digital

MDPI

Article

Explicit and Implicit Trust Modeling for Recommendation †

Utku Demirci and Pinar Karagoz *

Department of Computer Engineering, Middle East Technical University (METU), Ankara 06800, Turkey
* Correspondence: karagoz@ceng.metu.edu.tr; Tel.: +90-312-210-5518

† This paper is an extended version of our paper published in Demirci, U.; Karagoz, P. Trust Modeling in Recommendation: Explicit and Implicit Trust Model Compatibility and Explicit Trust Prediction. In Proceedings of the 13th International Conference on Management of Digital EcoSystems (MEDES'21), Hammamet, Tunisia, 1–3 November 2021.

Abstract: Recommendation has become an inseparable component of many software applications, such as e-commerce, social media and gaming platforms. Particularly in collaborative filtering-based recommendation solutions, the preferences of other users are considered heavily. At this point, *trust* among the users comes into the scene as an important concept to improve the recommendation performance. Trust describes the nature and the strength of ties between individuals and hence provides useful information to improve the recommendation accuracy, particularly against *data sparsity* and *cold start* problems. The *Trust* notion helps alleviate the effect of these problems by providing additional reliable relationships between the users. However, trust information, specifically *explicit trust*, is not straightforward to collect and is only scarcely available. Therefore, *implicit* trust models have been proposed to fill in the gap. The literature includes a variety of studies proposing the use of trust for recommendation. In this work, two specific sub-problems are elaborated on: the relationship between explicit and implicit trust scores, and the construction of a machine learning model for explicit trust. For the first sub-problem, an implicit trust model is devised and the compatibility of implicit trust scores with explicit scores is analyzed. For the second sub-problem, two different explicit trust models are proposed: Explicit trust modeling through users' rating behavior and explicit trust modeling as a link prediction problem. The performances of the prediction models are analyzed on a set of benchmark data sets. It is observed that explicit and implicit trust models have different natures, and are to be used in a complementary way for recommendation. Another important result is that the accuracy of the machine learning models for explicit trust is promising and depends on the availability of data.

Keywords: trust modeling; implicit trust; explicit trust; recommendation; recommender systems; supervised learning; one class classification

Citation: Demirci, U.; Karagoz, P. Explicit and Implicit Trust Modeling for Recommendation. *Digital* **2022**, *2*, 444–462. https://doi.org/10.3390/digital2040024

Academic Editors: Yannis Manolopoulos, Mirjana Ivanović and Richard Chbeir

Received: 3 August 2022
Accepted: 24 September 2022
Published: 29 September 2022

1. Introduction

Recommendation has become an indispensable part of software systems, particularly e-commerce and online streaming applications such as Spotify (spotify.com) and Netflix (netflix.com), alleviating the load of search for users in a vast item collection and positively affecting the perception of the users about the applications through improved user experience [1]. Recommender systems process user history to generate recommendations. One way of obtaining a user' previous experience on an item is through explicit ratings. As an alternative way, implicit rating indirectly provides information about the user's opinion on an item, based on activities of the user such as clicking, searching some keywords, purchases, etc., and gives hints for the user's intent and interest [2]. The Collaborative Filtering (CF) method [3], being a popular recommendation technique, uses the similarity between past preferences of users. However, CF suffers from well-known *data sparsity* and *cold start* problems. In order to overcome the performance degrading due to such problems, recommender systems employ a variety of auxiliary information including product details

of the items, social network of users or external contextual information such as weather or currency rate [4].

In social relationships, concepts such as *trust* and *loyalty* are important and useful to describe and quantify the nature of the relationship between users [1]. Loyalty expresses the strength of the tie between a user and an object or environment, whereas trust is merely about the relationship between the users in an environment, such as a social network. In recommendation methods, specifically in CF, these concepts provide valuable information to improve recommendation performance, and hence it has been used within recommender systems in the literature mostly to overcome the aforementioned issues of CF. Particularly, trust information helps to reduce the data sparsity through enrichment with the ratings of trusted neighbors. It is also useful against the cold start problem as the preferences of trusted neighbors or trusted users, in general, can provide a basis for recommendation.

In trust-aware recommendation studies, two types of trust data are used: *explicit* and *implicit trust*. Explicit trust is obtained through user feedback on other users. A well-known example is Epinions (http://www.epinions.com/help/faq/?show=faq_wot, accessed on 3 August 2022), which is a website of product reviews. It uses a trust system such that users can define their *web of Trust*, which is a set of reviewers whose reviews and ratings are consistently found to be useful, and their *block list*, which includes reviewers that a user consistently finds inaccurate or not useful (http://www.trustlet.org/epinions.html, accessed on 3 August 2022). The data set crawled from The Epinions website, namely *epinions data set*, has been popularly used as explicit trust data in various studies [5–8]. Explicit trust networks can be *unsigned*, including only positive trust links, or *signed*, where both negative and positive trust links are available.

On the other hand, *implicit trust* provides information about the trust relationship between users indirectly, generally through activities and behavior of users [5,9]. Since explicit trust information is scarcely available, and it is mostly sparse, several studies focus on generating implicit trust by using other data sources such as the rating data and social connection of users [10]. For example, in [11], interest similarity is used for inferring trust between two users, whereas in [9], trust propagation over a social network is employed for constructing the trust network of a given user.

Research Questions. Trust information has been used in a variety of recommendation studies both in explicit and implicit form and it has been shown that it improves recommendation accuracy [9,12,13]. In this work, focusing on a different aspect of trust-aware recommendation, the following two sub-problems of explicit and implicit trust are analyzed in the recommendation setting:

- What is the relationship between explicit and implicit trust scores? Are they replaceable?
- Would it be possible to construct a machine learning model of the explicit trust in a trust network feasibly?

The first one is about examining the compatibility between explicit and implicit trust scores. For this analysis, an implicit trust model is devised, and by using this implicit trust model, the matching between implicit and explicit trust scores is analyzed. This analysis is crucial for understanding the nature of implicit and explicit trust and using them in either a complementary way or as a replacement.

The second sub-problem is about constructing a machine learning model for explicit trust in order to predict missing trust relationships in a trust network. In this way, the data sparsity in explicit trust information can be reduced. There are two types of explicit trust networks: an *unsigned* network with only positive links and a *signed* trust network with negative and positive links. In an unsigned trust matrix, trust information is explicitly expressed as 1 to denote trust. However, 0 as the trust value may indicate either a neutral or unknown trust relationship. For this, two different explicit trust models are generated. In the first model, users' rating behavior is exploited for explicit trust modeling. A trust graph is generated in the second approach, and the problem is specified as a link prediction problem. In the graph model, trust value 1 in the matrix denotes a link, whereas trust value

0 shows that there is no edge between the given nodes (i.e., users). It is aimed to predict the missing trust relationships in the trust graph by constructing an explicit trust model. The effect of augmenting the trust matrix through the proposed approach is analyzed through trust-based recommendation methods in the literature.

