Knowledge-Based Systems 282 (2023) 111119

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Metapath-guided multi-headed attention networks for trust prediction in heterogeneous social networks☆

Yanwei Xu [a,b], Zhiyong Feng [a], Meng Xing [a], Hongyue Wu [a,*], Shizhan Chen [a], Xiao Xue [a], Schahram Dustdar [b]

[a] College of Intelligence and Computing, Tianjin University, Tianjin, 300350, China
[b] Distributed Systems Group, TU Wien, Vienna, 1040, Austria

ARTICLE INFO

Keywords:
Metapath
LSTM network
Multi-headed attention
Trust evaluation

ABSTRACT

Trust prediction facilitates the day-to-day functionality of diverse web-based applications, such as recommendation systems, market advertising and anomaly detection. However, existing works heavily rely on user–user trust interactions, which result in limited performance as the data sparsity. Previous studies have shown that the trust relationship between users is significantly affected by the category of items that the users interacted. In this paper, we propose a MetaTrust model, which generates redundant user-item interactions as the supplement of user–user trust to alleviate the data sparsity on trust prediction. Specifically, we propose category-aware metapaths, which generate abundant user–item–user interactions based on the common item category that users have interacted with. Further, Long Short Term Memory (LSTM) networks are utilized to mine features of multiple category-aware metapaths and their correlations. In order to filter the user–item–user interactions that are not related to the current task, the real trust relationship between users are embedd in the network with MLP. Finally, a multi-headed attention network is utilized to distinguish which metapath determines trust prediction between the current pair of users. Extensive experiments on three real-world dataset show that our proposed model can effectively achieve significant improvements over other competitive approaches and show the potential interpretability of trust building.

1. Introduction

Trust prediction facilitates the day-to-day functionality of diverse web-based applications, such as recommendation systems, market advertising, and anomaly detection. For instance, evidence suggests that users on Epinions, a service recommendation site, are more likely to seek advice from a trusted partner before purchasing the service [1]. The goal of trust prediction is to determine whether two partners share a trust relationship based on prior behavioral interactions, thereby reducing the risk associated with unpredictable future behavior of both partners [2]. To help us overcome these perceptions of risk and uneasiness, it becomes imperative to consider who and why we can trust.

Most of the existing work on trust prediction mainly focuses on extracting users' trust properties from trust networks (e.g., trust interactions between users) [3–6] and users' behavioral preferences from historical interaction records (e.g., ratings of item by users). By assuming the independence of these two types of interaction information from each other, earlier work has typically tackled trust interaction information and behavioral interaction information separately. Trust property-based approaches like STNE [3], OpinionWalk [4], and NeuralWalk [5] take use of propagation mechanisms to predict trust relationships among users. However, due to the sparsity of user–user trust interactions, such methods fail to predict the trust relationships of those user pairs without propagation paths. Behavioral preference-based methods [7,8], factorize a large amount of user–item interaction information into two low-rank matrices by matrix decomposition that yields a latent representation of the user. Since the dimensions of latent factor is determined by manual experience, these methods are prone to produce inaccurate user representations that result in overfitting.

☆ This work is supported by the National Natural Science Foundation of China grant No. 61832014 and No. 61972276, the Shenzhen Science and Technology Foundation, China grant No. JCYJ20170816093943197, and the Natural Science Foundation of Tianjin City, China grant No. 19JCQNJC00200.

* Corresponding author.
E-mail addresses: xuyanwei@tju.edu.cn (Y. Xu), zyfeng@tju.edu.cn (Z. Feng), xingmeng@tju.edu.cn (M. Xing), hongyue.wu@tju.edu.cn (H. Wu), shizhan@tju.edu.cn (S. Chen), jzxuexiao@tju.edu.cn (X. Xue), dustdar@dsg.tuwien.ac.at (S. Dustdar).

https://doi.org/10.1016/j.knosys.2023.111119
Received 14 February 2023; Received in revised form 10 August 2023; Accepted 23 October 2023
Available online 29 October 2023

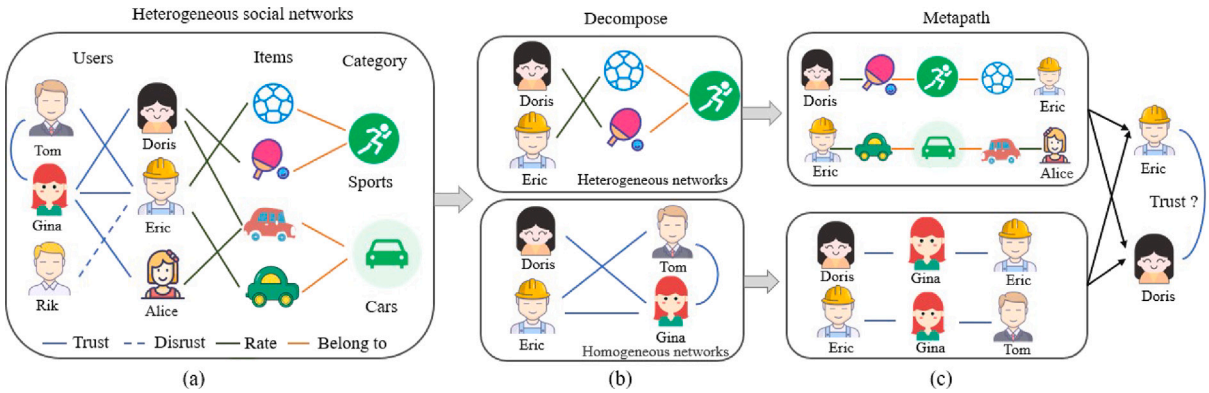

Fig. 1. Illustrative example of a heterogeneous network.

