. Explicit trust-based recommender system

Usually, an explicit trust-based recommender system (ETBRS) utilizes users' predefined trust connections in a system to improve the system's performance by alleviating existing issues such as data sparsity and cold-start. ETBRS either provides a way for its users to define their trusted users (such as *web of trust*, *trust statement*) or incorporates users' social trust relationships in the system. In both ways, the user plays an active role when defining his/her trust connection, so explicit trust is asymmetric by nature. In ETBRS, the user item-rating matrix and the users' explicit-trust

matrix are taken into account as the input; the output of the system is the list of the predicted ratings of those items that are not rated yet by the respective user. Here, Table 3 presents a sample of a user item-rating matrix by assuming a five-star rating scale system where a cell value denotes a rating that a specific user assigns to a specific item and an empty cell value defines a missing rating. Figure 4 demonstrates the architecture of ETBRS.

Table 3
User item-rating matrix

	Item1	**Item2**	**Item3**	**Item4**	**Item5**
User1		5		1	3
User2	2		5		4
User3	5	5	1	2	3
User4	1	5	3		
User5				3	

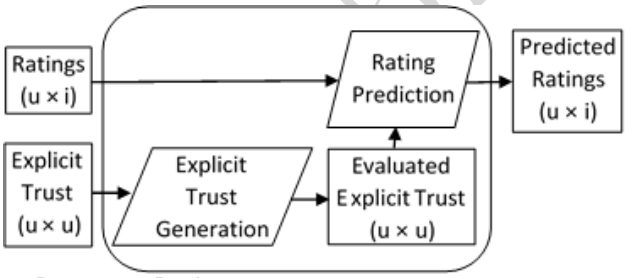

Figure 4. Structure of explicit trust-based recommender system (ETBRS)

Many researchers have proposed that their trust metrics enhance their systems' prediction accuracies. In this study, the following explicit-trust metrics have been explained (denoted as **ETM** prefix).

ETM1: (a) Guo et al. [27] proposed a trust metric called *Merge* by integrating users' social trust information within a system to solve any existing data-sparsity and cold-start issues. In the proposed method, the authors first measured the rating of the target user for a specific item i based on the trusted users' ratings on the same item (which is called "merging the ratings" – as shown in Equation 2).

$$\check{r}_{u,i} = \frac{\sum_{v \epsilon TN_u} t_{u,v} r_{v,i}}{\sum_{v \epsilon TN_u} t_{u,v}}. \tag{2}$$

Here, $\check{r}_{u,i}$ is the merge rating of item i for target user u in respect to the ratings of the trusted users TN_u of user u. The same process is then executed for each item

i of I. The set of merge-rated items is denoted as $\check{I}_u$ and represents target user u's preferences. By using the measured merge ratings, similar users are identified for target user u. For this, the Pearson correlation coefficient (PCC), a popular similarity detection method, is used in this proposed method. After defining the set of uniform users, another set of the nearest neighbors of target user u is selected by using Equation 3.

$$s_{u,v} = \frac{\sum_{i=1}^{I_{u,v}} (\check{r}_{u,i} - \bar{r_u}) \times (r_{v,i} - \bar{r_v})}{\sqrt{\sum_{i=1}^{I_{u,v}} (\check{r}_{u,i} - \bar{r_u})^2} \times \sqrt{\sum_{i=1}^{I_{u,v}} (r_{v,i} - \bar{r_v})^2}} \tag{3}$$

$$NN_u = \{v \mid s_{u,v} > \theta, v \in U\},$$

where $I_{u,v} \in \check{I}_u$, and user $v \notin TN_u$. NN_u indicates the set of the nearest neighbors of target user u. In the end, a rating prediction of the unrated items for user u is made by summing up the similarity and explicit-trust values as follows:

$$\hat{r}_{u,j} = \frac{\sum_{v \in NN_u} s_{u,v} r_{v,j} + \sum_{v \in TN_u} t_{u,v} r_{v,j}}{\sum_{v \in NN_u} s_{u,v} + \sum_{v \in TN_u} t_{u,v}}. \tag{4}$$

(b) Guo also proposed another metric by using explicit trust in the recommender system in [26]. In [26], Guo first applied the merge method to get a merged rating by using the formula of Equation 2. Then, the quality of the merged rating is validated by taking account of the certainty of liked and disliked items, and the formula shown in Equation 5.

$$C_{u,i} = \frac{1}{2} \int_0^1 \left| \frac{x^{p_{u,i}} (1-x)^{n_{u,i}}}{\int_0^1 x^{p_{u,i}} (1-x)^{n_{u,i}} dx} - 1 \right| dx. \tag{5}$$

Here, $C_{u,i}$ denotes the reliability of the merged rating, and $p_{u,i}$ and $n_{u,i}$ are the numbers of liked and disliked ratings, respectively, of user u. After this, the Bayesian similarity measure is used to define the users' similarities by considering the overall similarity, chance correlation (represented as $s''_{u,v}$), and user bias (symbolized as δ). The formula of measuring users' similarity by using the Bayesian similarity measure is presented in Equation 6.

$$s_{u,v} = max(s'_{u,v} - s''_{u,v} - \delta, 0). \tag{6}$$

$s'_{u,v}$ is the overall similarity between users u and v that is measured by inversely normalizing the user distance. The user distance is defined as the mean of the rating distance. The chance correlation is measured by the number of evidence falls at different distance levels independently, and the user bias is 0.04.

The proposed trust methods are asymmetric and transitive by nature, but inferred-trust identification is not taken into consideration.

Early bird

ETM2: Guo et al. [30] offered another metric that incorporates users' social trust relationships in RS to reduce low-accuracy and coverage issues. In [30], a clustering method was used to cluster users according to their rating-pattern similarities and trust relationships; the authors called their applied cluster a "multiview clustering method." Inferred trust is also calculated to strengthen the trust relationship; the formula is shown below in Equation 7. The renowned partitional-clustering method $k-medoids$ algorithm was used in the proposed approach to form a multiview cluster.

$$t_{u,v} = \frac{1}{d_{u,v}}. \tag{7}$$

Here, $d_{u,v}$ is the minimum distance between users u and v, which is identified by a breath-first search in a social trust network.

ETM3: Tian et al. [69] also measured two types of trust in their proposed approach (the inferred trust and comprehensive trust of users) by using their trust relationships of social networks. The authors first defined trust as a triple (such as $T = (U, P, D)$, where U is the set of users, and T, P, and D define the trust relationship, the set of trust paths, and the degree of trust, respectively, between a pair of users). Normally, the intensity of trust is controlled by the length of the trust path, and $L(P) = 2$ determines the direct-trust relationship between users that it gains from a social network. If $L(P) > 2$, then T is defined as the inferred-trust relationship. However, the degree of trust is defined by using the formula from Equation 8.

$$D = \begin{cases} D_{u,v}, & D_{u,v} \in P_n, min(L(P_n) = 2) \\ max(\prod_{D_{u,v} \in P_n} D_{u,v}), & min(L(P_n) > 2) \\ 0, & else. \end{cases} \tag{8}$$

Here, P_n indicates a possible trust path between users u and v, and $D_{u,v}$ denotes the degree of direct trust in path P_n. Tian et al. [69] also incorporated the dynamic nature of trust as the calculation of the degree of direct trust by considering the interactions of the users. The formula of the dynamic update of trust in the degree of direct trust is presented as follows:

$$D_{u,v} = 1 + \sum_{i \in RI_{v,u}} \frac{r_i^a - r_i^b}{r_{max}}, \tag{9}$$

where $RI_{v,u}$ is a set of recommended items for user u by user v, and r_{max} is 5. Also, $r_i^a > r_i^b$ denotes that user u is satisfied with the recommendation that is provided by user v, and the value of $D_{u,v}$ will be increased (which will positively affect $T_{u,v}$), while $r_i^a < r_i^b$ defines the opposite. However, $r_i^a = r_i^b$ determines the complete agreement of user u with the recommendation of user v, and this causes $D_{u,v}$ and

Early print

$T_{u,v}$ to remain unchanged. Afterward, the authors defined the comprehensive trust between users by using Equation 10.

$$C_{u,v} = \frac{D_{u,v}(1 + s_{u,v})}{max_{k \in U_u} D_{u,k} \;\; max_{l \in U_u}(1 + s_{u,l})}. \tag{10}$$

Here, $C_{u,v}$ is the comprehensive trust of user u with v, and U_u determines the set of users in the trust relationship of user u. $s_{u,v}$ is the user's similarity that is measured by using a matrix-factorization method. Afterward, $C_{u,v}$ is applied for the further proceedings of rating the predictions.

$$\hat{t}_{u,v} = \frac{1}{N \sum_{n=1}^{N-1} \frac{1}{\hat{t}_{P_n, P_{n+1}}}}, \tag{11}$$

where N is the number of users who exist on the shortest path between users u and v. Also, P_1 and P_n denote users u and v, respectively, and $P_1 \rightarrow P_2 \rightarrow ... \rightarrow P_n$ indicates the shortest path from user u to v.