Contributions. A preliminary version of the study is published in [14]. In this paper, the study is extended both with more detailed explanations and discussions, and additional machine learning models, algorithms and their analysis. For the explicit and implicit trust model comparison part, the approach and analyses are described in more detail. Similarly, in explicit trust modeling, descriptions of the proposed approaches are given with additional explanations. As a new modeling approach in this paper, unsupervised machine learning algorithms are applied for explicit trust modeling, and an outlier detection-based model is developed based on Isolation Forest and One-Class Support Vector Machine (SVM). Additionally, the explicit trust model is constructed by SVM in addition to Random Forest and Naive Bayes classifier. For all the experiments, the results are further discussed and elaborated on.

The contributions of this study can be summarized as follows:

- An implicit trust model is devised, which is adapted from the consistency model for reputation scores of users in [15]. This model is used for compatibility analysis of implicit and explicit scores.
- The implicit trust model generates a single score per user. In contrast, the available explicit trust data sets inform about the trust relationship between two users. To overcome this incompatibility, a mapping schema is proposed such that an explicit trust score per user is generated by using the explicit trust graph.
- A supervised learning model is constructed for explicit trust score prediction by using the ratings that users give to model explicit trust data. This method is used for both signed and unsigned trust data.
- Another explicit trust score prediction model is constructed such that finding an explicit trust between two users is considered an edge prediction problem and a supervised learning model is generated to predict unknown trust values. The effectiveness of an augmented trust network is analyzed through recommendation performance.

Organization. The rest of the paper is organized as follows. In Section 2, related studies in the literature are summarized. The methods proposed and employed in this study are presented in Section 3. The experiments and results are presented in Section 4. Finally, Section 5 concludes the paper with an overview and future work.

2. Literature Review

Trust-aware recommendation is a challenging research problem and there is a variety of solutions that focus on the use of the trust information to improve the accuracy of recommendations, particularly alleviating cold start and rating data sparsity problems.

As one of the initial trust-based studies, in [11], Htun and Tar consider trust as a solution to cold-start problems in recommender systems. To this aim, explicit trust ratings are used for neighbor formation. Since reliable explicit trust data is rarely available, the authors propose a method to derive implicit trust relationships based on the similarity of user interests. Trust between users is measured according to the following similarity measures: user interest similarity, resource item similarity, and interest similarity on resource items. The resulting trust metric is incorporated into the recommender system. The performance of the proposed approach is reported to outperform traditional CF.

In [16], Chen et al. propose a cold start recommendation method that integrates the user model with trust and distrust for each new user. With the proposed approach, trustworthy users can be identified by analyzing the web of trust of experienced users. In the proposed method, a user model is constructed by using a clustering algorithm to group experienced users into clusters. Each cluster is formed with users that have similar item

preferences. A web of trust is constructed for each cluster and the PageRank algorithm is used for finding experienced users in the cluster. The authors use distrust networks to find unreliable users in a similar way. Following this, the most closely related cluster is identified for a cold start new user to predict an unrated item's possible rating. Previously identified experienced users in the cluster are exploited to recommend new cold-start users. Moreover, the proposed method identifies implicit trust links between users by exploiting the given rating.

In another study [13], Guo et al. propose three factored similarity models that use social trust based on implicit user feedback. The proposed trust-based recommendation approach generates top-N item recommendations based on social trust relationships between users. In [17], the authors develop another trust-based recommender that uses explicitly specified social trust information for generating recommendations. The method merges the ratings of a user's trusted neighbors in order to find similar users.

In [12], Yang et al. propose TrustMF, a matrix factorization-based method that fuses rating and trust information. TrustMF defines two models: the *truster* model which denotes how others will affect user u's preferences and the *trustee* model which denotes how user u will affect others' preferences. The main motivation for the use of truster and trustee models is to link ratings and trust information.

In order to overcome accuracy issues due to cold start and data sparsity, in [9], Li et al. propose an implicit trust recommendation approach (ITRA) that utilizes implicit user information. The method generates a set of trusted neighbors of a given user by exploiting the social network and trust diffusion features in a trust network. After finding the trust neighbor set, trust values are determined by computing the shortest distance between a user and inferred trusted neighbor.

In [18], Wang et al. introduce TeCF, a trust-enhanced collaborative filtering method that integrates user-based, item-based, and trust-based techniques to predict unrated items. The conducted experiments show that the proposed approach significantly reduces the effects of data sparsity by making the rating matrix denser.

The trust model of the SSL-SVD method in [5] incorporates social trust (explicit trust) and sparse trust (implicit trust) information to improve recommendation accuracy. In the study, Hu et al. report that social trust is influenced by many social factors and has a limited effect on improving the accuracy of recommendations.

In recent studies, neural network-based solutions are also employed in trust-aware recommender systems. In [19], a trust network is used in order to determine reliable implicit ratings of users. Once the rating profile of a user is augmented with such ratings, latent features of users are derived by using a deep representation model. The recommendation is generated based on the similarity of users through their latent feature representations.

As seen in the above-mentioned studies, the nature of trust information used in the recommendation and how it is incorporated varies; however, it is reported that overall the use of trust information has a positive effect on recommendation performance. In this study, another aspect of the use of trust modeling in the recommendation is focused on. The nature of implicit and explicit trust modeling and their compatibility are analyzed to further increase this positive effect.

3. Proposed Methods for Trust Modeling and Comparison

In this section, the compatibility analysis of explicit and implicit trust models, and generating explicit trust prediction models are described in detail. The symbols used in the formulas are given in Table 1.

Table 1. The list of symbols.

Notation	Explanation
u	User
r	Rating
m	Item
R_u	Set of ratings by user u
r_m	Average rating of item m
r_u	Average of the ratings given by user u
$r_u m$	The rating given by user u to item m
O_u	Conformity of user u
$O_{r_{um}}$	Conformity of rating r by user u for item m
C_u	Consistency of user u
s_m	Standard deviation of ratings for item m

3.1. Compatibility Analysis of Explicit and Implicit Trust Models

In the literature, trust information is reported to improve the quality of recommendation accuracy [20]. However, explicit trust data is scarcely available. Therefore, there are studies inferring implicit trust between the users and the trust value of a user from other sources, such as the behavior of the user. However, the relationship between explicit and implicit trust is not always clear and the number of studies focusing on such analysis is limited [5]. Thus, in this work, it is investigated how compatible implicit and explicit trust scores are. This analysis is crucial to be able to understand whether these two models have an overlapping or complementary nature.