As shown in Fig. 1(a)(b), there are various types of nodes in a heterogeneous social network (HSN), including users, items, and categories. Users have distinct trust relationships with one another, and items that users interact with belong to the same category or different. Based on the interaction types between nodes, heterogeneous social networks can be decomposed into homogeneous networks (e.g., trust networks) and heterogeneous networks; the former solely contains trust interactions between users, while the latter contains behavioral interactions between users and items. In homogeneous networks, due to the sparsity of the trust relationship, there is no sufficient propagation path to determine the trust relationship between Doris and Eric. In contrast, there are a large number of user–item interactions in heterogeneous network. Generally, items in the same category usually have similar functions and characteristics. As an example, Eric enjoys playing soccer, whereas Doris enjoys playing table tennis. Table tennis and soccer belong to the same category while indicating that Doris and Eric have similar behavioral preferences for sports. Furthermore according to the homophily theory, interpersonal trust is encouraged by such similar behavioral preferences based on shared category items [9]. Therefore, adequate user–item behavioral interactions can serve as complementary information to sparse user–user trust interactions that jointly contribute to trust prediction [10,11].

However, it is a challenging task to fuse trust and behavioral interaction information in HSN for learning an effective user representation as following three critical issues. (1) Originating from the sparsity of trust interactions (e.g., user–user), how to incorporate sufficient user–item interaction information including ratings, reviews and categories of items. These heterogeneous data reveal users' behavioral preferences, but it is a hard task to represent and fuse these high-dimensional and multi-category data. (2) Due to the heterogeneity of HSN structures, it is a challenge to preserve the structural features of user–user and user–item interactions and their intrinsic correlations. (3) How to fuse the semantic features of different structures to obtain informative user representations that can then be utilized to infer the strength of trust between users.

Recently, considering that metapaths can connect different types of nodes, there has been significant interest in developing metapath models for mining various node sequence features in well-known domains, such as link prediction and recommender systems. Inspired by the rich semantics of node sequences revealed through metapaths, we exploit metapaths to portray fine-grained interaction information in HSNs (as shown in Fig. 1(c)). To address the above challenges, we propose a deep metapath-guided trust prediction model in HSN. Specifically, we draw two metapaths based on behavioral preferences and trust properties, where the former characterizes users' multifaceted behavioral preferences, while the latter can characterize users' trust properties based on trust propagation. Together, the double semantics reflected by the two types of metapaths expose different facets of the user's characteristics. Then, embedding techniques are applied to map the two heterogeneous metapaths into the same feature space; Further, we employ Long Short-Term Memory (LSTM) networks and multilayer perceptron (MLP) networks to extract metapath features, respectively. Among them, the LSTM network is capable of parsing the semantics of behavioral interactions of pairwise users, while the MLP network is able to maximize the co-occurrence probability of neighboring nodes. Finally, multiple metapath features are delivered into the multi-headed attention network to capture the correlation between metapath features and provide a reliable basis for the trust relationship between any two users.

The main contributions of this paper are as follows.

- We formally propose a metapath-guided trust prediction approach based on the fruitful behavioral interaction records and trust interactions between users in heterogeneous social networks.
- To mine sufficient user–item interaction information, category-aware metapaths are constructed and transformed into the same space by embedding while preserving the original structure. The sparsity of trust relationships is addressed.
- To preserve the heterogeneity of HSN structure, category-aware metapath features and trust property-based metapath features are mined through LSTM and MLP networks. Further, multiple metapaths are fused through a multi-headed attention network to obtain informative user representations for trust prediction.
- This is the first interaction-based metapath-guided trust prediction model. The scalable experiments validated on three datasets demonstrate that the proposed approach outperforms other competitive baseline approaches with extraordinary excellence. Moreover, it can work well even with sparse or no trust interaction information.

The remaining part of this paper is organized as follows. Section 2 describes the related work; Section 3 presents the specific steps of the proposed model; Section 4 shows the practical experimental results; Section 5 summarizes the whole work and points out the direction of future research.

2. Related work

Prior works on trust prediction can be grouped into two main categories: (1) trust prediction in homogeneous social networks; (2) trust prediction in heterogeneous social networks.

2.1. Trust prediction in homogeneous social networks

The trust prediction approaches based on homogeneous social networks mainly assume the existence of different trust strengths between two users. By exploiting the fact that propagation is one of the properties of trust, trust propagation mechanism is utilized to calculate the

Table 1
Literature summary.

Category	Approaches	Trust propagation	Context-aware information	Behavior information
Trust prediction in homogeneous social networks	Web of trust [12]	✓	×	×
	NeuralWalk [5]	✓	×	×
	OpinionWalk [4]	✓	×	×
	Guardian [13]	✓	×	×
	Medley [14]	✓	×	×
Trust prediction in heterogeneous social networks	mTrust [15]	×	✓	×
	C-DeepTrust [16]	×	×	✓
	Context-aware trust [17]	×	✓	✓
	Ante-trust [18]	×	×	✓
	JMF [19]	×	✓	✓
	MemTrust [20]	×	×	✓

indirect trust relationships without relying on the previous behavioral interactions between two users.

Guha et al. [12] believes that social networking sites maintain a web of trust, while arguing that trust and distrust are equally important. The trust and distrust values between users are added to the atomic propagation and cocitation operations to infer trust relationships between indirect users. To further identify the factors affecting trust propagation, the approach proposed in [5] to estimate the factors affecting the trust building relationship. First, WalkNet is employed to assess single-hop trust values by learning the network parameters, and Neuralwalk is then able to iteratively deduce indirect multi-hop trust relationships. Furthermore, the trust values between users are propagated through discounting and combining operations to obtain more accurate results [4,21,22].