ETM5: Duricic et al. [15] offered a method that utilized explicit-trust scores to address the cold-start issue of a system. They applied the direct-trust connections of users to build an adjacency-trust matrix. Afterward, the Katz similarity (KS) measure was used on the adjacency-trust matrix to identify the users' similarities. The formula for the KS measure is as follows:

$$\sigma = \sum_{k=0}^{k_{max}} (\alpha A)^k, \tag{12}$$

where σ denotes the users' similarity matrix, and the single value of the matrix (denoted as $\sigma_{u,v}$) represents the similarity value of users u and v. A indicates the adjacency-trust matrix. Also, $\alpha < \frac{1}{\lambda_A}$, where λ_A is the largest eigenvalue of the adjacency-trust matrix. $k_{max} = 2$ for the proposed method. By using the following formulas, the authors also defined the users' similarities through propagation when a pair of users do not explicitly trust each other.

$$\begin{gathered} \hat{\sigma}_{u,v} = \begin{cases} \sigma^3_{u,v}, & if A_{u,v} = 0 \\ 0 & otherwise \end{cases} \\ \sigma^{(k_{max}+1)}_{Dnorm} = D^{-1}(\sum_{k=0}^{k_{max}} (\alpha A)^k) D^{-1} \\ \sigma_{boost} = A + \hat{\sigma}_{norm} \end{gathered} \tag{13}$$

Here, D denotes the degree matrix of the trusted network. If $\sigma_{boost} = 1$, then it implies users' similarities that are identified with the existence of an explicit-trust connection in adjacency-trust matrix A; otherwise, this is measured through propagation.

Early bird

3.3. Implicit trust-based recommender system

The implicit trust-based recommender system (ITBRS) takes the user item-rating matrix as the input. It detects the trust connection between users by identifying the intensity of the users' rating-pattern similarities from the rating matrix. Also, the output is a list of predicted item ratings for the users. Figure 5 shows the structure of ITBRS.

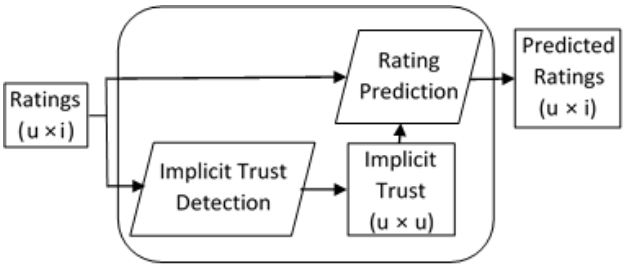

Figure 5. Structure of implicit trust-based recommender system (ITBRS)

Many trust metrics have proposed measuring the implicit trust from users' ratings to improve the recommender system's performance by alleviating existing problems. In this study, 13 implicit-trust metrics are elaborated as follows (denoted as **ITM1**–**ITM13**).

ITM1: Papagelis et al. [57] defined user-user implicit trust based on their rating similarities by using the well-known similarity-measure algorithm called the Pearson correlation coefficient (PCC). After determining the direct trust of the users, the trust-propagation mechanism is also applied to identify the indirect-trust connection between the users in order to eliminate the data-sparsity problem. And, the trust-propagation mechanism is used for positive-implicit trust. The calculation of direct-implicit trust and inferred trust through trust-propagation is presented in Equations 14 and 15.

$$s_{u,v} = \frac{\sum_{i=1}^{I_{u,v}} (r_{u,i} - \bar{r_u}) \times (r_{v,i} - \bar{r_v})}{\sqrt{\sum_{i=1}^{I_{u,v}} (r_{u,i} - \bar{r_u})^2} \times \sqrt{\sum_{i=1}^{I_{u,v}} (r_{v,i} - \bar{r_v})^2}}. \tag{14}$$

Here, $t_{u,v} = s_{u,v}$.

$$t_{u,w} = \frac{|I_{u,v}|}{|I_{u,v}| + |I_{v,w}|} t_{u,v} + \frac{|I_{v,w}|}{|I_{u,v}| + |I_{v,w}|} t_{v,w}. \tag{15}$$

Here, $u, v, w \in U$ are the users, and $t_{u,w}$ is computable if $t_{u,v}$ and $t_{v,w}$ are not negative. Even though the metric is transitive, it is not asymmetric [29].

ITM2: Donovan et al. [55] defined two type of implicit trust. One of these is user-user trust, (known as profile-level trust), and the other is item-item trust (known

as item-level trust). Trust is calculated as the proportion of the correct rating set and the common rating item set (which is used for recommendations) of a pair of users. A rating is treated as correct in the correct rating set if the prediction error is equal to or lower than a conferred threshold. However, Resnick's prediction method is used to determine the predicted ratings [60]. The predicted rating is presented in Equation 1, while the correct rating and trust measurement are shown in Equation 16. This metric is also not asymmetric, and inferred trust is not taken into account.

$$\begin{gathered} correct(v) = correct(r_{u,i}, r_{v,i}) \Longleftrightarrow |p_{u,i} - r_{u,i}| \leq \varepsilon \\ t_{u,v} = \frac{|CorrectSet(v)|}{|RecSet(v)|}. \end{gathered} \tag{16}$$

ITM3: Hwang et al. [37] also proposed a trust metric where the trust score is computed by deriving the mean prediction error on co-rated items between a pair of users. The rating prediction is measured by a straightforward form of Resnick's prediction formula (given in Equation 1). Also, the formula of the trust-score measure of Hwang et al. [37] is shown in Equation 17.

$$t_{u,v} = \frac{1}{|I_{u,v}|} \sum_{i=1}^{I_{u,v}} (1 - \frac{|p_{u,i} - r_{u,i}|}{r_{max}}). \tag{17}$$

Hwang et al. [37] also measured the inferred-trust value by propagating the trust score to solve the data-sparsity limitation and escalate the rating coverage. The inferred-trust value is determined by using Equation 15. The authors also proposed another trust (called "global trust") that takes account of the average of the direct-trust scores of the users. According to the trust property, the proposed trust metric only supports the criteria of transitivity [29].

ITM4: Lathia et al. [43] offered an implicit-trust metric by emphasizing the rating differences between users. This trust metric defines the trust between users if they have even a single co-rated item. Mathematically, trust is defined in [43] as per the following Equation 18.

$$t_{u,v} = \frac{1}{|I_{u,v}|} \sum_{i=1}^{I_{u,v}} (1 - \frac{|r_{u,i} - r_{v,i}|}{r_{max}}). \tag{18}$$

According to the experiments, the authors claimed that the proposed trust metric improved the rating coverage and fixed the data-sparsity issue effectively (although the trust propagation was not considered). However, this metric is also symmetric and transitive by nature, but the rest of the trust properties (such as dynamicity and context dependence) were not taken into consideration.

ITM5: Yuan et al. [76] proposed an implicit binary trust based on the similarities between users. For calculating the users' similarities, they applied PCC (shown in

Early bird

Equation 14). After this, the binary trust was determined by setting two threshold values in Equation 19.

$$t_{u,v} = \begin{cases} 1 & \text{if } s_{u,v} > \theta_s,\ |I_{u,v}| > \theta_i \\ 0 & \text{otherwise} \end{cases}. \tag{19}$$

Here, θ_s and θ_i are the thresholds, and the value of θ_s is 0.75, as PCC becomes transitive if the value of it exceeds 0.707 [10] and θ_i is 2. The authors also considered trust propagation by using Equation 20.

$$t_{u,w} = \frac{\log n / \log k - d_{u,w} + 1}{\log n / \log k}, \tag{20}$$

where n, k, and $d_{u,w}$ denote the size, the mean degree of the implicit trust, and the trust propagation span, respectively, between users u and w.

ITM6: Bedi et al. [6] generated an implicit-trust metric by combining users' similarities with the confidence measure. Usually, confidence is defined as the reliability between users in terms of the number of co-rated items [57]. The users' similarities are measured by taking consideration of the positive value of PCC (shown in Equation 14). Also, the confidence between users is calculated by Equation 21.

$$conf_{u,v} = \frac{|I_{u,v}|}{|I_v|}. \tag{21}$$

After this, the implicit trust is identified by performing a harmonic mean based on the users' similarities and confidence.

$$t_{u,v} = \begin{cases} \frac{2 * s_{u,v} * conf_{u,v}}{s_{u,v} + conf_{u,v}} & \text{if } s_{u,v} \neq 0 \ \&\ conf_{u,v} \neq 0 \\ k * conf_{u,v} & \text{if } s_{u,v} = 0 \ \&\ conf_{u,v} \neq 0 \\ 0 & \text{if } s_{u,v} = 0 \ \&\ conf_{u,v} = 0. \end{cases} \tag{22}$$

Here, k denotes a small constant. Since the confidence depends on the trusted user's ratings, $conf_{u,v}$ may not be similar to $conf_{v,u}$ (which deduces the asymmetric trust between users u and v). Also, the authors used an ant colony algorithm for defining the inferred trust and applied a pheromone that updated the strategy for supporting the dynamic nature of trust. The proposed metric is asymmetric, transitive, and dynamic; however, context dependence was not taken into account.