The overview of the proposed approach for the compatibility analysis is shown in Figure 1. As the first step, the implicit trust score model is constructed using the rating data of users, and implicit trust scores are generated for each user with this model. In parallel, explicit trust scores are generated for each user based on explicit trust data. Finally, the top-k elements of both lists are compared for the compatibility check.

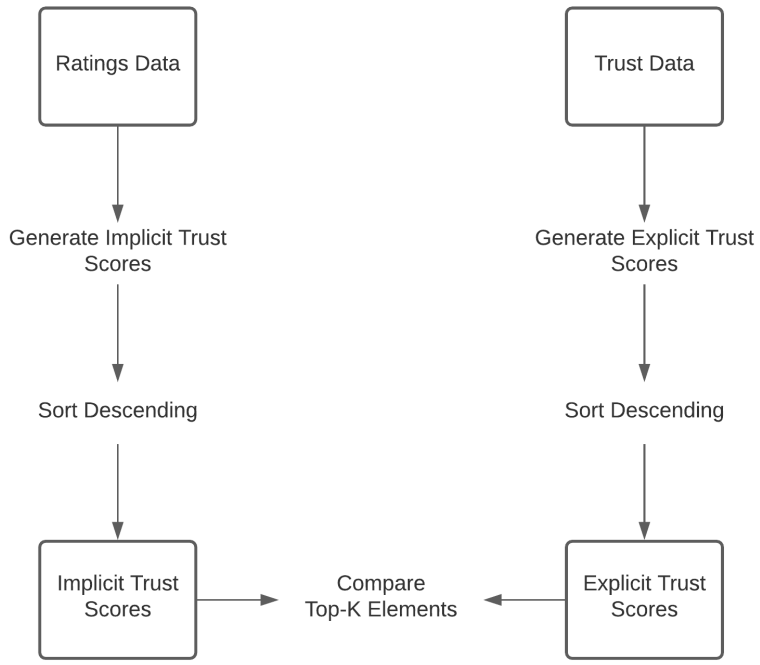

Figure 1. Overview of the process of analyzing the compatibility of the implicit and explicit scores.

3.1.1. Implicit Trust Model

Trust and reputation are important concepts in social network analysis, as in recommendation. In [15], Oh and Kim introduce the mathematical models for *activity*, *objectivity*, and *consistency* of social media users. These are used for calculating the reputation scores of users.

In this work, the features presented in [15] are adapted for generating the implicit trust scores of users (a similar adaptation of Oh and Kim's features in [15] for calculation of implicit trust scores for location-based social networks is also presented in [21]. In this study, these mathematical models are adapted for rating data). In the proposed implicit trust model, users with a high count of ratings (above a given threshold) are considered as *active* users. The conformity of a rating, $O_{r_{um}}$, is a measure of whether a rating r by user u on item m differs from the average rating on item m (denoted as $\bar{r}_m$). s_m denotes the standard deviation of the ratings on item m. Conformity of a given rating increases as $O_{r_{um}}$ approaches zero (Equation (1)).

$$O_{\mathrm{r_{um}}} = \mid \frac{r_{\mathrm{um}} - \bar{r}_m}{s_m} \mid \tag{1}$$

The conformity of user, O_u, is the average of ratings, $O_{r_{um}}$, by user u. As in the rating conformity, as the value gets closer to zero, the conformity of the user is considered to be higher (Equation (2)). Here, R_u denotes the number of ratings by user u.

$$O_u = \frac{1}{\mid R_u \mid} \sum O_{r_{um}} \tag{2}$$

If the user u behaves similarly to other users in the system, it can be inferred that the user's behavior is consistent. The consistency of a user, C_u, is defined as the variation in conformity of her/his own evaluations (Equation (3)). In the proposed approach, the C_u score of a user u is used as the *implicit trust score*.

$$C_u = \frac{1}{\mid R_u \mid} \sum_{r \in R_u} (O_r - O_u)^2 \tag{3}$$

3.1.2. Construction of Explicit Trust Score per User

The implicit trust model generates a trust score per user. On the other hand, explicit trust in the network is not a per-user score, it rather indicates the trust relationship between two users. To provide compatibility between implicit and explicit trust models, a mapping schema is defined that generates an explicit trust score per user from the explicit trust graph.

The proposed mapping schema is as follows: given a user in an *unsigned* trust network, the number of incoming trust edges is determined as the explicit trust score per user. For *signed* networks, the number of incoming edges with weight 1 denotes the trust score per user. Similarly, the number of incoming edges with weight -1 is the *distrust score* of the user. For example, for an unsigned trust network, if the node of *usera* has 10 incoming edges, this denotes that 10 users trust this user. Then, the explicit trust score of *usera* is set as 10.

As an alternative mapping schema, the well-known PageRank algorithm [22] is used. In order to generate a trust score per user, the PageRank algorithm is applied to the trust network. This scoring also gives the ranking of the users in the trust network, which is used for comparison with implicit trust score ranking. Although there are several other personalized node ranking algorithms proposed for signed networks in the literature [23], in this study, the conventional PageRank algorithm is used for both unsigned and signed trust networks.

3.1.3. Comparison of Explicit and Implicit Trust Scores

After generating the implicit and explicit trust scores per user, a comparison schema is applied to them. Each set of scores is sorted separately in descending order. The compatibility of the implicit and explicit scores is analyzed as the overlapping of users on top-k% items between the sorted implicit and explicit trust scores. The analysis results conducted on three data sets are presented and discussed in Section 4.3.

3.2. Explicit Trust Modeling

The basic motivation for constructing a supervised explicit trust model is to be able to estimate the unknown values in the explicit trust matrix and augment the trust graph, thus increasing the accuracy of trust-based recommendation. Two different approaches are proposed for explicit trust modeling. In the first one, explicit trust is modeled by using rating behavior. In the second approach, an explicit trust network is created, and missing links between users are aimed to be determined with link prediction.

3.2.1. Explicit Trust Modeling through Rating Behavior

The proposed approach aims to construct a supervised learning model to predict explicit trust scores by using rating information-based features. The explicit trust predictions are used for augmenting the available explicit trust data. The overview of the proposed approach is shown in Figure 2.