Different from the above manually set rules or complex trust fusion operations in trust propagation, Lin et al. propose an end-to-end framework that divides the associated trust relationships between users into popularity trust and engagement trust. The corresponding propagation mechanisms and social network structure are captured by exploiting graph convolutional networks [13]. Furthermore, in order to preserve the structural proximity in social networks, social trust network embedding [3] are proposed, and structural balance theory is utilized to infer the potential relationships between multi-hop users. Given the presence of both positive and negative interactions in social networks, a signed social network (SSN) is constructed. Subsequently, diffTrbML is proposed to model SSNs and predict diverse trust values between users [23]. Lin et al. [14] contend that it is important to predict social trust from dynamic social interaction. To explicitly capture time-varying latent factors, attention mechanisms are exploited to assign greater weight to recent social interactions. By fusing evolving topological networks, social trust is predicted dynamically over time.

Since the explicit trust values between users are very sparse, the above mentioned methods mainly speculate the trust relationship between indirect two users through trust propagation and trust transfer patterns [24,25]. These methods face two drawbacks: (1) There is no unified trust propagation mechanism, which lacks certain universality due to the artificially established trust propagation rules; (2) For two users without any common trust neighbor, the trust relationship cannot be inferred.

2.2. Trust prediction in heterogeneous social networks

Due to the complexity of trust formation, The various node types and relationships in heterogeneous social trust networks are exploited for trust prediction. The types of nodes can be categorized as user and item, and there are multiple kinds of relationships between them (e.g., rating, review, helpful-rating) [26–28]. Such approaches can be mainly classified into context-aware approaches and historical behavior-based approaches [29–31].

The fact that people have multifaceted interests and have different aspects of expertise, shows that users have varying levels of trust in various social group. Tang et al. [15] argue that the difference in the degree of expertise of users demonstrates the existence of heterogeneous trust relationships among users. And the strength of multifaceted trust relationship is explored by applying matrix decomposition. Trust can be built virtually based on contextual information, which can effectively reflect the interaction context between users, e.g., time and geographical location. The dynamic and static preferences of the users are obtained in each context, and finally context-aware features are fused and fed into the MLP network to obtain the final context-aware latent feature for trust evaluation [16]. Considering that rich information within a context also affects trust estimation, Zheng et al. [17] add personal/interpersonal properties into the trust transfer model and use matrix decomposition to compute trust relationships in the target context.

Social science theory suggests that users with similar behaviors and preferences are more likely to build trust relationships [32,33]. Mining user behavior characteristics from users' historical behavior records is helpful for trust building. Huang et al. [19] modeled user–user trust graph and user behavior auxiliary graph as heterogeneous social networks. The joint manifold factorization (JMF) method is proposed to share the common structure and patterns of the two graphs. The group-level trust are predicted. Negi et al. [34] argue that there are multiple link types in heterogeneous social networks and propose the corresponding distance metric for each link type. Wang et al. [18] believe that users' attribute information has an impact on the establishment of trust relationships, and treat users' rating/review records of items as users' attribute information. Xu et al. [20] argue that the dynamic behavioral preferences of users can provide strong evidence for trust relationship establishment. The LSTM network is employed to extract the behavioral features of users in multiple time periods, and then the trust relationship between any two users is calculated by latent feature.

In Table 1, we present a detailed analysis of the strengths and weaknesses of prior research, highlighting their respective contributions to trust prediction in online social networks. Our proposed metapath-guided trust prediction method based on heterogeneous social networks is different from previous work. (1) This work adopts a metapath-based approach to preserve the order of interactions between users as well as the sequence semantics, which is different from previous work that utilizes all the interaction records without sequential nature. (2) By leveraging the trust property-based metapath, we can effectively aggregate trust neighbor information. (3) This work is based on metapath-guided trust evidence search, which can effectively expand the trust-oriented information to find potential trust relationship among users.

3. Problem definition

In this section, we will describe some of the concepts used in the paper.

Definition 1 (*Trust Prediction*)**.** Give a set $\langle \mathcal{U}, \mathcal{I}, \mathcal{A}, \mathcal{B} \rangle$, where $\mathcal{U}$ indicates the set of users, $\mathcal{I}$ indicates the set of items. $\mathcal{A}$ indicates the attributes linked with the object. $\mathcal{B}$ indicates the behavioral interactions

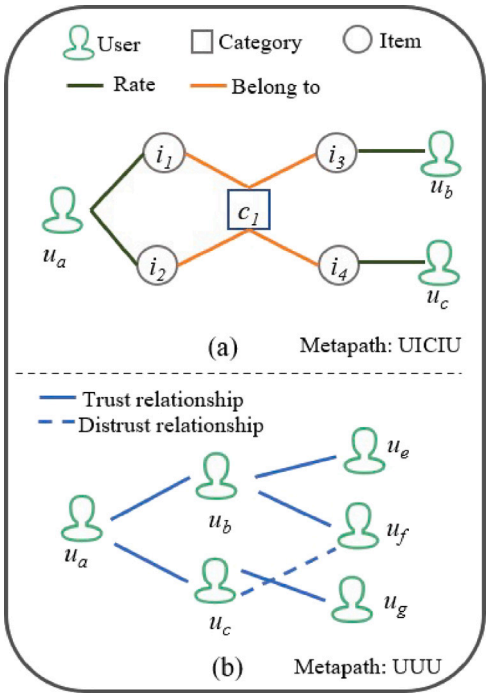

Fig. 2. Illustrative example of metapaths in HSN.

between different types of objects. In our task, trust prediction aims to estimate the level of trust and distrust between any two users by relying on user–user and user–item interaction behavior. We model trust prediction in the context of HSN [34,35]. A HSN can defined as a graph G = (V, E), where contain multiple objects and links. In HSN, network schema is designed to characterize the meta-structure of a network, which can show object types and their interaction behavior. Metapaths [36] are sequences of objects that capture the structure and semantic relations across multiple objects.