ITM7: Shambour et al. [63] also used Resnick's prediction formula (shown in Equation 1), the mean squared distance (MSD), and Jaccard to define the proposed implicit-trust metric. The mean squared distance (MSD), Jaccard, and the computed trust are presented in Equations 23, 24, and 25, respectively.

$$MSD_{u,v} = 1 - \frac{\sum_{i=1}^{I_{u,v}} (p_{u,i} - r_{u,i})^2}{|I_{u,v}|}; \tag{23}$$

Early Bird

$$Jaccard_{u,v} = \frac{|I_{u,v}|}{|I_u \cup I_v|}; \tag{24}$$

$$t_{u,v} = Jaccard_{u,v} * MSD_{u,v}. \tag{25}$$

If the calculated trust of a pair of users is above a threshold (λ), then the users are treated as trusted neighbors and are able to process in trust propagation. The authors proposed direct-trust propagation for generating inferred trust, combined trust, and similarity for rating prediction. The proposed implicit trust was transitive, but the asymmetry, dynamicity, and context dependence were not taken into consideration.

ITM8: Roy et al. [61] defined an implicit-trust metric by using Resnick's prediction method [60], the mean squared distance (MSD), and confidence. The authors modified Resnick's prediction method by integrating the users' rating times to emphasize the users' current interests. The modified formula of Resnick's prediction method is presented in Equation 26.

$$p_{u,i} = \bar{r_u} + (r_{v,i} - \bar{r_v})e^{-T\lambda}. \tag{26}$$

Here, λ is a personalized constant that defines the decay rate, and T denotes the time interval between user v's recent rating time and specific rating time of item i. Also, the formula for MSD and confidence are given in Equations 23 and 21, respectively. The formula for the trust metric is presented in Equation 27.

$$\begin{aligned} t_{u,v} &= MSD_{u,v} * Conf_{u,v} \\ &= \frac{I_{u,v} - \sum_{i=1}^{I_{u,v}} \left(\left(\bar{r_u} + (r_{v,i} - \bar{r_v})e^{-T\lambda} \right) - r_{u,i} \right)^2}{I_v}. \end{aligned} \tag{27}$$

Since confidence is asymmetric by nature, the proposed trust is not symmetric. Also, the proposed trust is dynamic, as it considers the users' rating times in the account. On the other side, the proposed implicit trust is potentially transitive, as the authors did not offer any method for identifying inferred trust (they also failed to consider context dependence).

ITM9: Azadjalal et al. [4] proposed a metric by using Pareto dominance and confidence to recognize the most-trusted users of a target user. In the first step, the implicit-trust statements of the users are determined based on their similarities; if these similarities exceed the predefined threshold (θ_t), then the users are treated as the trusted users. The users' similarities are calculated by using the Pearson correlation coefficient (shown in Equation 14). After this, the MoleTrust algorithm was applied to define the inferred trust among the users [50]. The authors also proposed rating imputation by estimating the new rating of an item to reduce the data-sparsity issue and calculate the reliability by validating the estimated ratings. The formula in

Early bird

Equation 5 was used to validate the reliability (denoted as $C_{u,i}$). Also, the formula for rating the imputation is given in Equation 28.

$$\tilde{r}_{u,i} = \frac{\sum_{v \in T_u} t_{u,v} r_{v,i}}{\sum_{v \in T_u} t_{u,v}}. \tag{28}$$

Here, $\tilde{r}_{u,i}$ denotes the estimated rating of item i for user u, and T_u represents the set of the trusted users of user u. Also, $C_{u,i} \in (0,\ 1]$ is the reliability value of the estimated rating $\tilde{r}_{u,i}$, and $p_{u,i} = \mid r_{v,i};\ r_{v,i} > r_{median};\ v \in T_u \mid$ and $n_{u,i} = \mid r_{v,i}; r_{v,i} \leq r_{median};\ v \in T_u \mid$ denote the numbers of like and dislike ratings of item i, respectively, that are rated by all of the trusted users of user u (where $r_{median} = 3$ in the five-star rating scale recommender system). Afterward, the users' confidence is computed by considering the reliability of the estimated ratings in the classical Pearson correlation coefficient (shown in Equation 29). Any confidence values that are higher than the threshold (θ_C) are taken under consideration for the next proceedings.

$$Conf_{u,v} = \frac{\sum_{i=1}^{I_{u,v}} C_{u,i}(r_{u,i} - \bar{r_u}) C_{v,i}(r_{v,i} - \bar{r_v})}{\sqrt{\sum_{i=1}^{I_{u,v}} C_{u,i}^2 (r_{u,i} - \bar{r_u})^2} \sqrt{\sum_{i=1}^{I_{u,v}} C_{v,i}^2 (r_{v,i} - \bar{r_v})^2}}. \tag{29}$$

After defining the confidence between a pair of users, the Pareto dominance concept is applied to determine the set of the most effective trusted neighbors for the target user and the final computed trust value as follows:

$$TW_{u,v} = t_{u,v} \times Conf_{u,v}. \tag{30}$$

Azadjalal et al. [4] used Resnick's prediction formula for predicting recommendations. Although the proposed trust metric is transitive and asymmetric, the dynamicity and context dependence were not taken into account.

ITM10: Choudhary et al. [13] introduced two types of trust metrics; namely, similarity-based and knowledge-based trust metrics. For similarity-based trust, the users' ratings are normalized to 0 to 1, and each item is classified into three categories (liked, disliked, and neutral). Afterward, the similarity-based trust between users is defined based on the items' classifications. The formulas of the ratings' normalization and similarity-based trust calculations are presented in Equation 31.

$$O(u_i) = \begin{cases} 0 & r_{u,i} = min \\ \frac{r_{u,i} - min}{max - min} & min < r_{u,i} < max \\ 1 & r_{u,i} = max \end{cases} \tag{31}$$

$$t_{u,v} = \frac{1}{2} \left[\frac{|LItem_u \bigcap LItem_v|}{|LItem_u|} + \frac{|ULItem_u \bigcap ULItem_v|}{|ULItem_u|} \right].$$

Here, $LItem$ denotes a liked item, where $LItem = \{i : O(u_i) > 0.5\}$ and $ULItem$ determine disliked items where $ULItem =$

Early bird

$\{i : O(u_i) < 0.5\}$. Also, a neutral item is defined as $NItem = \{i : O(u_i) = 0.5\}$, where $NItem$ denotes a neutral item.

For knowledge-based trust, Choudhary et al. [13] considered the rating-pattern similarities between users based on their commonly rated items; for this, the deviations of the ratings of the common items are identified first, and then the deviation is normalized from 1 through 5. Then, the trust is determined by using the formula from Equation 32.

$$\dot{r}_{u,v} = \begin{cases} 5 & 0.0 \leqslant |r_{u,i} - r_{v,i}| \leqslant 0.5 \\ 4 & 0.5 < |r_{u,i} - r_{v,i}| \leqslant 1.0 \\ 3 & 1.0 < |r_{u,i} - r_{v,i}| \leqslant 2.0 \\ 2 & 2.0 < |r_{u,i} - r_{v,i}| \leqslant 2.0 \\ 1 & otherwise \end{cases} \tag{32}$$

$$Kt_{u,v} = \begin{cases} 0 & \dot{r}_{u,v} = 1 \\ \frac{\dot{r}_{u,v}-1}{4} & 1 < \dot{r}_{u,v} < 5 | \dot{r}_{u,v} I_u \\ 1 & \dot{r}_{u,v} = 5 \end{cases},$$

where $\dot{r}_{u,v}$ denotes the indicator of the rating-pattern similarity. Both of the trust metrics are asymmetric, as the similarity-based trust metric is not symmetric. Also, the trust metrics are potentially transitive. However, neither trust metric considers the dynamicity and context-dependence properties of trust. Also, inferred trust is not taken into account.

ITM11: Zahir et al. [77] also applied the liked-and-disliked-items concept in their trust metric and calculated the trust ($AgreeRelTrust$) by combining the users' agreements and relative activities in the system. The agreement $A_{u,v}$ of a pair of users is defined according to the positive and negative agreements of co-rated items, where a positive agreement denotes liked items by both users, and a negative agreement determines the disliked items of both users. The formula for positive and negative agreement as well as $AgreeRelTrust$ are presented in Equation 33.

$$\begin{gathered} posAgreement_{u,v} = |r : R_{(u,r)\in R_v} \cap R_{(v,r)\in R_u} \cap \\ R_{(u,r)} \geq \beta \cap R_{(v,r)} \geq \beta| \\ \\ negAgreement_{u,v} = |r : R_{(u,r)\in R_v} \cap R_{(v,r)\in R_u} \\ \cap R_{(u,r)} < \beta \cap R_{(v,r)} < \beta| \\ \\ A_{u,v} = \frac{posAgreement_{u,v} + negAgreement_{u,v}}{|R_u \cap R_v|}. \end{gathered} \tag{33}$$

Here, β is the separator of the positive and negative ratings, R_u and R_v are the individual rating vectors of users u and v, respectively, and the range of the users'

Early bird

agreement is [0, 1] (where "0" indicates no agreement between users, and "1" denotes complete agreement).

Furthermore, Zahir et al. [77] measured the relative activity of those users who have not rated an item commonly. Equation 34 denotes the formula for measuring the relative activity of user u with respect to user v.

$$RelA_{u,v} = \begin{cases} \frac{1}{1+e^{-ac}} & if \;\; |R_u| + |R_v| > 0 \; AND \;\; v \neq u \\ 0 & else, \end{cases} \tag{34}$$

where $ac = \frac{|R_u|}{|R_u + R_v|}$, and $|R_u|$ and $|R_v|$ denote the lengths of the rating vectors of the respective users. The final trust metric is computed by using Equation 35, where λ and ε are the hyper parameters that manage the engagement of the users' relative activity and agreement, respectively, in the final trust calculation.

$$AgreeRelTrust_{u,v} = A_{u,v}^{\lambda} + \varepsilon RelA_{u,v}. \tag{35}$$

Based on their experiment, Zahir et al. [77] stated that the proposed metric had improved the prediction accuracy with the item coverage and was able to define trust even when the users did not contain any commonly rated item. Also, the calculated trust was asymmetric, as the relative activity among the users was not symmetric.