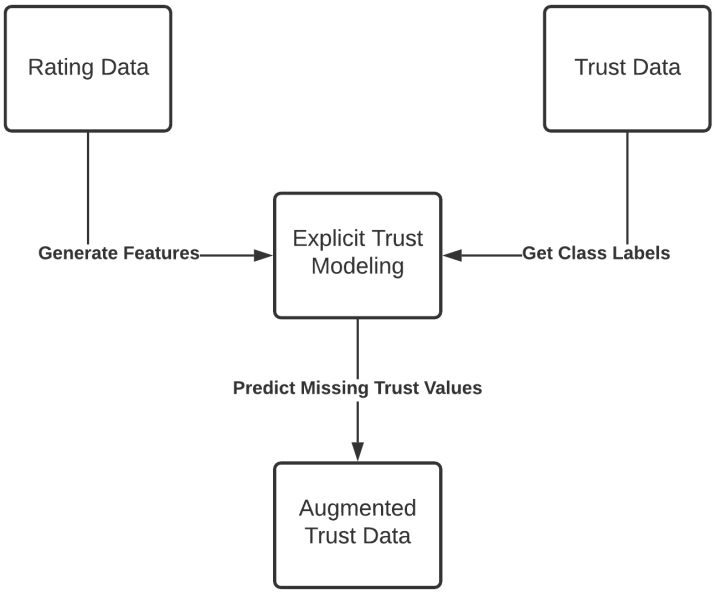

Figure 2. Overview of the process of explicit trust modeling through user's rating behavior.

In this approach, the problem is considered a classification task. Therefore, a set of features are generated for each explicit trust relationship by using the ratings given by the users. While generating those features, the concepts of *liked item* and *disliked item* are considered. The definition of these concepts is given in Equation (4). Here, r_{um} denotes the rating of the user u for item m, and r_u denotes the average rating for user u.

$$isItemLiked = \begin{cases} True, & \text{if } r_{um} \geq r_u \\ False, & \text{otherwise} \end{cases} \quad (4)$$

As reflected in the equation, an average rating score is created for each user by checking the ratings given by the user. Then, the rating given to an item by the target user and the average rating score of the user is compared. If the rating given to the item is greater than or equal to the average rating score of the user, the item is considered to be a *liked item*,

otherwise, it is considered as a *disliked item*. The procedure is described in Algorithm 1. For instance, suppose a user gives the ratings of 1.0, 2.0, 5.0 to the movies a, b, and c, respectively. Since the average rating of the user is 3.0, it is considered that the user disliked movies a and b and liked the movie c.

Algorithm 1 Discovering Liked and Disliked Items for Each User

```
U: a set of users
R: a set of ratings
procedure DiscoverLikedAndDislikedItems(U, R)
    Build R_u, a list of ratings given by each user u using U and R
    for each user u in R do
        Calculate average rating a_u for u
        for each rating r in R_u do
            if r ≥ a_u then
                rated item is a liked item for user u
            else
                rated item is a disliked item for user u
            end if
        end for
    end for
end procedure
```

Given two users, the target model aims to predict the nature of the trust between them. Given two users, trustor and trustee, the features constructed over them for the supervised learning model are as follows:

- The number of mutual rated items;
- The number of mutual liked items;
- The number of mutual disliked items;
- The average of the ratings given by the trustor;
- The number of the ratings given by the trustor;
- The average of the ratings given by the trustee;
- The number of the ratings given by the trustee.

For each user, separate lists are created for the rated items, liked items, and disliked items. By scanning these lists, the intersection of rated, liked, and disliked items between two users can be found easily.

The explicit trust value between two users is the class label in each trust relationship. However, the value of the class label varies depending on whether the trust network is signed or unsigned. Hence, the model construction follows two different mechanisms:

- For the signed trust network, the class labels are 1 and -1. In this case, the problem can be considered a binary classification problem. For modeling signed explicit trust data, SVM, Random Forest, and Naive Bayes classifier algorithms are used. These supervised learning algorithms are preferred since they have been successfully applied for prediction problems in a variety of domains.
- The class labels are slightly different for unsigned trust networks. Since there is no distrust information in such data, every relationship in the network is expressed with 1. In this case, the problem is considered an outlier/novelty detection or a one-class classification problem. Isolation Forest and One-class SVM algorithms are used for modeling unsigned explicit trust data.

In both of the cases, new trust relationships are predicted with these models, and the explicit trust network is updated with the predicted trust relationships. The effect of updated/augmented trust networks is analyzed through various trust-based recommendation algorithms. The related experiments are described in Section 4.4.

3.2.2. Trust Prediction Modeling as a Link Prediction Problem

For explicit trust modeling, another approach is also devised such that the problem is considered as an edge (link) prediction task on the directed trust network. More specifically, a supervised learning model is built for the *inference of explicit trust* between users by using features extracted from the trust network. The process is visualized in Figure 3.

In this classification task, the edges correspond to class labels. For unsigned trust networks, an edge denotes a trust relationship, and it is represented with *class label 1*. The rest of the (non-existing) edges in the graph are assumed to correspond to *class label 0*. For signed trust networks, the setting has a slight difference such that the edges are labeled (signed) as either 1 or -1, denoting trust or distrust, respectively.

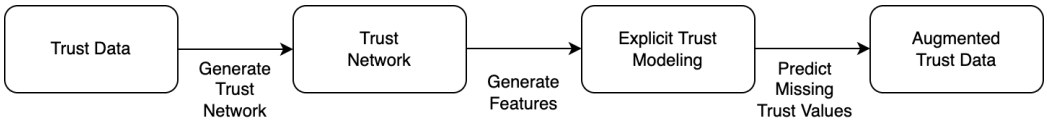

Figure 3. Overview of the process of trust prediction modeling as a link prediction problem.

To determine a balanced set of training instances, links with 0 labels are randomly included in as many as the number of trust links. On the unsigned trust graph, for each edge, the following features are extracted:

- Jaccard similarity for destination (trustee) and source (trustor) nodes;
- Cosine similarity for destination and source nodes;
- Katz centrality for destination and source nodes;
- Adar Index [24] for destination and source nodes;
- Number of nodes that trust the source node;
- Number of nodes that trust the destination node;
- Number of nodes that the source node trusts;
- Number of nodes that the destination node trusts;
- Intersection of the nodes that trust source and destination nodes;
- Intersection of the nodes that both source and destination nodes trust;
- Trust back;
- The shortest trust path between nodes.

As given in the list, a total of 12 features are extracted from the trust graph. Among the features given above, *Adar Index* is a measure to predict links in a network by using the shared links between two nodes. *Trust back* is a binary field that denotes whether the destination node trusts the source node back or not. For calculating the shortest trust path between nodes, firstly, if they have been already connected, the link between them is deleted. Then, the shortest path between the nodes is computed.

Before constructing the supervised learning model, feature elimination is applied by using Extra-Trees Classifier [25] and Jaccard similarity, Cosine similarity, Katz centrality, and Adar Index features are filtered out. As the supervised learning algorithms, Random Forest Classifier and SVM Classifier are employed [25] to construct the explicit trust model.

4. Experiments

4.1. Data Sets and Experiment Environment

The experiments are conducted on MacOS Catalina, Intel(R) Core(TM) i5 CPU @1.4 GHz, 16 GB of RAM. The proposed methods are coded in Python programming language by using scikit-learn [25], and RecQ [26] frameworks.