In Fig. 2, we can observe that the HSN contains multiple types of objects (eg., Users(U), Items(I), Category(C)) and abundant interactions relations between objects. Since our task is to predict the trust relationship between an individual user and other users, we focus specifically on metapaths from users to show the interaction behavior characteristics and trust properties of users. As shown in Fig. 2 (a), the category-aware metapath "User-Item-Category-Item-User"($UICIU$), indicates that two users have interacted with two items, which belong to the same category, respectively. For instance, u_a rates i_1 and i_2, u_b rates i_3, whereas i_1, i_2 and i_3 all belong to category c_1. The intuitive presentation of semantic relationships as: $User \xrightarrow{rate} Item \xrightarrow{belong\ to} Category \xleftarrow{belong\ to} Item \xleftarrow{rate} User$. In Fig. 2 (b), the trust property-based metapath "User-User-User"(UUU) indicates the trust relationship between adjacent users. For instance, "U_a–U_b–U_e" means that U_a trusts U_b and U_b trusts U_e. It can be represented intuitively as: $User \xrightarrow{trust} User \xrightarrow{trust} User$.

Definition 2 (*Metapath-guided Neighbors*)**.** Give a object a and a metapath $\mathcal{P}$. The metapath-guided neighbors are defined as the set of objects that traverse the entire metapath from object a. Furthermore, we define the i-hop neighbor of object a as $N_i^{\mathcal{P}}(a)$. Particularly, $N_0^{\mathcal{P}}(a)$ refers to object a

As shown in Fig. 2(a), for the metapath $UICIU$ from u_a, we can obtain the methpath-aware neighbors as $N_1^{UICIU}(u_a) = \{i_1, i_2\}$, $N_2^{UICIU}(u_a) = \{c_1\}$, $N_3^{UICIU}(u_a) = \{i_3, i_4\}$, $N_4^{UICIU}(u_a) = \{u_b, u_b\}$. All metapath neighbors starting with u_a are $N^{UICIU}(u_a) = \{N_1^{UICIU}(u_a), N_2^{UICIU}(u_a), N_3^{UICIU}(u_a), N_4^{UICIU}(u_a)\} = \{i_1, i_2, c_1, u_1, u_2\}$. As shown in Fig. 2(b), for the metapath UUU starting with u_a, the 1-hop and 2-hop metapath neighbors of u_a are $N_1^{UUU}(u_a) = \{u_b, u_c\}$, $N_2^{UUU}(u_a) = \{u_d, u_e, u_f\}$. All the metapath neighbors of u_a are $N^{UUU}(u_a) = \{N_1^{UUU}(u_a), N_2^{UUU}(u_a)\} = \{u_b, u_c, u_d, u_e, u_f\}$.

Existing attempts on HSN-based trust prediction usually utilize matrix decomposition to capture the behavioral characteristics of users for predicting trust relationships. Instead, we utilize a metapath sequence search way to discover trust-oriented interaction information. This manner is capable of reflecting the semantic and structural relations between objects and provides certain interpretability for trust relationship building.

4. The MetaTrust model

We propose a deep metapath-guided trust prediction model, named MetaTrust. The framework of the MetaTrust model is shown in Fig. 3. Specifically, (1) Metapath Embedding. Interaction-based metapaths are mapped into a unified feature space by embedding. In this way, category-aware metapaths preserve the semantics of behavioral interactions between users, while trust property-based metapaths enable the preservation of first-order second-order trust neighbors. (2) Metapath Feature Extraction: Different from the previous way of fusing metapaths, we employ LSTM networks to extract features from category-aware metapaths in parallel. Simultaneously, we exploit the multi-layer perception network to extract the features of users' multi-hop trust neighbors, which helps to find co-occurrence trust neighbors between users based on trust property-based metapaths. (3) Inter-metapath Fusion. Based on multiple parallel metapath features, we use a multi-headed attention network to automatically extract the comprehensive metapath-guided trust features. (4) Trust Evaluation. Trust values between pairs of users are evaluated by softmax function. We present the MetaTrust model thoroughly in the following subsections.

4.1. Metapath embedding

In previous HGN-based trust prediction models [37,38], behavioral preferences of users were modeled mainly by mining user–item rating interaction records through matrix decomposition [39,40]. These approaches have two main drawbacks. (1) It requires the completion of item ratings that users have not interacted with; then feature decomposition is performed, which is prone to overfitting and leads to inaccurate rating predictions. (2) With millions of items on the networking sites, such an approach is more computationally intensive than practical. With embedding's ability to maintain semantic relevance, we can only retain the items that each user has interacted with, and without needing to keep items that they have not.

In social networking sites, such as Epinions, there are three object types including users, items and categories. There are three types of interaction between objects, such as user–user interactions and user–item and item–item interactions. Specifically, user–user interactions involve the existence of a trust relationship between users, user–item interactions involve the items rated by users, and item–item interactions involve the item belonging to the same category or not.

The symbols that represent these objects and the records of their interactions are specified as below.

- $\mathcal{U} = \{u_1, \ldots, u_M\}$: denote the set of M users.
- $\mathcal{I} = \{i_1, \ldots, i_N\}$: denote the set of N items.
- $\mathcal{C} = \{c_1, \ldots, c_K\}$: denotes the set of K categories of items. Typically, an item belongs to only one category.
- $\mathcal{C}_{N*K}$ denotes a item-categeory matrix. If $c_{N*K} = 1$, it denotes that item N belongs to category K and otherwise, $c_{N*K} = 0$.
- $\mathcal{T}_{M*M}$ denotes the user–user trust relationships matrix. If $T_{a,b} = 1$, it indicates that there exists a trust relationship between pairwise users (u_a, u_b), and otherwise, $T_{a,b} = 0$.