ITM12: Son et al. [67] proposed an implicit-trust metric by considering the users' relative and asymmetric trust nature of a recommender system. In the proposed metric, the authors first defined the relative similarity between a pair of users based on the average ratings of items; this was then used to generate an asymmetric trust network. After that, trust propagation was applied to identify the inferred trust in order to reduce the data-sparsity problem; this was done by using the shortest-path method. Even though the proposed metric was asymmetric and transitive, the dynamic and context-dependence characteristics were not taken into account. The formula for the direct and inferred-trust calculations are given in Equation 36.

$$\begin{gathered} rs_{u,v} = \begin{cases} \frac{r_{max} - |r_{u,i} - r_{v,i}|}{r_{max} - |r_{u,i} - \bar{r}_i|} & if |I| > 0 \\ 0 & otherwise \end{cases} \\ t_{u,v} = \frac{\sum_{i \epsilon I_{u,v}} (rs_{u,v})_i}{I_{u,v}} \\ Int_{u,v} = \mathbf{max}_{p \epsilon S_{u,v}} \sum_{i=1}^{k} \frac{k-i+1}{k} . t_{a_i, a_{i-1}}. \end{gathered} \tag{36}$$

Here, $rs_{u,v}$, $t_{u,v}$, and $Int_{u,v}$ denote the relative similarity, direct trust, and inferred trust, respectively, between users u and v. Also, $p = u, u_1, u_2, ..., u_k, v$ is the set of users who exits in the shortest path of users u and v. $S_{u,v}$ denotes the list of the total shortest paths from user u to user v in the fixed distance.

ITM13: Barzegar et al. [5] took users' similarities, confidence, analogous opinions, and rating distances into account for measuring the direct-implicit trust of a

Early Bird

pair of users. The trust was asymmetric, as the calculation of the confidence of users was asymmetric by nature. For defining the users' similarities, the Pearson correlation coefficient (PCC) was used (presented in Equation 14) and the confidence was measured by utilizing the formula from Equation 21. However, the rating distance was calculated by using the users' rating intervals of common rated items (as shown in Equation 37).

$$rateDistance_{u,v} = \frac{1}{1 + (\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - r_{v,i})^2})}. \tag{37}$$

However, an analogous opinion between users is computed by measuring the tendency ratio of providing similar ratings to the common rated items of the pair of users. This ratio is defined by three aspects: satisfaction, dissatisfaction, and indifference toward items. In a five-star rating scale system, the users' satisfaction is identified if the ratings of the common rated items are at four or above. If the ratings of the common rated items are below three, these are accounted as the users' dissatisfaction. Also, the indifference is defined if the ratings of a common rated item is between 3 and 4. The satisfaction, dissatisfaction, and indifference calculations are shown in Equation 38. The analogous opinion of a pair of users and the final trust calculation are demonstrated by Equation 39. According to the authors' statement, the trust metric improved the system accuracy, precision, and recall by mitigating the data-sparsity issues. However, the dynamicity and context-dependence properties of trust were not considered at the time of the trust measurement. Also, the inferred trust was not defined for a pair of users.

$$\begin{aligned} Satisfied_{u,v} &= \frac{|I^S_{u,v}|}{|I^S_u \cup I^S_v|} \\ DisSatisfied_{u,v} &= \frac{|I^D_{u,v}|}{|I^D_u \cup I^D_v|} \\ Indifference_{u,v} &= \frac{|I^I_{u,v}|}{|I^I_u \cup I^I_v|} \end{aligned} \tag{38}$$

$$\begin{aligned} similarOpinion_{u,v} &= \frac{Satisfied_{u,v} + DisSatisfied_{u,v} + Indifference_{u,v}}{3} \\ t_{u,v} &= s_{u,v} + conf_{u,v} + similarOpinion_{u,v} + rateDistance_{u,v}. \end{aligned} \tag{39}$$

3.4. Hybrid trust-based recommender system

A hybrid trust-based recommender system (HTBRS) utilizes the advantages of both the explicit and implicit trust of a system by alleviating the limitations of both trusts. If a system only relays explicit trust, then the system will not recommend any item

Early Bird

to its users without the presence of explicit trust. The implicit-trust measurement depends on the users' ratings; in this case, a user typically has no control over it, so it causes a reliability issue in the system. Hybrid trust is a combination of both trusts; this generates valuable and meaningful recommendations by considering the limitations of both trusts. HTBRS takes the user item-rating matrix and the users' explicit-trust matrix as the inputs of the system, and it provides a list of predicted items as the output. Figure 6 shows the architecture of HTBRS.

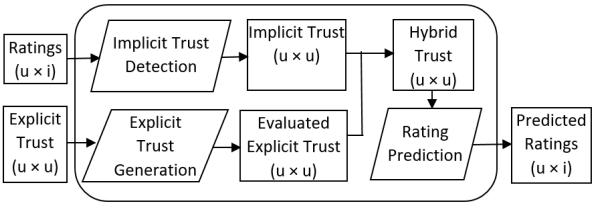

Figure 6. Structure of hybrid trust-based recommender system (HTBRS)

Much research has been done on HTBRS, and different hybrid-trust metrics have been proposed. In this study, five trust metrics are short-listed for a comparative analysis that is based on popularity and publication time.

HTM1: Zheng et al. [79] proposed a hybrid-trust metric for the online community of practices (CoPs) to incorporate user-user explicit and implicit trust. In online CoPs, a learner acquires votes on one's own posts from other learners. Also, the voting actions reflect the author's reputation, who posted the post, and other learners' attitudes toward the post. This voting and user reputation defines the explicit-trust connection from which global trust can be deduced. A learner's learning priorities can be exposed by mining one's own posted textual contents in an online community of practices; hence, accounting for having interests in a common topic allows the users' implicit trust to be deduced. This is called local trust in the CoP perspective. In the proposed method, the authors measured global trust based on the users' (learners') reputation scores and total achieved votes and deduced the users' local trust according to the learning preferences from their own question&answer histories. Afterward, the authors proposed hybrid trust by combining both the global and local trust of the users. The formula for the global, local, and hybrid trust are shown as follows:

$$\begin{gathered} GT_u = a \times Rp_u + (1-a) \times Vote_u, 0 < a < 1 \\ Rp_u = f(x) = (logistic(\tfrac{x}{Rp_{avg}}) - 0.5) \times 2 \\ Vote_u = f(x) = (logistic(\tfrac{x}{Vote_{avg}}) - 0.5) \times 2 \\ logistic(x) = \tfrac{1}{1+e^{(-x)}} \\ LT_{u,v} = s_{u,v} = cos(\theta) = \tfrac{A \cdot B}{\|A\|\|B\|} = \tfrac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \\ HT_{u,v} = b \times GT_v + (1-b) \times LT_{u,v}, 0 < b < 1 \end{gathered} \tag{40}$$

Here, $GT_u, LT_{u,v}$, & $HT_{u,v}$ denote the global, local, & hybrid trust, respectively, of the users. a & b are the parameters that are deduced from the model training (where parameter a balances the constitution's proportions from a user's reputation scores and achieved votes), and b is defined by the constitution's proportions of the local and global trust accordingly. Rp_{avg} denotes the average reputation score of all of the users, and $Vote_{avg}$ is defined as the number of the average received votes of all of the users. However, Latent Dirichlet Allocation (LDA) is applied to execute a text-mining inspection for defining the local trust among the users. However, the hybrid-trust method is not symmetric, and inferred trust, dynamicity, and context dependence are not taken into consideration.

HTM2: Chen et al. [11] proposed a hybrid metric for refining prediction correctness and convergence speed by using both the explicit and implicit trust of users. The authors also offered a new trust (composite trust) by using both trusts. The recommendation task is executed by incorporating it into the probabilistic matrix factorization (PMF). Usually, explicit trust is a pre-defined or manually user-entered value; however, it is in binary format (for privacy concerns), which cannot accurately state the users' trust relationships. By considering this, the authors substituted explicit trust by measuring the incoming and outgoing trust link of a user. Afterward, the implicit trust of the users is measured by deducing the users' similarities that are calculated through PCC and by applying mapping function $f(x) = (x + 1)/2$ (which converts the range of implicit trust into [0, 1] from [-1, 1]). Furthermore, composite trust is defined by using linear regression. The explicit (et), implicit (it), and composite-trust (ct) calculations are shown in Equation 41.

$$
\begin{aligned}
et_{u,v} &= \sqrt{\frac{d^-(V_v)}{d^+(V_u)+d^+(V_v)}} \\
it_{u,v} = s_{u,v} &= \frac{\sum_{i=1}^{I_{u,v}} (r_{u,i}-\bar{r_u})\times(r_{v,i}-\bar{r_v})}{\sqrt{\sum_{i=1}^{I_{u,v}} (r_{u,i}-\bar{r_u})^2}\times\sqrt{\sum_{i=1}^{I_{u,v}} (r_{v,i}-\bar{r_v})^2}} \\
ct_{u,v} &= \beta \times et_{u,v} + (1-\beta)\times it_{u,v}, 0 < \beta < 1,
\end{aligned}
\tag{41}
$$

where $d^+(V_v)$ and $d^-(V_v)$ represent the outgoing and incoming trust links, respectively, of user v. β is a parameter that is obtained from the training model; this was 0.5 for the proposed model. However, $et_{u,v} \neq et_{v,u}$, so $ct_{u,v} \neq ct_{v,u}$. Afterward, Chen et al. [11] applied PMF to perform the recommendation task.