For the analysis, Epinions (Unsigned) [27], Epinions (Signed) [28], FilmTrust [29], and Ciao [30] data sets are used. All of those data sets are frequently used for recommendation systems analysis, specifically in trust-based systems. The statistical details about the data sets are given in Table 2. FilmTrust is a platform that allows its users to evaluate the movies they watch. Epinions is a social networking site where users can share their opinions about

various products and express their trust network. Ciao is a product review and online shopping portal that contains trust relationships between users.

Table 2. Statistics on the data sets.

	# of Users	# of Items	# of Ratings
FilmTrust	1508	2071	35,497
Epinions (Unsigned)	75,888	29,000	681,213
Epinions (Signed)	132,492	755,760	13,668,320
Ciao	7375	105,114	284,086

4.2. Evaluation Metrics

In this study, the following metrics are used for measuring prediction performance.

Accuracy measures the proportion of correct predictions among the total number of predictions (Equation (5)).

$$Accuracy = \frac{True\ Positive + True\ Negative}{All\ Predictions} \tag{5}$$

Precision measures the number of positive class predictions that actually belong to the positive class (Equation (6)).

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{6}$$

Recall measures the number of positive class predictions made from all positive samples (Equation (7)).

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{7}$$

F1-score provides a single score that balances both precision and recall as their harmonic mean in one score (Equation (8)).

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{8}$$

Outlier Ratio shows how many samples in the test data are determined as outliers (Equation (9)).

$$Outlier\ Ratio = \frac{Number\ Of\ Outliers}{Number\ Of\ Test\ Samples} \tag{9}$$

4.3. Implicit and Explicit Trust Models Compatibility Analysis Results

In explicit and implicit trust model comparison, the implicit trust scores are generated as described in Section 3.1 in various configurations by filtering users according to the number of activities. The users are filtered with respect to the number of ratings given and the values 3, 5 and 10 are used as the minimum rating count threshold. As a result, two rankings of users are obtained with respect to implicit and explicit trust scores, respectively, and we measure how well the top k-percent elements match. In the experiments, 10 and 20 values are used as the k value. The results are given in Tables 3–5 on the data sets FilmTrust, Epinions and Ciao, respectively.

Table 3. Implicit vs. Explicit trust model comparison results (FilmTrust).

	min. 3 Ratings	**min. 5 Ratings**	**min. 10 Ratings**
Recall@10%	0.003	0.003	0.004
Precision@10%	0.025	0.025	0.038
Recall@20%	0.023	0.023	0.027
Precision@20%	0.106	0.106	0.125

Table 4. Implicit vs. Explicit trust model comparison results (Epinions).

	min. 3 Ratings	**min. 5 Ratings**	**min. 10 Ratings**
Recall@10%	0.000	0.000	0.000
Precision@10%	0.002	0.004	0.020
Recall@20%	0.000	0.000	0.001
Precision@20%	0.006	0.015	0.041

Table 5. Implicit vs. Explicit trust model comparison results (Ciao).

	min. 3 Ratings	**min. 5 Ratings**	**min. 10 Ratings**
Recall@10%	0.006	0.006	0.008
Precision@10%	0.064	0.064	0.080
Recall@20%	0.035	0.035	0.042
Precision@20%	0.173	0.173	0.214

The same comparison is applied between the implicit trust scores and PageRank scores on the explicit trust network. Similarly, the results are given in Tables 6–8 on the data sets FilmTrust, Epinions and Ciao, respectively.

Table 6. Implicit vs. PageRank model comparison results (FilmTrust).

	min. 3 Ratings	**min. 5 Ratings**	**min. 10 Ratings**
Recall@10%	0.003	0.005	0.006
Precision@10%	0.038	0.50	0.062
Recall@20%	0.025	0.024	0.027
Precision@20%	0.138	0.131	0.150

Table 7. Implicit vs. PageRank model comparison results (Epinions).

	min. 3 Ratings	**min. 5 Ratings**	**min. 10 Ratings**
Recall@10%	0.000	0.000	0.001
Precision@10%	0.004	0.008	0.022
Recall@20%	0.000	0.000	0.001
Precision@20%	0.009	0.019	0.042

Table 8. Implicit vs. PageRank model comparison results (Ciao).

	min. 3 Ratings	**min. 5 Ratings**	**min. 10 Ratings**
Recall@10%	0.006	0.006	0.007
Precision@10%	0.057	0.057	0.070
Recall@20%	0.035	0.035	0.040
Precision@20%	0.174	0.174	0.203

In the analysis, precision and recall metrics are used for measuring the overlap between the top elements of implicit and explicit trust score rankings, rather than any prediction accuracy. In the measurement, the ranking of the explicit trust model is considered as the basis and the results of implicit trust score results are compared against them to obtain precision and recall values. As given in the tables, the low precision and recall scores show that the matching between two rankings is very scarce. However, as the implicit modeling is performed among more active users, it is seen that the precision and recall scores in the amount of match between implicit and explicit scores slightly increase. For the results of the proposed matching schema and the PageRank-based scoring, there are slight differences in the matching scores; however, overall, both show very similar behavior. The decrease, particularly in the recall value in the PageRank-based scoring, could be due to the incompatibility between the network propagation nature of the PageRank algorithm and the trust propagation behavior.

Overall, the low precision and recall scores show that explicit and implicit trust models rank the users differently and hence they model different aspects of the trust relationship. In [5], it is reported that the combination of explicit and implicit trust models increases the accuracy of estimates compared to using them separately. In the same study, it is also noted that the explicit trust relationship is also related to the social ties between users, so this cannot be entirely determined only by ratings given by users. The results of our analyses are compatible with the findings given in [5].

4.4. Explicit Trust Modeling Results

4.4.1. Explicit Trust Modeling through User's Rating Behavior Results

Before creating the features to model the explicit trust data implicitly, data exploration is performed to observe which features are needed to be created. Firstly, a signed trust network is used to elaborate on the concepts of liked and disliked items, since there is no distrust in the unsigned data set, the contrast here cannot be fully seen in the data. When we examine Table 9, the values in positive and negative trust relationships are calculated separately for each feature to be created.

Table 9. The Mean of the Generated Features (Signed Epinions).

	Positive Trust	**Negative Trust**
# of mutually rated items	98.807	61.179
# of mutually liked items	83.241	49.783
# of mutually disliked items	5.607	3.819
avg of the ratings given by the trustor	4.637	4.417
# of the ratings given by the trustor	1310.380	6482.471
avg of the ratings given by the trustee	4.605	4.367
# of the ratings given by the trustee	5947.152	2434.528

In this exploratory analysis, significant findings of users in positive trust and negative trust relationships can be obtained. Users in a positive trust relationship rate more common items on average than those in a negative relationship. In addition, the positive trust relationship correlates with the number of common favorite items. Although the negative trust items appear to be fewer than the positive trust items in the number of common disliked items, after normalizing the value, the number of common disliked items is also correlated with a negative trust relationship.