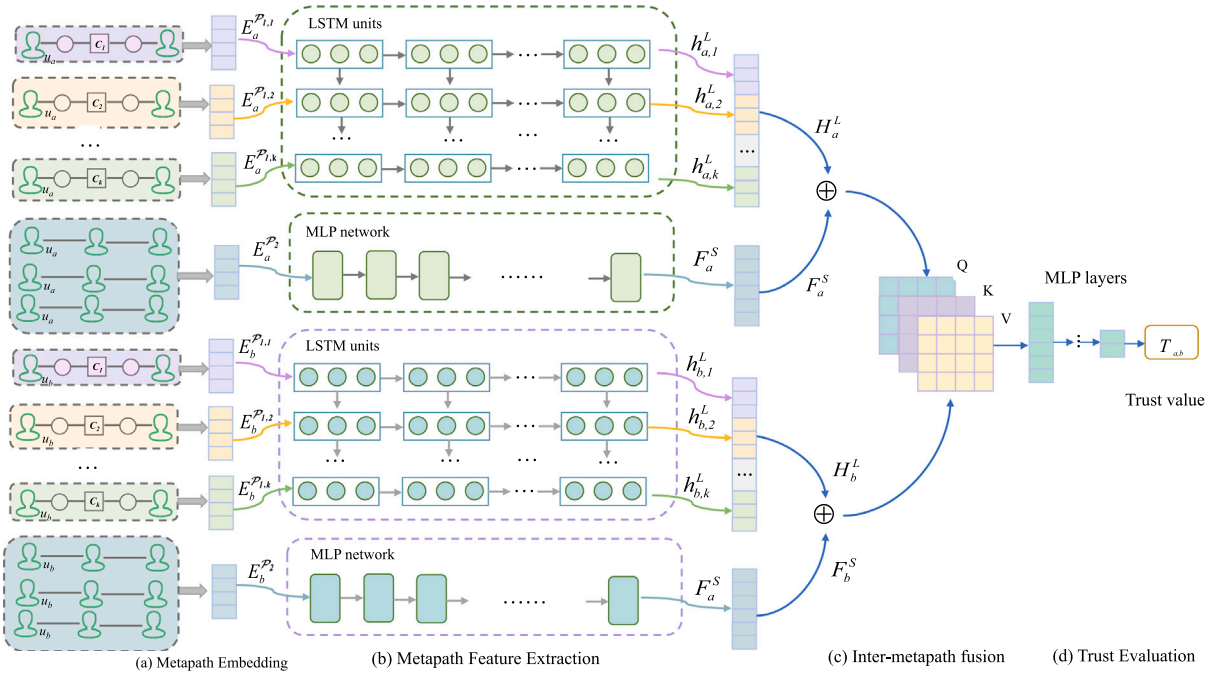

Fig. 3. The MetaTrust model.

Subsequently, we model the behavior information of each user.

- $\mathcal{I}_a^k = \{i_{a,1}^k, i_{a,2}^k, \ldots, i_{a,Q}^k, \}$: denotes the set of Q ($1 \le Q \le N$) items that u_a has interacted with in category c_k ($1 \le k \le K$).
- $\mathcal{R}_a^k = \{r_{a,1}^k, r_{a,2}^k, \ldots, r_{a,Q}^k, \}$: denotes the set of u_a' ratings for Q ($1 \le Q \le N$) items interacted with in category c_k ($1 \le k \le K$).
- $\mathcal{U}^k = \{u_1^k, u_2^k, \ldots, u_G^k\}$: denotes the set of G ($1 \le G \le M$) users who have interacted with items in category c_k ($1 \le k \le K$).

For each $u_g \in \mathcal{U}^k$, $\mathcal{I}_g^k = \{i_{g,1}^k, i_{g,2}^k, \ldots\}$ denotes the set of items that u_g has interacted with in category c_k. For each $u \in \mathcal{U}^k$, The set of items that all users have interacted with in category c_k is $\{\mathcal{I}_1^k, \mathcal{I}_2^k, \ldots, \mathcal{I}_G^k\}$. Since there are millions of items in the set $\{\mathcal{I}_1^k, \mathcal{I}_2^k, \ldots, \mathcal{I}_G^k\}$, we randomly choose $\mathcal{I}^k = \{i_1^k, i_2^k, \ldots, i_F^k\}$ from the set $\{\mathcal{I}_1^k, \ldots, \mathcal{I}_g^k, \ldots, \mathcal{I}_G^k\}$ by uniform distribution and the set of corresponding ratings can be denoted as $\mathcal{R}^k = \{i_1^k, i_2^k, \ldots, i_F^k\}$.

With fine-grained delineation of users' interaction information, metapaths can effectively link different types of interaction information between users and maintain the original semantic relationships. Here, for simplicity of presentation, the category-aware metapath *UICIU* is denoted as $\mathcal{P}_1$ and the trust property-based metapath *UUU* is denoted as $\mathcal{P}_2$. According to Definition 2, the 1-hop metapath-guided neighbor under metapath $\mathcal{P}_1$ for u_a is $N_1^{\mathcal{P}_1}(u_a) = \{\mathcal{I}_a^k\}$, the 2-hop, 3-hop and 4-hop metapath-guided neighbor for u_a are $N_2^{\mathcal{P}_1}(u_a) = \{c_k\}$, $N_3^{\mathcal{P}_1}(u_a) = \{\mathcal{I}^k\}$, and $N_4^{\mathcal{P}_1}(u_a) = \{\mathcal{U}^k\}$. All the metapath-guided neighbor are $N^{\mathcal{P}_1}(u_a) = \{N_1^{\mathcal{P}_1}(u_a), N_2^{\mathcal{P}_1}(u_a), N_3^{\mathcal{P}_1}(u_a), N_4^{\mathcal{P}_1}(u_a)\} = \{\mathcal{I}_a^k, c_k, \mathcal{I}^k, \mathcal{U}^k\}$.