HTM3: Wang et al. [71] used two neural models to enhance the recommendation quality by integrating explicit and implicit trust in order to reduce data-sparsity and cold-start problems. The authors applied one Denoising Autoencoder (DAE) in TBRS (namely, $TDAE$) to incorporate the users' ratings with the explicit-trust connections of social networks to accurately model the users' choices. Another neural network model (called $TDAE++$) pulled out the implicit-trust connections of the users by deducing their rating similarities. Finally, both trust values were inserted into the input and a hidden layer of the neural network to gain more-trustworthy semantic

Early Bird

portrayals of the users. The formula of the explicit and implicit-trust insertion at the input and the hidden layer of the network is shown in Equation 42.

$$\hat{z} = \rho(W'(\rho(W^T\{x_u, t_u\} + b)) + b');$$

$$s_{u,v} = \begin{cases} 1, & u = v \\ (1 - \frac{1}{n})(\frac{PCC_{u,v}+1}{2}) & u \neq v; \end{cases} \tag{42}$$

$$t_{u,v} = \begin{cases} 1 & if s_{u,v} \geq \theta \\ 0 & otherwise. \end{cases}$$

Here, $\hat{z}$ denotes the conclusive portrayal of the output layer, and ρ defines the hyperbolic tangent function. T is the proportion of the trust value that is incorporated into the input or hidden layer. $t_u \in \mathbb{R}^T$ indicates the trust information, and x_u is the input vector for user u. $b \in \mathbb{R}^H$ and $b' \in \mathbb{R}^N$ are the bias vectors; on the other side, $W' \in \mathbb{R}^{T \times H}$ and $W^T \in \mathbb{R}^{H \times (N+T)}$ are the weight matrices. n denotes the total co-rated items between users u and v.

HTM4: Ayub et al. [3] introduced another hybrid metric as an integration of explicit and implicit trust with user choice uniformity to generate a merged rating profile for a specific user. In the proposed metric, the trusted users of a specified user are figured out first and then assembled by the explicit trust that is either delivered by the existing system users or deduced from the trust propagation. For propagating trust, the MoleTrust algorithm was applied to place more importance on trusted users who exist within a short distance [50]. Afterward, the ratings of the assembled trusted users were merged into one value for each item that is not rated by the specified user but rated by a minimum of one trusted user of the specified user. However, if the specified user is not explicitly trusted by others, then the implicit-trust connections are determined by using the ratings that are given by the other users. The implicit trust of the users is measured by applying the **ITM4** method (also mentioned in Equation 18). Onward, the calculated explicit and implicit trust were combined to define the hybrid trust between users. Equation 43 demonstrates the explicit inferred-trust and hybrid-trust formula.

$$et_{u,v} = \frac{1}{d} \times et'_{u,v}$$

$$ht_{u,v} = et_{u,v} \times it_{u,v}. \tag{43}$$

Here, $et_{u,v}, et'_{u,v}, it_{u,v}$ & $ht_{u,v}$ denote the explicit trust, inferred explicit trust, implicit, and hybrid trust of users u and v, respectively. d defines the trust-propagation distance, and the limit is [0, 3]. The user preference uniformity (UPU) is measured by using Jaccard and the user rating preference behavior (RPB). However, RPB is computed through a cosine function that is based on the users' mean rating and variance (as shown in Equation 44).

Early view

$$UPU_{u,v} = Jaccard_{u,v} \times RPB_{u,v}$$
$$RPB_{u,v} = cos(|\bar{R}_u - \bar{R}_v| \times |var_u - var_v|). \tag{44}$$

HTM5: Parvin et al. [59] proposed a metric that utilized the trust statement as an auxiliary information with the ant colony optimization (ACO) method. The proposed approach contains three phases: in the first phase, the users' explicit-trust connection was measured based on the trust statements and inferred trust, and the implicit trust was identified by calculating the users' similarities through PCC. Afterward, both trusts were applied to rank the users based on their trust relationships. In the second phase, ACO was used on the top highly ranked and trusted neighbors in order to identify their importance values. In the last step, the prediction task was executed. Equation 45 presents the formula for the explicit trust-connection identification and the weight calculation for the ranked user.

$$w_{u,v} = \begin{cases} \frac{2 \times s_{u,v} \times t_{u,v}}{s_{u,v} + t_{u,v}} & s_{u,v} + t_{u,v} \neq 0 \; and \; s_{u,v} \times t_{u,v} \neq 0 \\ t_{u,v} & t_{u,v} \neq 0 \; and s_{u,v} = 0 \\ s_{u,v} & t_{u,v} = 0 \; and s_{u,v} \neq 0; \end{cases}$$
$$t_{u,v} = \frac{d_{max} - d_{u,v} + 1}{d_{max}}; \tag{45}$$
$$d_{max} = \frac{ln(n)}{ln(k)},$$

where d_{max} determines the highest propagation limit between the pair of users, and $d_{u,v}$ indicates the trust-propagation length of users u and v. However, k is the mean degree of the trust network, and n represents the number of users that exist between the network.

Early bird

4. Evaluation metrics

After proposing a new metric, each author must evaluate its performance as well as their claims of the benefits of the proposed metric. Various evaluation metrics are applied to validate the efficiency of a system, this is measured by its correctness, coverage, and diversity. Most of the applied evaluation metrics are described here according to the following categories [36]:

4.1. Predictive accuracy metrics

Usually, the metrics that belong in this category measure the closeness between the predictive and true ratings. MAE, iMAE, MAUR, and RMSE are associated with this category.

- **Mean absolute error (MAE)** is a commonly used evaluation metric that measures the level of accuracy of a proposed approach by collating the deviation

of the predicted and real ratings of the items [36]. Usually, the relationship between a system's performance and its MAE is inverse.

- **Inverse mean absolute error (iMAE)** is the transpose of MAE that is normalized by data that is set to the highest and lowest rating scales [28].
- **Mean absolute user error (MAUE)** is an alternative of MAE that measures errors from the user's perspective [4].
- **Root-mean-square error (RMSE)** measures the accuracy of predictions based on the root mean square difference of the predicted and true ratings of the items; lower values of RMSE denote higher prediction accuracy [3].

Mathematically MAE, iMAE, MAUE, and RMSE are defined as follows:

$$\begin{aligned} MAE &= \frac{\sum_{i=1}^{I_u} |r_{u,i} - p_{u,i}|}{I_u}; \\ iMAE &= 1 - \frac{MAE}{R_{max} - R_{min}}; \\ MAUE &= \frac{\sum_{u=1}^{U_u} MAE_u}{N_u}; \\ RMSE &= \sqrt{\frac{\sum_{i=1}^{N_u} |r_{u,i} - p_{u,i}|^2}{N_u}}, \end{aligned} \tag{46}$$

where I_u signifies the number of rated items of user u, and U_u defines the number of users for whom the proposed algorithm could predict at least one rating.

4.2. Suitability metrics

This category contains coverage and is one of the popular metrics for validating a proposed approach's performance. Usually, coverage is applied in order to identify the prediction percentage of a proposed approach [67]. The coverage is divided into two subcategories: user coverage (UC), and rating coverage (RC).

- **User coverage (UC)** denotes the ratio of users for which the proposed approach can predict at least one rating.
- **Rating coverage (RC)** measures the proportion of items for which the algorithm can predict the ratings.

UC and RC are mathematically defined as follows:

$$\begin{aligned} UC &= \frac{N_v}{N_U}; \\ RC &= \frac{N_p}{N_R}. \end{aligned} \tag{47}$$

Here, N_v denotes the users' count for which the proposed approach could predict at least one rating, and N_U is the total number of users who exist in the system. N_p and N_R indicate the numbers of predicted ratings and total ratings, respectively.

Early bird

4.3. Classification accuracy metrics

Classification accuracy metrics assess the occurrence of the accurate or inaccurate predictions of a proposed system by claiming that an item is good. The following metrics are affiliated with this category:

- **Accuracy** denotes the ratio of correct predictions among an entire group that are predicted by a proposed approach [5].
- **Precision** measures the ratio of predicted items that are matched with users' choices in a data set [36].
- **Recall** identifies the ratio of the existence of correctly predicted items among the ranked listed items in a data set.
- If any two evaluation metrics cannot anticipate any decent validation results, then **f-measure (F1)** is applied as a weighted harmonic mean of those evaluation metrics in order to ensure a better evaluation of the proposed approach.
- **Receiver operating characteristic (ROC)** measures the diagnostic power of the proposed approach. Usually, ROC is a curve that is plotted by the recall and 1-specificity [57].