In addition, when Table 9 is checked, it is observed that the total number of ratings given by users who give negative trust is higher than the total number of ratings given by users who give positive trust. It can be interpreted that more active users, who give higher ratings, are also more selective and evaluate other users accordingly.

Similar data exploration is also applied to the unsigned data sets. The summary of the analysis is shown in Table 10. According to the results, the number of commonly liked items in trust relationships established in all three unsigned trust data sets is higher than that of commonly disliked items. It shows that liked items and established trust relationships correlate in unsigned trust data sets as well as in signed trust data sets.

Table 10. The Mean of the Generated Features (Unsigned Data Sets).

	FilmTrust	**Epinions**	**Ciao**
# of mutually rated items	9.079	1.194	2.018
# of mutually liked items	3.287	0.481	0.761
# of mutually disliked items	1.818	0.292	0.397
avg of the ratings given by the trustor	3.033	4.064	4.174
# of the ratings given by the trustor	39.865	69.332	150.835
avg of the ratings given by the trustee	3.041	4.015	4.209
# of the ratings given by the trustee	38.670	108.083	83.771

The performance of the unsupervised model generated by the unsigned data set is presented in Table 11. According to the results, the Isolation Forest model classifies most of the randomly generated trust instances to be in the regular class, which does not reflect the outlier (edge label 1) ratio in the data set. On the other hand, it is shown that the one-class SVM model labels nearly 50% of the Epinions and Ciao data sets as outliers. Here, too, the prediction accuracy is limited. Only FilmTrust data shows an outlier ratio close to expected. This difference in the result could be due to the differences in the nature of the data sets. As an example, Table 10 shows that the number of mutually rated items by users in the FilmTrust data set is significantly higher. Considering that the FilmTrust data set is smaller than the others, the unsupervised model can construct a model separating the outlier from regular cases better.

Table 11. Outlier Ratio of the Unsigned Data Sets.

	Isolation Forest	**One-Class SVM**
FilmTrust	0.134	0.883
Epinions	0.098	0.512
Ciao	0.111	0.538

The outlier/novelty detection methods mentioned above are also applied to Epinions data, a signed trust network. For both the Isolation Forest classifier and One-class SVM classifier, predictions are obtained with the following models:

- The model trained with positive trust data and tested with positive instances;
- The model trained with positive trust data and tested with negative instances;
- The model trained with negative trust data and tested with negative instances;
- The model trained with negative trust data and tested with positive instances.

The aim of this analysis is to see how much positive and negative trust relationships differ using the created features and outlier/novelty detection methods or whether positive and negative trust relationships can be modeled consistently within themselves. Table 12 shows the results of the experiment. Based on the results here, one can say that the One-class SVM model does not perform well in distinguishing negative and positive trust relationships. On the other hand, the Isolation Forest model successfully models both positive and negative trust data within itself. However, it does not perform the same success level in distinguishing positive and negative trust. One reason could be that the created features may not be fitting for the outlier/novelty detection method. In Table 9, it is observed that although there are points where positive and negative trust differ, the users

who have established these two relationships are also active users who have interacted with each other.

Table 12. Outlier Ratio of the Signed Epinions Data Set.

	Isolation Forest	**One-Class SVM**
Trained with positive, tested with positive	0.076	0.495
Trained with negative, tested with negative	0.077	0.501
Trained with positive, tested with negative	0.267	0.679
Trained with negative, tested with positive	0.114	0.526

Besides unsupervised outlier/novelty detection methods, Signed Epinions data is also modeled using multi-class classification algorithms. SVM, Random Forest, and Naive Bayes classifiers are used for model construction. The data is partitioned as training and test subsets with a ratio of 0.8 and 0.2. Table 13 shows the prediction performances of the models. When these results are examined, multi-class classification methods give more successful results than outlier/novelty detection methods. The reason is that both negative and positive trust relationships are used in the training phase. In addition, it can be seen that the Random Forest method gives the best results among these three classification methods. This can be considered an expected result since the Random Forest is a boosting-based method and it is reported to give successful prediction performance for a variety of domains.

Table 13. The Performance of Supervised Learning Models (Signed Epinions).

	Precision	**Recall**	**F1-Score**
SVM	0.780	0.743	0.747
Random Forest	0.873	0.865	0.868
Naive Bayes	0.744	0.723	0.726

In the next analysis, a trust network augmented with a multi-class classification model's predictions is used within trust-aware recommendation. The trust-aware recommendation algorithms used in the experiments are as follows:

- SBPR [31] is a ranking-based model that exploits social connections between users to build better prediction models. The model is based on the idea that users tend to give higher rankings to items that their connections prefer.
- SREE [32] is a social recommendation approach based on Euclidean Space. The idea behind this algorithm is to place users and items in a unified Euclidean space where users are close to both the items they want and their social friends.
- TBPR [33] classifies strong and weak ties in a social network and learns latent feature vectors for all users and items. It is an extension of the Bayesian Personalized Ranking model.

In the experiments, users who rate at least one mutual item are selected while choosing new trust relationships to be predicted. The results are given in Table 14. Precision and recall values are calculated by considering the top-10 item rankings in each recommendation algorithm. Judging by the results, performance gains have been observed in almost every case where augmented trust data is used. It can be said that the trust inference method is effective for improving recommendation accuracy.

Table 14. The effect of Modeled Explicit Trust Inference with SBPR, SREE and TBPR algorithms (Signed Epinions).

	w/o Trust Inference		With Trust Inference	
	Precision	**Recall**	**Precision**	**Recall**
SBPR [31]	0.005	0.016	0.009	0.027
SREE [32]	0.002	0.002	0.003	0.001
TBPR [33]	0.001	0.004	0.002	0.008

4.4.2. Trust Prediction Modeling as a Link Prediction Problem Results

In explicit trust modeling analysis, the basic idea is to construct a trust prediction model and to reduce data sparsity by filling in the trust matrix by using the predictions of the explicit trust model. In other words, a prediction is generated for the edge weights, which are 0 in the original network. The accuracy performances of the models generated with Random Forest and SVM classifiers for explicit trust prediction are given in Tables 15 and 16, respectively.