To preserve the sequential and heterogeneous nature of the interaction information [41,42], embedding technique is adopted to project the metapath-guided neighbors in a unified feature space. As the 1-hop and 3-hop metapath-guided neighbours involve the items interacted by u_a and $\mathcal{U}^k$, the embedding of the ratings of these items are denoted as $E_a^{R,k} \in \mathbb{R}^{Q*d_e}$ and $E_u^{R,k} \in \mathbb{R}^{F*d_e}$, respectively. Thus, the 1-hop, 2-hop, 3-hop, and 4-hop metapath-guided neighbor embedding matrix for u_a are $E_a^{I,k} \in \mathbb{R}^{Q*d_e}$, $E_a^{c_k} \in \mathbb{R}^{1*d_e}$, $E_u^{I,k} \in \mathbb{R}^{F*d_e}$, $E_u^{U,k} \in \mathbb{R}^{G*d_e}$, respectively. The sequence semantics based on the metapath $\mathcal{P}_1$ are preserved by Eq. (1).

$$E_a^{\mathcal{P}_{1,k}} = E_a^{I,k} \oplus E_a^{R,k} \oplus E_a^{c_k} \oplus E_u^{I,k} \oplus E_u^{R,k} \oplus E_u^{U,k} \tag{1}$$

where $\oplus$ denotes the concatenation operation between two matrices. To capture the multifaceted nature of user behavioral preferences, category-centric paths were designed based on $\mathcal{P}_1$. $E_a^{\mathcal{P}_{1,k}}$ denotes the semantic features of u_a based on category $c_k (1 \le k \le K)$ in metapath $\mathcal{P}_1$. Thus, the set of path by metapath $\mathcal{P}_1$ can be represented as $\{E_a^{\mathcal{P}_{1,1}}, E_a^{\mathcal{P}_{1,2}}, \ldots, E_a^{\mathcal{P}_{1,k}}, \ldots, E_a^{\mathcal{P}_{1,K}}\}$.

For the category-aware metapaths, we designed these metapaths to capture the interaction patterns between users and items of the same category. These metapaths are constructed by traversing the user–item–user interaction network while considering the category information associated with the items. The metapaths connect users who have interacted with items of the same category, thereby generating additional user–item–user interactions.

For the metapath $\mathcal{P}_2$, the set of 1-hop and 2-hop metapath-guided neighbors by u_a are denoted as $N_1^{\mathcal{P}_{2,1}}(u_a) = \{u_1, u_2, \ldots\}$ and $N_2^{\mathcal{P}_{2,2}}(u_a) = \{u_3, u_4, \ldots\}$, respectively. We search the trusted neighbors of u_a with width-first manner. The embedding matrices of 1-hop and 2-hop metapath-guided neighbors can be represented as $E_a^{\mathcal{P}_{2,1}}$ and $E_a^{\mathcal{P}_{2,2}}$. To effectively fuse the multi-hop meta-path-aware neighbors for u_a, the specific representation is shown in Eq. (2) (3).

$$E_a^{\mathcal{P}_{2,1}} = g(E_a^{\mathcal{P}_{2,2}}) \tag{2}$$

$$E_a^{\mathcal{P}_2} = g(E_a^{\mathcal{P}_{2,1}}) \tag{3}$$

where $g(\cdot)$ denotes the meanpooling function, which can weaken the influence of trust neighbors of 2-hops on trust prediction. $E_a^{\mathcal{P}_2}$ represents the semantic features of u_a in metapath $\mathcal{P}_2$.

Here, we construct two metapaths with three advantages: (1) category-aware metapath $\mathcal{P}_1$, which can effectively expand the range of behavioral interactions between two users, i.e., from interacting with the same items to interacting with items under the same category. (2) The trust property-based metapath $\mathcal{P}_2$ organizes users' trust neighbors, which assists in the extraction of behavior records of both users who trust each other in the category-aware metapath. (3) Different from the traditional approach of searching all interaction records of users, we uniformly select a small number of interaction records, which greatly reduces the complexity of computation.

4.2. Metapath feature extraction

With the ability of LSTM networks to handle parallel sequences and discern their correlation, we employ multi-layer LSTM networks to extract features from multiple parallel paths based on $\mathcal{P}_1$. The structure of each LSTM unit is as follows.

$$\begin{bmatrix} \tilde{\boldsymbol{c}}^l_{a,k} \\ \boldsymbol{i}^l_{a,k} \\ \boldsymbol{f}^l_{a,k} \\ \boldsymbol{o}^l_{a,k} \end{bmatrix} = \begin{bmatrix} \tan h \\ \sigma \\ \sigma \\ \sigma \end{bmatrix} \left(\boldsymbol{W} \begin{bmatrix} h^l_{a,k-1} \\ h^l_{a,k} \end{bmatrix} + \boldsymbol{b} \right) \tag{4}$$

$$\boldsymbol{c}^l_k = \boldsymbol{f}^l_{a,k} \odot \boldsymbol{c}^l_{a,k-1} + \boldsymbol{i}^l_{a,k} \odot \tilde{\boldsymbol{c}}^l_{a,k} \tag{5}$$

$$\boldsymbol{h}^l_{a,k} = o^l_{a,k} \odot \tanh(\boldsymbol{c}^l_k) \tag{6}$$

where l $(1 \le l \le L)$ denotes the horizontal depth of LSTM network. $\boldsymbol{h}^l_{a,k}$ denotes the hidden layer output at the horizontal depth l and vertical input of the kth path sequence of LSTM network by u_a. Similarly, $\boldsymbol{h}^l_{b,k}$ can be obtained. For u_a and u_b, the input matrices of LSTM networks are $[h^0_{a,1}, h^0_{a,2}, \ldots, h^0_{a,K}] = [E_a^{\mathcal{P}_{1,1}}, E_a^{\mathcal{P}_{1,2}}, \ldots, E_a^{\mathcal{P}_{1,K}}]$ and $[h^0_{b,1}, h^0_{b,2}, \ldots, h^0_{b,K}] = [E_b^{\mathcal{P}_{1,1}}, E_b^{\mathcal{P}_{1,2}}, \ldots, E_b^{\mathcal{P}_{1,K}}]$, when the horizontal depth of LSTM network is set as $l = 1$. After the multiple path sequences are processed by the multi-layer LSTM network, the integrative features of u_a and u_b based on the metapath $\mathcal{P}_1$ are obtained as shown below.