Assuming that positive ratings are within a range of 3–5, and negative ratings are within a range of 1–2 in a five-star rating-based RS, the prediction of an item i can be one of four types:

- if $r_i \in [3,5]$ and $p_i \in [3,5]$, then it can be concluded to be a true-positive prediction (TPP);
- if $r_i \in [1,2]$ and $p_i \in [1,2]$, then it is a true-negative prediction (TNP);
- if $r_i \in [3,5]$ and $p_i \in [1,2]$, then it is called a false-negative prediction (FNP);
- if $r_i \in [1,2]$ and $p_i \in [3,5]$, then it is known as a false-positive prediction (FPP).

Here, r_i and p_i denote the actual and predicted ratings, respectively, of item i. The mathematical form of the accuracy, recall, precision, F1, and (1-specificity) is presented in Equation 48.

$$\begin{aligned}
Accuracy &= \frac{\sum TPP + \sum TNP}{\sum(TPP + TNP + FPP + FNP)} \\
Precision &= \frac{\sum TPP}{\sum(TPP + FPP)} \\
Recall &= \frac{\sum TPP}{\sum(TPP + FNP)} \\
F1 &= \frac{2 \times Precision \times Recall}{Precision + Recall} \\
1 - specificity &= \frac{\sum FPP}{\sum(FPP + TNP)}
\end{aligned} \tag{48}$$

Early bird

However, the relationship between a system's performance and the above-mentioned metrics is proportional.

Another evaluation metric exists that is used to measure the ranking correctness of recommended items by a proposed approach; this is known as normalized discounted cumulative gain ($nDCG$). The mathematical formation of $nDCG$ is as follows:

$$nDCG = \frac{\sum_{q=1}^{p} \frac{rel_q}{log_2(q+1)}}{\sum_{q=1}^{P} \frac{2^{rel_q}-1}{log_2(q+1)}} \tag{49}$$

Here, p denotes an item that exits in a recommended items list, rel_q is the relevancy of an item i at position q of a ranked list of recommendations, and P indicates a list of relevant items of the ranked recommended items.

5. Comparative analysis

This section represents a comparative analysis of 24 different trust metrics, where the metrics are categorized according to the type of trust definition from TBRS. In the following portion, Tables 4, 5, and 6 demonstrate a comparative classification of trust metrics from the perspective of the methodology, trust properties, trust measurement, and evaluation metrics of TBRS.

5.1. Comparison of trust metrics: ETBRS

In Table 4, most of the trust metrics (**ETM1 (b), ETM2, & ETM3**) applied both memory- and model-based methodologies to achieve better performance from each system.

However, all of the trust metrics (**ETM1 through ETM5**) fulfilled the asymmetry and transitivity properties of trust. Only the **ETM3** metric supported the criteria of the dynamicity property of trust by updating the trust value according to the users' interactions. Furthermore, none of the explicit-trust metrics took the context-dependence property into account. In conclusion, all of the mentioned trust metrics partially carried the trust characteristic as per the prospective of the trust property.

Furthermore, Table 4 shows that each trust metric defined the direct-trust relationship between the users and the maximum metrics except for **ETM1 (a) & (b)**; they also defined inferred trust through the trust-propagation method.

Furthermore, the maximum trust metrics (**ETM1 (a) & (b), ETM2, ETM4**) used popular evaluation metrics – that is, MAE – and the second-most-applied evaluation metric (rating coverage [RC]).

Epinions was the most-applied data set according to Table 4 for its sufficient representation of explicit-trust data.

Early bird

Table 4
Comparative analysis of different explicit-trust metrics (ETM)

Trust Metric	Metho-dology	Trust Properties				Trust Measurement		Evaluation Metric	Data sets
		Asymm-etry	Transi-tivity	Dyna-micity	Context Dependence	Trust Calculation (Direct Trust)	Trust Propagation (Inferred Trust)		
ETM1 (a) [27]	Memory-based	Yes	Yes	No	No	Yes	No	MAE & RC	FilmTrust, Flixster, & Epin-ions
ETM1 (b) [26]	Memory- & Model-based	Yes	Yes	No	No	Yes	No	MAE, RC, & F1	FilmTrust, Flixster, ML-1M, & BookCross-ing
ETM2 [30]	Memory- & Model-based	Yes	Yes	No	No	Yes	Yes	MAE, RMSE, & RC	FilmTrust, Flixster, & Epin-ions
ETM3 [69]	Memory- & Model-based	Yes	Yes	Yes	No	Yes	Yes	RMSE	Epinions
ETM4 [35]	Memory-based	Yes	Yes	No	No	Yes	Yes	MAE	FilmTrust
ETM5 [15]	Model-based	Yes	Yes	No	No	Yes	Yes	nDCG, Precision, & Recall	Epinions

Early bird

5.2. Comparison of trust metrics: ITBRS

Each trust metric of Table 5 except for **ITM6** used a memory-based approach in order to generate implicit-trust relationships among the users in RS. **ITM6** also applied a memory-based approach along with the model-based approach to achieve the maximum property of trust.

Furthermore, all of the trust metrics (**ITM1–ITM13**) from Table 5 were transitive. However, the similarity measure-based trust metrics (**ITM1, ITM5, ITM6, ITM9, & ITM13**) where the users' uniformity was deduced by PCC needed to exceed a particular threshold in order to qualify the criteria of transitivity; this threshold value was 0.707. On the other side, **ITM1–ITM5 & ITM7** considered implicit trust as being symmetric, whereas the rest of the trust metrics supported the asymmetric criteria of trust. However, more than 80% of the trust metrics were not dynamic except for **ITM6 & ITM8**, and none of the metrics qualified the criteria of context dependence. According to the trust properties, it can therefore be claimed that the trust metrics that are listed in Table 5 did not fully contain each characteristic of trust.

In terms of the trust measurement, each implicit-trust metric calculated trust in order to form a direct trust between users, whereas nearly 50% of the trust metrics from Table 5 did not consider the inferred trust between users.

Furthermore, 10 out of the 13 metrics used mean absolute error (MAE) to validate the performance; the second-most-applied evaluation metric was coverage (UC and/or RC).

Table 5
Comparative analysis of different implicit-trust metrics (ITM)

Trust Metric	Methodology	Trust Properties				Trust Measurement		Evaluation Metric	Data sets
		Asymmetry	Transitivity	Dynamicity	Context Dependence	Trust Calculation (Direct Trust)	Trust Propagation (Inferred Trust)		
ITM1 [57]	Memory-based	No	Yes	No	No	Yes	Yes (if direct trust is positive)	MAE & ROC	movie recommendation data (MRS)
ITM2 [55]	Memory-based	No	Yes	No	No	Yes	No	MAE	ML
ITM3 [37]	Memory-based	No	Yes	No	No	Yes	Yes	MAE & Coverage	ML-100k
ITM4 [43]	Memory-based	No	Yes	No	No	Yes	No	MAE & Coverage	ML
ITM5 [76]	Memory-based	No	Yes	No	No	Yes	Yes	MAE & UC	Epinions
ITM6 [6]	Memory- & Model-based	Yes	Yes	Yes	No	Yes	Yes	Precision, Recall & F1	ML-100k & Jester
ITM7 [63]	Memory-based	No	Yes	No	No	Yes	Yes	MAE & Coverage	ML-100k & Yahoo
ITM8 [61]	Memory-based	Yes	Yes	Yes	No	Yes	No	MAE	ML-1M
ITM9 [4]	Memory-based	Yes	Yes	No	No	Yes	Yes	MAE, MAUE, UC & RC	Epinions & FilmTrust
ITM10 [13]	Memory-based	Yes	Yes	No	No	Yes	No	nDCG	ML-100k
ITM11 [77]	Memory-based	Yes	Yes	No	No	Yes	No	MAE & RMSE	ML-20M, ML-100k, & Jester
ITM12 [67]	Memory-based	Yes	Yes	No	No	Yes	Yes	MAE, RMSE, UC & RC	FilmTrust
ITM13 [5]	Memory-based	Yes	Yes	No	No	Yes	No	Accuracy, Precision & Recall	ML-100k & ML-1M

Early bird

According to Table 5, the maximum implicit-trust metrics applied different popular MovieLens (ML) data sets such as ML-100k, ML-1M, and ML-20M for verification. However, other known data sets like Epinions, Yahoo, and FilmTrust were also used to measure the benefits of the proposed trust metrics.

5.3. Comparison of trust metrics: HTBRS

Table 6 demonstrates a comparative analysis of five different hybrid-trust metrics, where only **HTM4** was implemented by the memory-based method, and **HTM5**

used both the memory- and model-based methods. The rest of the metrics applied the model-based method.

Table 6
Comparative analysis of different hybrid-trust metrics (HTM)

Trust Metric	Methodology	Trust Properties				Trust Measurement		Evaluation Metric	Data sets
		Asymmetry	Transitivity	Dynamicity	Context Dependence	Trust Calculation (Direct Trust)	Trust Propagation (Inferred Trust)		
HTM1 [79]	Model-based	Yes	Yes	No	No	Yes	No	Precision & Recall	Stack Overflow
HTM2 [11]	Model-based	Yes	Yes	No	No	Yes	No	MAE	Epinions
HTM3 [71]	Model-based	Yes	Yes	No	No	Yes	No	MAE & RMSE	FilmTrust, Epinions, & Douban
HTM4 [3]	Memory-based	Yes	Yes	No	No	Yes	Yes	MAE, RMSE, RC, iMAE, & F1	FilmTrust, CiaoDVD, & Epinions
HTM5 [59]	Model- & Memory-based	Yes	Yes	No	No	Yes	No	MAE, RMSE, & RC	FilmTrust, Epinions, & Ciao

Early bird

However, each trust metric was asymmetric and transitive by nature. None of the trust metrics took the dynamic and context-dependence criteria of the trust property into account.