According to the results, the proposed explicit trust models can predict trust classes at a satisfactory rate, and an augmented matrix can be created effectively by inferring unknown trust links between users with the proposed modeling technique. It is also observed that the highest prediction accuracy is obtained on the Ciao data set, whereas the performance of the prediction on the unsigned Epinions data set is better than those on FilmTrust. FilmTrust is comparatively small in size, and hence the amount of trust information captured in the data is also comparatively limited. This possibly negatively affects the performance of the constructed prediction models. Moreover, according to the results, the Random Forest classifier performs better than the SVM method. For this reason, the output of the Random Forest model is used when creating the augmented trust network in the following experiment. Random Forest, being a boosting-based classifier, has been shown to be successful for a variety of prediction problems in the literature. Therefore, our observations are also in line with the literature in general.

Table 15. Accuracy results for explicit trust prediction (Random Forest).

Data Sets	**Accuracy**	**Precision**	**Recall**	**F1-Score**
FilmTrust	0.675	0.952	0.639	0.765
Epinions	0.930	0.979	0.879	0.926
Ciao	0.940	0.969	0.904	0.935

Table 16. Accuracy results for explicit trust prediction (SVM).

Data Sets	**Accuracy**	**Precision**	**Recall**	**F1-Score**
FilmTrust	0.819	0.891	0.728	0.801
Epinions	0.961	0.870	0.916	0.892
Ciao	0.901	0.955	0.843	0.895

To analyze the effect of explicit trust inference, the performance of the augmented trust matrix is compared against the original one by using a set of trust-based recommendation algorithms, SBPR, SREE, and TBPR on FilmTrust, unsigned Epinions, and Ciao data sets, given in Tables 17–19, respectively. The results indicate a minor increase in the recommendation performance with the inclusion of explicit trust inference. This result may be due to the fact that the trust values to be predicted do not have a significant change. Hence, the results hint at the possibility for improvement by carefully selecting the trust relationships to be predicted and updated.

Table 17. The effect of Explicit Trust Inference on Recommendation with SBPR, SREE and TBPR algorithms (FilmTrust).

	w/o Trust Inference		**With Trust Inference**	
	Precision	**Recall**	**Precision**	**Recall**
SBPR [31]	0.301	0.537	0.303	0.549
SREE [32]	0.310	0.402	0.306	0.397
TBPR [33]	0.294	0.472	0.287	0.471

Table 18. The effect of Explicit Trust Inference on Recommendation with SBPR, SREE and TBPR algorithms (Epinions).

	w/o Trust Inference		**With Trust Inference**	
	Precision	**Recall**	**Precision**	**Recall**
SBPR [31]	0.007	0.017	0.008	0.018
SREE [32]	0.007	0.013	0.007	0.013
TBPR [33]	0.001	0.003	0.002	0.004

Table 19. The effect of Explicit Trust Inference on Recommendation with SBPR, SREE and TBPR algorithms (Ciao).

	w/o Trust Inference		**With Trust Inference**	
	Precision	**Recall**	**Precision**	**Recall**
SBPR [31]	0.015	0.022	0.016	0.023
SREE [32]	0.004	0.003	0.005	0.004
TBPR [33]	0.003	0.004	0.004	0.005

5. Conclusions

In this work, trust modeling within the recommendation context is studied. More specifically, two sub-problems are focused on: (1) inferring the implicit trust information by examining the past user behaviors and analyzing the compatibility of implicit and explicit trust scores; (2) building an explicit trust model and predicting the missing explicit trust information.

For the first sub-problem, an implicit trust model is created. The implicit trust information is inferred by defining notions of conformity and consistency. After extracting implicit trust scores, the compatibility of implicit and explicit trust values is analyzed. The analysis of the approach reveals that there is no clear correlation between the implicit and explicit scores. The conducted experiments analyze how well the implicit and explicit scores match at the top-20% and top-10% of the trust scores. Under varying parameters, precision and recall scores are generally below 0.1. In addition, when the compatibility analysis is performed among more active users, it is seen that the precision increases above 0.1. The results hint at the effect of social ties in the trust relationship, and hence the implicit trust model cannot replace explicit trust but is merely helpful as complementary information.

For the second sub-problem, two different explicit models exhibit two different approaches. In the first approach, explicit trust is modeled by generating a set of new features containing liked and disliked items. While creating these features, users' rating behavior is used. Here, separate experiments are performed for signed and unsigned trust networks. Expected performance could not be achieved in models created with one-class classification. However, in the experiments conducted with the augmented trust data created with the multi-class classification model, the precision and recall values in the SBPR and TBPR

algorithms are boosted approximately twice. Here, it is seen that trust-based recommendation accuracy can be increased by modeling the ratings and explicit trust given by the users together.

For explicit trust score prediction, another solution is devised using the trust network itself. After generating the augmented trust network with this method, the effect of the augmented network is analyzed using various trust-based algorithms. The results show that the augmented trust matrix leads to improvement in performance, but the effect is not very high. This can be due to the fact that the trust values to be predicted are selected randomly, and the predictions do not significantly change the edge labels. Hence, with a more detailed mechanism for selecting the unknown trust relationships to be predicted, the performance could be further improved.

The proposed analysis on trust modeling for recommendation can be extended in a variety of directions. As one of the future dimensions, the proposed explicit trust model created by using rating behavior can be modified to be used for recommendation environments without any explicit trust data. Since the trust data is generally only scarcely available, such a solution widens the applicability of trust-based recommendation. In another future study, hybrid machine learning models and deep learning methods can be investigated to construct explicit trust models. The number of available data sets incurs a limitation, particularly for data-hungry deep learning-based solutions. At this point, mechanisms to incentive explicit trust in social network environments will be helpful to increase the amount of publicly available explicit trust data. Similar mechanisms have been employed in e-commerce platforms to express the reliability of e-stores. These mechanisms can be adapted and extended to social networks.

Another future work direction is conducting studies to detect and prevent attacks that can manipulate trust-based systems and affect users' trust scores. Additionally, generating different implicit trust models and elaborating on their compatibility with explicit trust scores can be further studied. Another interesting direction could be developing an implicit trust model that produces a distrust score as well as trust value.

Author Contributions: Conceptualization, U.D. and P.K.; methodology, U.D. and P.K.; software, U.D.; validation, U.D. and P.K.; writing, U.D. and P.K.; project administration, P.K.; funding acquisition, P.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by TUBITAK grant number 118E356.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sets used in the study are publicly available through the related references. For the analysis, Epinions (Unsigned) [27], Epinions (Signed) [28], FilmTrust [29], and Ciao [30] data sets are used.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Raza, S.; Ding, C. News recommender system: A review of recent progress, challenges, and opportunities. *arXiv* **2021**, arXiv:2009.04964.
2. Dhelim, S.; Ning, H.; Aung, N. ComPath: User interest mining in heterogeneous signed social networks for Internet of people. *IEEE Internet Things J.* **2020**, *8*, 7024–7035. [CrossRef]
3. Herlocker, J.L.; Konstan, J.A.; Terveen, L.G.; Riedl, J.T. Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. Inf. Syst.* **2004**, 5–53. [CrossRef]
4. Chae, D.K.; Kim, J.; Chau, D.H.; Kim, S.W. AR-CF: Augmenting Virtual Users and Items in Collaborative Filtering for Addressing Cold-Start Problems. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Xi'an, China, 25–30 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1251–1260.