$$\begin{cases} H^L_a = (h^L_{a,1} + h^L_{a,2} + \ldots, + h^L_{a,K})/K \\ H^L_b = (h^L_{b,1} + h^L_{b,2} + \ldots, + h^L_{b,K})/K \end{cases} \tag{7}$$

where H^L_a and H^L_b are both metapath features of u_a and u_b by metapath $\mathcal{P}_1$ and the output of the average hidden vector at horizontal depth L of the LSTM network. $h^L_{a,K}$, $h^L_{b,K}$ represent the hidden outputs of the LSTM network with horizontal depth L and vertical input of the Kth metapath sequence by u_a and u_b, respectively. The structural and semantic information implied in the metapath embeddings are fed into the LSTM network in parallel. By this way, single path features are parsed and feature correlations between multiple paths are also captured. The user behavior-centric metapath $\mathcal{P}_1$ reflects the interaction behavior between users and reveals the user behavior preferences.

The metapath $\mathcal{P}_2$ represents the interaction of trust relationships between users, which characterizes the trust properties of users. Based on Eq. (3), we can get the embedding $E_a^{\mathcal{P}_2}$ and $E_b^{\mathcal{P}_2}$ for u_a and u_b, respectively. The MLP network are exploited to further extract the trust property-based metapath features for paired users. The specific equation is shown below.

$$\begin{cases} F^1 = \boldsymbol{w}^1_{mlp} x + \boldsymbol{b}^1_{mlp} \\ F^s = \boldsymbol{w}^s_{mlp} F^{s-1} + \boldsymbol{b}^s_{mlp} \quad (s = 2..., S) \\ F^S = \boldsymbol{w}^S_{mlp} F^{S-1} + \boldsymbol{b}^S_{mlp} \quad (s = 2..., S) \end{cases} \tag{8}$$

$$\begin{cases} F^S_a = (...(\boldsymbol{w}^2_{mlp}(\boldsymbol{w}^1_{mlp} E_a^{\mathcal{P}_2} + \boldsymbol{b}^1_{mlp}) + \boldsymbol{b}^2_{mlp})...) \\ F^S_b = (...(\boldsymbol{w}^2_{mlp}(\boldsymbol{w}^1_{mlp} E_b^{\mathcal{P}_2} + \boldsymbol{b}^1_{mlp}) + \boldsymbol{b}^2_{mlp})...) \end{cases} \tag{9}$$

where $\boldsymbol{w}^s_{mlp}$ and $\boldsymbol{b}^s_{mlp}$ denote the weight matrix and bias matrix in the S-layer MLP network, respectively. F^S denote the output of the MLP network at the Sth layer in Eq. (8). F^S_a and F^S_b represent the metapath features of u_a and u_b by the metapath $\mathcal{P}_2$ respectively, which are also the outputs of the MLP network at the Sth layer. By nonlinear feature transformation, pairwise features can be effectively obtained from the first-order and second-order trust neighbors of users, while maintaining the semantic sequential nature of the neighbors.

4.3. Inter-metapath fusion

In order to capture the behavioral characteristics and trust trends of users, the two types of metapath-guided features for pairwise users (u_a, u_b) are merged. The concatenation operation is taken as shown in Eq. (10).

$$\boldsymbol{T}_{glo} = H^L_a \oplus H^L_b \oplus F^S_a \oplus F^S_b \tag{10}$$

where $T_{glo} \in \mathbb{R}^{D_{glo} * N}$ is denoted as a global trust feature for u_a and u_b. Thereby, pairs of metapath features are linked together to represent the behavior features and trust features based on the interaction information of u_a and u_b as a whole.

Since global trust features T_{glo} are composed of four features by pairs of users, how can we investigate which aspects are essential for trust prediction? In order to effectively locate trust-oriented features, a multi-headed self-attention network [43] is utilized to automatically assign unequal weights to the four features. The operation mechanism of a multi-headed self-attention network can be defined by the following equations.

$$atten(Q, K, V) = V\,softmax(\frac{K^T Q}{\sqrt{D_d}}) \tag{11}$$

$$Q = W_q T_{glo}, K = W_k T_{glo}, V = W_v T_{glo}, \tag{12}$$

where Q, K and V denote query, key and value matrix for linear projection, respectively. $W_q \in \mathbb{R}^{D_d * D_{glo}}$, $W_k \in \mathbb{R}^{D_d * D_{glo}}$, and $W_v \in \mathbb{R}^{D_v * D_{glo}}$ are the input weight matrices of the linear projection.