Furthermore, all of the hybrid-trust metrics determined direct trust among the users, and only the **HTM4** metric measured the inferred trust between users.

Like the explicit- and implicit-trust metrics, the maximum trust metrics from Table 6 applied MAE and RMSE as the assessment metric in order to verify the performance improvements that the authors claimed.

From Table 6, Epinions was the most-used data set for the experiment, and FilmTrust was the second-most-used data set.

6. Conclusion

This paper represents a systematic comprehensive review of different known and upgraded approaches of a trust-based recommender system. However, it also surveyed the task of recommendation and the definition of trust from different aspects with the computational properties of trust. Afterward, different trust metrics were examined by categorizing them into explicit, implicit, and hybrid TBRS. A total of 24 trust metrics were reviewed and compared according to the methodology, trust properties & measurement, evaluation metrics, and data sets.

From Tables 4, 5, and 6, it can be concluded that all of the trust metrics partially carry the trust characteristic according to the trust properties. Only 3 out of the 24

trust metrics qualified the dynamicity criteria of trust, whereas none of the trust metrics took context-dependence criteria into account. Moreover, only 50% of the metrics applied trust propagation to deduce the inferred trust between users.

After analyzing all of the metrics, the following points are advisable to take into consideration:

- Usually, trust between users changes over time; this is why trust is dynamic. In order to qualify the dynamicity criteria of trust and update the trust value over time, the rating time should be considered along with the item-rating information during the formation of the trust metric.
- Since trust values among users vary regarding the context of the items, it is important to know in which context trust is built up. This is why some auxiliary information (such as the contextual and behavioral information of the users and items) should be incorporated into the trust metrics in order to carry out the context-dependence criteria.
- To efficiently resolve data-sparsity and cold-start issues, the trust-propagation technique should be implemented in all trust metrics.

Usually, existing approaches claim that they implement trust in RS to enhance the performance of the existing RS by alleviating some issues, but they implement trust partially without considering the trust build-up context and its dynamic nature. These approaches may also result in some outdated and false trust relationships among users, which could cause some poor recommendations in a system. This is why developing a new trust metric by incorporating user item-interaction time and the demographic information of users is considered to be the next direction of the work.

Early bird

References

[1] Ar Y., Bostanci E.: A genetic algorithm solution to the collaborative filtering problem, *Expert Systems with Applications*, vol. 61, pp. 122–128, 2016.

[2] Ayachi R., Boukhris I., Mellouli S., Amor N.B., Elouedi Z.: Proactive and reactive e-government services recommendation, *Universal Access in the Information Society*, vol. 15(4), pp. 681–697, 2016.

[3] Ayub M., Ghazanfar M.A., Mehmood Z., Alyoubi K.H., Alfakeeh A.S.: Unifying user similarity and social trust to generate powerful recommendations for smart cities using collaborating filtering-based recommender systems, *Soft Computing*, pp. 1–24, 2019.

[4] Azadjalal M.M., Moradi P., Abdollahpouri A., Jalili M.: A trust-aware recommendation method based on Pareto dominance and confidence concepts, *Knowledge-Based Systems*, vol. 116, pp. 130–143, 2017.

[5] Barzegar Nozari R., Koohi H., Mahmodi E.: A novel trust computation method based on user ratings to improve the recommendation, *International Journal of Engineering*, vol. 33(3), pp. 377–386, 2020.

[6] Bedi P., Sharma R.: Trust based recommender system using ant colony for trust computation, *Expert Systems with Applications*, vol. 39(1), pp. 1183–1190, 2012.

[7] Bobadilla J., Ortega F., Hernando A., Bernal J.: A collaborative filtering approach to mitigate the new user cold start problem, *Knowledge-based systems*, vol. 26, pp. 225–238, 2012.

[8] Bobadilla J., Ortega F., Hernando A., Gutiérrez A.: Recommender systems survey, *Knowledge-based systems*, vol. 46, pp. 109–132, 2013.

[9] Castelfranchi C., Falcone R.: *Trust theory: A socio-cognitive and computational model*, vol. 18, John Wiley & Sons, 2010.

[10] Castro Sotos A.E., Vanhoof S., Van Den Noortgate W., Onghena P.: The Transitivity Misconception of Pearson's Correlation Coefficient, *Statistics Education Research Journal*, vol. 8(2), 2009.

[11] Chen C., Zheng X., Zhu M., Xiao L.: Recommender system with composite social trust networks, *International Journal of Web Services Research (IJWSR)*, vol. 13(2), pp. 56–73, 2016.

[12] Cho J., Kwon K., Park Y.: Q-rater: A collaborative reputation system based on source credibility theory, *Expert Systems with Applications*, vol. 36(2), pp. 3751–3760, 2009.

[13] Choudhary N., Bharadwaj K.: Leveraging trust behaviour of users for group recommender systems in social networks. In: *Integrated intelligent computing, communication and security*, pp. 41–47, Springer, 2019.

[14] Davoudi A., Chatterjee M.: Social trust model for rating prediction in recommender systems: Effects of similarity, centrality, and social ties, *Online Social Networks and Media*, vol. 7, pp. 1–11, 2018.

[15] Duricic T., Lacic E., Kowald D., Lex E.: Trust-based collaborative filtering: Tackling the cold start problem using regular equivalence. In: *Proceedings of the 12th ACM Conference on Recommender Systems*, pp. 446–450, 2018.

[16] Gantz J., Reinsel D.: The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east, *IDC iView: IDC Analyze the future*, vol. 2007(2012), pp. 1–16, 2012.

[17] Gao P., Miao H., Baras J.S., Golbeck J.: Star: Semiring trust inference for trust-aware social recommenders. In: *Proceedings of the 10th ACM conference on Recommender systems*, pp. 301–308, 2016.

[18] George G., Lal A.M.: Review of ontology-based recommender systems in e-learning, *Computers & Education*, vol. 142, p. 103642, 2019.

[19] Gohari F.S., Aliee F.S., Haghighi H.: A new confidence-based recommendation approach: Combining trust and certainty, *Information Sciences*, vol. 422, pp. 21–50, 2018.

[20] Gohari F.S., Haghighi H., Aliee F.S.: A semantic-enhanced trust based recommender system using ant colony optimization, *Applied Intelligence*, vol. 46(2), pp. 328–364, 2017.

Early bird

[21] Golbeck J.: Generating predictive movie recommendations from trust in social networks. In: *International Conference on Trust Management*, pp. 93–104, Springer, 2006.

[22] Golbeck J.: Tutorial on using social trust for recommender systems. In: *Proceedings of the third ACM conference on Recommender systems*, pp. 425–426, 2009.

[23] Golbeck J.A.: *Computing and applying trust in web-based social networks (Doctoral dissertation)*, Ph.D. thesis, University of Maryland, College Park., 2005.

[24] Goldberg D., Nichols D., Oki B.M., Terry D.: Using collaborative filtering to weave an information tapestry, *Communications of the ACM*, vol. 35(12), pp. 61–70, 1992.

[25] Gomez-Uribe C.A., Hunt N.: The netflix recommender system: Algorithms, business value, and innovation, *ACM Transactions on Management Information Systems (TMIS)*, vol. 6(4), pp. 1–19, 2015.

[26] Guo G.: Integrating trust and similarity to ameliorate the data sparsity and cold start for recommender systems. In: *Proceedings of the 7th ACM conference on Recommender systems*, pp. 451–454, 2013.

[27] Guo G., Zhang J., Thalmann D.: A simple but effective method to incorporate trusted neighbors in recommender systems. In: *International conference on user modeling, adaptation, and personalization*, pp. 114–125, Springer, 2012.

[28] Guo G., Zhang J., Thalmann D.: Merging trust in collaborative filtering to alleviate data sparsity and cold start, *Knowledge-Based Systems*, vol. 57, pp. 57–68, 2014.

[29] Guo G., Zhang J., Thalmann D., Basu A., Yorke-Smith N.: From ratings to trust: an empirical study of implicit trust in recommender systems. In: *Proceedings of the 29th annual acm symposium on applied computing*, pp. 248–253, 2014.

[30] Guo G., Zhang J., Yorke-Smith N.: Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems, *Knowledge-Based Systems*, vol. 74, pp. 14–27, 2015.

[31] Guo G., Zhang J., Yorke-Smith N.: Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In: *Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 123–129, 2015.

[32] Guo G., Zhang J., Zhu F., Wang X.: Factored similarity models with social trust for top-N item recommendation, *Knowledge-Based Systems*, vol. 122, pp. 17–25, 2017.

[33] Gupta S., Nagpal S.: Trust aware recommender systems: a survey on implicit trust generation techniques, *International Journal of Computer Science and Information Technologies*, vol. 6(4), pp. 3594–3599, 2015.

[34] Hasan M., Roy F.: An Item–Item Collaborative Filtering Recommender System Using Trust and Genre to Address the Cold-Start Problem, *Big Data and Cognitive Computing*, vol. 3(3), p. 39, 2019.