5. Hu, Z.; Xu, G.; Zheng, X.; Liu, J.; Li, Z.; Sheng, Q.Z.; Lian, W.; Xian, H. SSL-SVD: Semi-Supervised Learning–Based Sparse Trust Recommendation. *ACM Trans. Internet Technol.* **2020**, *20*, 1–20. [CrossRef]
6. Khan, J.; Lee, S. Implicit user trust modeling based on user attributes and behavior in online social networks. *IEEE Access* **2019**, *7*, 142826–142842. [CrossRef]
7. Jamali, M.; Ester, M. A Matrix Factorization Technique with Trust Propagation for Recommendation in Social Networks. In Proceedings of the Fourth ACM Conference on Recommender Systems (RecSys'10), Barcelona, Spain, 26–30 September 2010; Association for Computing Machinery: New York, NY, USA, 2010;pp. 135–142.
8. Zhang, C.; Yu, L.; Wang, Y.; Shah, C.; Zhang, X. Collaborative User Network Embedding for Social Recommender Systems. In Proceedings of the 2017 SIAM International Conference on Data Mining (SDM), Houston, TX, USA, 27–29 April 2017; pp. 381–389.
9. Li, Y.; Liu, J.; Ren, J.; Chang, Y. A Novel Implicit Trust Recommendation Approach for Rating Prediction. *IEEE Access* **2020**, *8*, 98305–98315. [CrossRef]
10. Guo, G.; Zhang, J.; Thalmann, D.; Yorke-Smith, N. ETAF: An extended trust antecedents framework for trust prediction. In Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), Beijing, China, 17–20 August 2014; pp. 540–547.
11. Htun, Z.; Tar, P.P. A Trust-aware Recommender System Based on Implicit Trust Extraction. *Int. J. Innov. Eng. Technol. (IJIET) Technol. (IJIET)* **2013**, *2*, 271–276.
12. Yang, B.; Lei, Y.; Liu, J.; Li, W. Social Collaborative Filtering by Trust. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1633–1647. [CrossRef] [PubMed]
13. Guo, G.; Zhang, J.; Zhu, F.; Wang, X. Factored similarity models with social trust for top-N item recommendation. *Knowl.-Based Syst.* **2017**, *122*, 17–25. [CrossRef]
14. Demirci, M.U.; Karagoz, P. Trust Modeling in Recommendation: Explicit and Implicit Trust Model Compatibility and Explicit Trust Prediction. In Proceedings of the 13th International Conference on Management of Digital EcoSystems, Virtual Event, Tunisia, 1–3 November 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 8–14.
15. Oh, H.K.; Kim, S.W. Identifying and Exploiting Trustable Users with Robust Features in Online Rating Systems. *TIIS* **2017**, *11*, 2171–2195.
16. Chen, C.C.; Wan, Y.H.; Chung, M.C.; Sun, Y.C. An effective recommendation method for cold start new users using trust and distrust networks. *Inf. Sci.* **2013**, *224*, 19–36. [CrossRef]
17. Guo, G.; Zhang, J.; Thalmann, D. Merging trust in collaborative filtering to alleviate data sparsity and cold start. *Knowl.-Based Syst.* **2014**, *57*, 57–68. [CrossRef]
18. Wang, F.; Zhong, W.; Xu, X.; Rafique, W.; Zhou, Z.; Qi, L. Privacy-aware Cold-Start Recommendation based on Collaborative Filtering and Enhanced Trust. In Proceedings of the 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), Sydney, Australia, 6–9 October 2020; pp. 655–662.
19. Ahmadian, M.; Ahmadi, M.; Ahmadian, S. A reliable deep representation learning to improve trust-aware recommendation systems. *Expert Syst. Appl.* **2022**, *197*, 116697. [CrossRef]
20. Zahir, A.; Yuan, Y.; Moniz, K. AgreeRelTrust—A Simple Implicit Trust Inference Model for Memory-Based Collaborative Filtering Recommendation Systems. *Electronics* **2019**, *8*, 427. [CrossRef]
21. Canturk, D.; Karagoz, P. SgWalk: Location Recommendation by User Subgraph-Based Graph Embedding. *IEEE Access* **2021**, *9*, 134858–134873. [CrossRef]
22. Brin, S.; Page, L. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.* **1998**, *30*, 107–117. [CrossRef]
23. Lee, W.; Lee, Y.C.; Lee, D.; Kim, S.W. Look Before You Leap: Confirming Edge Signs in Random Walk with Restart for Personalized Node Ranking in Signed Networks. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, 11–15 July 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 143–152.
24. Adamic, L.A.; Adar, E. Friends and neighbors on the web. *Soc. Netw.* **2003**, *25*, 211–230. [CrossRef]
25. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
26. Yu, J.; Gao, M.; Yin, H.; Li, J.; Gao, C.; Wang, Q. Generating reliable friends via adversarial training to improve social recommendation. In Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), Beijing, China, 8–11 November 2019; pp. 768–777.
27. Richardson, M.; Agrawal, R.; Domingos, P. Trust management for the semantic web. In Proceedings of the International Semantic Web Conference, Sanibel Island, FL, USA, 20–23 October 2003; pp. 351–368.
28. Hamedani, M.R.; Ali, I.; Hong, J.; Kim, S.W. TrustRec: An effective approach to exploit implicit trust and distrust relationships along with explicitones for accurate recommendations. *Comput. Sci. Inf. Syst.* **2021**, *18*, 93–114. [CrossRef]
29. Golbeck, J.; Hendler, J. Filmtrust: Movie recommendations using trust in web-based social networks. In Proceedings of the IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA, 8–10 January 2006; Volume 96, pp. 282–286.
30. Tang, J.; Gao, H.; Liu, H.; Sarma, A.D. eTrust: Understanding Trust Evolution in an Online World. In Proceedings of the Eighteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012.

31. Zhao, T.; McAuley, J.; King, I. Leveraging social connections to improve personalized ranking for collaborative filtering. In Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, Shanghai, China, 3–7 November 2014; pp. 261–270.
32. Li, W.; Gao, M.; Rong, W.; Wen, J.; Xiong, Q.; Jia, R.; Dou, T. Social recommendation using Euclidean embedding. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 589–595.
33. Wang, X.; Lu, W.; Ester, M.; Wang, C.; Chen, C. Social recommendation with strong and weak ties. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, Indianapolis, IN, USA, 24–28 October 2016; pp. 5–14.