Early print

[35] He X., Liu B., Chen K.: ITrace: an implicit trust inference method for trust-aware collaborative filtering. In: *AIP conference proceedings*, vol. 1955, p. 040102, AIP Publishing LLC, 2018.

[36] Herlocker J.L., Konstan J.A., Terveen L.G., Riedl J.T.: Evaluating collaborative filtering recommender systems, *ACM Transactions on Information Systems (TOIS)*, vol. 22(1), pp. 5–53, 2004.

[37] Hwang C.S., Chen Y.P.: Using trust in collaborative filtering recommendation. In: *International conference on industrial, engineering and other applications of applied intelligent systems*, pp. 1052–1060, Springer, 2007.

[38] Isinkaye F., Folajimi Y., Ojokoh B.: Recommendation systems: Principles, methods and evaluation, *Egyptian Informatics Journal*, vol. 16(3), pp. 261–273, 2015.

[39] Jallouli M., Lajmi S., Amous I.: Similarity and trust metrics used in Recommender Systems: A survey. In: *International Conference on Intelligent Systems Design and Applications*, pp. 1041–1050, Springer, 2016.

[40] Kardan A.A., Ebrahimi M.: A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups, *Information Sciences*, vol. 219, pp. 93–110, 2013.

[41] Kitisin S., Neuman C.: Reputation-based trust-aware recommender system. In: *2006 Securecomm and Workshops*, pp. 1–7, IEEE, 2006.

[42] Koohi H., Kiani K.: User based Collaborative Filtering using fuzzy C-means, *Measurement*, vol. 91, pp. 134–139, 2016.

[43] Lathia N., Hailes S., Capra L.: Trust-based collaborative filtering. In: *IFIP international conference on trust management*, pp. 119–134, Springer, 2008.

[44] Li Y.M., Kao C.P.: TREPPS: A trust-based recommender system for peer production services, *Expert systems with applications*, vol. 36(2), pp. 3263–3277, 2009.

[45] Logesh R., Subramaniyaswamy V.: Exploring hybrid recommender systems for personalized travel applications. In: *Cognitive informatics and soft computing*, pp. 535–544, Springer, 2019.

[46] Lu J., Wu D., Mao M., Wang W., Zhang G.: Recommender system application developments: a survey, *Decision Support Systems*, vol. 74, pp. 12–32, 2015.

[47] Lv G., Hu C., Chen S.: Research on recommender system based on ontology and genetic algorithm, *Neurocomputing*, vol. 187, pp. 92–97, 2016.

[48] Ma H., King I., Lyu M.R.: Learning to recommend with explicit and implicit social relations, *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2(3), pp. 1–19, 2011.

[49] Massa P., Avesani P.: Trust-aware collaborative filtering for recommender systems. In: *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pp. 492–508, Springer, 2004.

[50] Massa P., Avesani P.: Trust metrics on controversial users: Balancing between tyranny of the majority, *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 3(1), pp. 39–64, 2007.

Early Viewed

[51] McNee S.M., Riedl J., Konstan J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: *CHI'06 extended abstracts on Human factors in computing systems*, pp. 1097–1101, 2006.

[52] Miyahara K., Pazzani M.J.: Collaborative filtering with the simple Bayesian classifier. In: *Pacific Rim International conference on artificial intelligence*, pp. 679–689, Springer, 2000.

[53] Moghaddam M.G., Mustapha N., Elahian A.: A Review on Similarity Measurement Methods in Trust-based Recommender Systems, *International Journal of Information Science and Management (IJISM)*, pp. 13–22, 2014.

[54] Nobahari V., Jalali M., Mahdavi S.J.S.: ISoTrustSeq: a social recommender system based on implicit interest, trust and sequential behaviors of users using matrix factorization, *Journal of Intelligent Information Systems*, vol. 52(2), pp. 239–268, 2019.

[55] O'Donovan J., Smyth B.: Trust in recommender systems. In: *Proceedings of the 10th international conference on Intelligent user interfaces*, pp. 167–174, 2005.

[56] Pan Y., He F., Yu H., Li H.: Learning adaptive trust strength with user roles of truster and trustee for trust-aware recommender systems, *Applied Intelligence*, vol. 50(2), pp. 314–327, 2020.

[57] Papagelis M., Plexousakis D., Kutsuras T.: Alleviating the sparsity problem of collaborative filtering using trust inferences. In: *International conference on trust management*, pp. 224–239, Springer, 2005.

[58] Park M.H., Hong J.H., Cho S.B.: Location-based recommendation system using bayesian user's preference model in mobile devices. In: *International conference on ubiquitous intelligence and computing*, pp. 1130–1139, Springer, 2007.

[59] Parvin H., Moradi P., Esmaeili S.: TCFACO: Trust-aware collaborative filtering method based on ant colony optimization, *Expert Systems with Applications*, vol. 118, pp. 152–168, 2019.

[60] Resnick P., Iacovou N., Suchak M., Bergstrom P., Riedl J.: GroupLens: an open architecture for collaborative filtering of netnews. In: *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pp. 175–186, 1994.

[61] Roy F., Sarwar S.M., Hasan M.: User similarity computation for collaborative filtering using dynamic implicit trust. In: *International Conference on Analysis of Images, Social Networks and Texts*, pp. 224–235, Springer, 2015.

[62] Selmi A., Brahmi Z., Gammoudi M.M.: Trust-based recommender systems: an overview. In: *Proceedings of 27th International Business Information Management Association (IBIMA) Conference, Milan, Italy*, 2016.

[63] Shambour Q., Lu J.: A trust-semantic fusion-based recommendation approach for e-business applications, *Decision Support Systems*, vol. 54(1), pp. 768–780, 2012.

[64] Shambour Q., Lu J.: An effective recommender system by unifying user and item trust information for B2B applications, *Journal of Computer and System Sciences*, vol. 81(7), pp. 1110–1126, 2015.

Early bird

[65] Shokeen J., Rana C.: A study on features of social recommender systems, *Artificial Intelligence Review*, vol. 53(2), pp. 965–988, 2020.

[66] Smith B., Linden G.: Two Decades of Recommender Systems at Amazon.com, *IEEE Internet Computing*, vol. 21(3), pp. 12–18, 2017.

[67] Son J., Choi W., Choi S.M.: Trust information network in social Internet of things using trust-aware recommender systems, *International Journal of Distributed Sensor Networks*, vol. 16(4), p. 1550147720908773, 2020.

[68] Son J., Kim S.B.: Content-based filtering for recommendation systems using multiattribute networks, *Expert Systems with Applications*, vol. 89, pp. 404–412, 2017.

[69] Tian H., Liang P.: Improved recommendations based on trust relationships in social networks, *Future Internet*, vol. 9(1), p. 9, 2017.

[70] Wang H., Wang N., Yeung D.Y.: Collaborative deep learning for recommender systems. In: *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1235–1244, 2015.

[71] Wang M., Wu Z., Sun X., Feng G., Zhang B.: Trust-aware collaborative filtering with a denoising autoencoder, *Neural Processing Letters*, vol. 49(2), pp. 835–849, 2019.

[72] Wang X., Liu Y., Lu J., Xiong F., Zhang G.: TruGRC: trust-aware group recommendation with virtual coordinators, *Future Generation Computer Systems*, vol. 94, pp. 224–236, 2019.

[73] Yadav A., Chakraverty S., Sibal R.: A survey of implicit trust on social networks. In: *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp. 1511–1515, IEEE, 2015.

[74] Yao W., He J., Huang G., Zhang Y.: Modeling dual role preferences for trust-aware recommendation. In: *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pp. 975–978, 2014.

[75] Yuan W., Guan D., Lee Y.K., Lee S., Hur S.J.: Improved trust-aware recommender system using small-worldness of trust networks, *Knowledge-Based Systems*, vol. 23(3), pp. 232–238, 2010.

[76] Yuan W., Shu L., Chao H.C., Guan D., Lee Y.K., Lee S.: ITARS: trust-aware recommender system using implicit trust networks, *IET communications*, vol. 4(14), pp. 1709–1721, 2010.

[77] Zahir A., Yuan Y., Moniz K.: AgreeRelTrust: A Simple Implicit Trust Inference Model for Memory-Based Collaborative Filtering Recommendation Systems, *Electronics*, vol. 8(4), p. 427, 2019.

[78] Zhang Z., Liu Y., Jin Z., Zhang R.: A dynamic trust based two-layer neighbor selection scheme towards online recommender systems, *Neurocomputing*, vol. 285, pp. 94–103, 2018.

Early bird

[79] Zheng X.L., Chen C.C., Hung J.L., He W., Hong F.X., Lin Z.: A hybrid trust-based recommender system for online communities of practice, *IEEE Transactions on Learning Technologies*, vol. 8(4), pp. 345–356, 2015.

Affiliations

Falguni Roy
Noakhali Science and Technology University, Institute of Information Technology, Sonapur 3814, Noakhali, Bangladesh, falguniroy.iit@nstu.edu.bd

Mahamudul Hasan
East West University, Department of Computer Science and Engineering, Dhaka 1212, Dhaka, Bangladesh., munna09bd@gmail.com

Received: 24.04.2021
Revised: 23.05.2022
Accepted: 06.06.2022

Early bird