After multiple linear mapping, the four features in T_{glo} are effectively assigned inequitable weights. For the sake of revealing the correlation between four features from different aspects, the multi-head attention network is utilized, which can capture variable correlation in multiple group projection spaces. The attention model is applied in O group projection spaces with the following equation.

$$\begin{cases} Q_i = W^i_q T_{glo}, K_i = W^i_k T_{glo}, Q_i = W^i_k T_{glo} \quad i \in [1, O] \\ Z_i = atten(Q_i, K_i, V_i) \quad i \in [1, O] \\ T_{fea} = W_O(Z_1 \oplus Z_2 \oplus, \ldots, Z_O) \quad i \in [1, O] \end{cases} \tag{13}$$

where $W_O \in \mathbb{R}^{D_{glo} * M d_v}$ denotes the output matrix in O group projection space. $W^i_q \in \mathbb{R}^{D_d * D_{glo}}$, $W^i_k \in \mathbb{R}^{D_d * D_{glo}}$ and $W^i_v \in \mathbb{R}^{D_v * D_{glo}}$ denote the input matrix in O group projection space. T_{fea} denotes the final trust feature, which perform a decisive role in the establishment of trust relationships between u_a and u_b. In turn, the metapath-guided features obtained from LSTM networks and MLP networks are optimized by back-propagation of the multi-headed attention network. Ultimately, the representation of the model is enhanced. The multi-headed attention network is a mechanism that allows the model to attend to different aspects of the input data (i.e., the user–item–user interactions generated by category-aware metapaths) simultaneously. It consists of multiple attention heads, each of which learns to focus on a specific subset of user–item–user interactions. By doing so, the model can capture diverse patterns and correlations within the data, enhancing its ability to distinguish between different metapaths for trust prediction. During the training process, each attention head in the network learns its own set of attention weights, which determine the importance of each user–item–user interaction for trust prediction. The final trust prediction is then obtained by combining the outputs of all attention heads, where each head contributes to the prediction based on its learned attention weights.

By analyzing the attention weights learned by the model, the most influential factors in trust prediction can be identified. For example, they can uncover which item categories or types of user–item interactions have a significant impact on trust building in social networks. These pieces of information can provide valuable insights into how trust forms and evolves among users.

Embedding real trust relationships between users in the network helps to filter out irrelevant user–item–user interactions by introducing a mechanism to prioritize and weigh the significance of different interactions. On the one hand, users in the metapath "UUU" have real trust relationships with each other, whereas users in the metapath "UICIU" are any two users (not necessarily having trust relationships). After LSTM network and MLP network, these two metapath features are incorporated into the multi-headed attention network. In the multi-head attention network, more weight will be given to the interactions generated by two users who trust each other. Then, the metapath "UUU" imposes restrictions on the interactions generated by the metapath "UICIU". As a result, real trust relationships are embedded in the network and will help filter out irrelevant user–item–user interactions.

4.4. Trust evaluation

Following the trust features of each user pair (u_a, u_b) obtained from the multi-headed attention network, T_{fea} is further transformed to obtain the trust value. The specific equation is shown below.

$$\hat{T}_{a,b} = softmax(linear(T_{fea})) \tag{14}$$

where one layer of linear functions is utilized to decrease the dimension of T_{fea}. $\hat{T}_{a,b}$ is represented as the trust value between pairs of users (u_a, u_b), which is obtained through softmax function. Most of works as described in this paper converts continuous predicted values into discrete binary ranges by using threshold settings. Here, we consider that $\hat{T}_{a,b}$ is greater than 0.5 where there is a trust relationship between u_a and u_b, otherwise it is a distrust relationship.

The task of the whole network can be considered as a classification and supervision model, while the cross-entropy function is more suitable for the classification model. For the overall optimization of the model, we adopt the cross-entropy function as loss function to optimize the whole network. It is shown in the following equation.

$$\mathcal{L}_{loss} = -\sum_{T_{a,b} \in TRU} T_{a,b} \log(\hat{T}_{a,b}) \tag{15}$$

Where $T_{a,b}$ represents the ground-truth label between pairs of users. To ensure gradient descent in model training, we use the Adam optimizer [44] to iteratively update the parameters in the entire neural network.

In summary, the MetaTrust model finds deep trust-oriented features in a metapath-guided manner, which uses LSTM networks and MLPs to maximize the extraction of interaction features between users including both category-aware behavioral preference-based features and trust property-based features. The intuitive idea of the metapath *UICIU* design is to model the interaction behavior of two users explicitly without repetition. The intuitive idea of metapath *UUU* is that maintains the trust neighbors of 1-hop and 2-hop in the metapath, and the usage of meanpooling operation is to weaken the influence of trust neighbors of 2-hop on trust prediction. Through the feature extraction for both metapaths, the trust-oriented comprehensive features are obtained to provide a deterministic judgment for pairwise trust relationship. On the whole, this approach is interaction-based and supports the sociology of fundamental definition on trust.

The numerical model proposed in the MetaTrust framework is based on trust prediction, which aims to estimate trust relationships between users in web-based applications. The model incorporates category-aware metapaths to generate user–item–user interactions, which provide a supplement to the traditional user–user trust interactions. Regarding the emergence of other types of trust, such as Luhmann trust [45], the numerical model is empowered to capture category-specific interactions that opens up possibilities for exploring different trust dimensions. Luhmann trust, as conceptualized by Niklas Luhmann, refers to the trust in system structures, and it is distinct from trust between individual actors. The model's capacity to consider item categories could potentially be extended to incorporate system-level trust dynamics and the stability of trust relationships at a larger scale. Furthermore, the proposed MetaTrust model could contribute to validating the trust continuum hypothesis, which suggests that trust can exist along a spectrum, ranging from strong personal trust to weaker impersonal or institutional trust. By examining trust patterns within different item categories and user interactions, the model could shed light on the continuum of trust in diverse contexts and offer insights into how trust evolves or varies across various levels of human interaction and system structures.

Table 2
Statistics for the three real-world datasets.

Dataset	Epinions	Ciao	Yelp
#Users	7458	2248	1 809 412
#Items	6149	16 861	134 109
#Rating	209 769	36 065	6 011 303
#Category	27	28	31
#Trust relationships	300 548	57 544	10 800 586
Data sparsity	0.4574%	0.0951%	0.0025%

5. Experiments

In this section, we perform a series of extensive experiments to answer the following research questions:

- **RQ1:** How does the MetaTrust model compare with state-of-the-art baseline approaches in trust prediction models?
- **RQ2**: How does the setting of the trust value threshold affect the performance of different methods?
- **RQ3**: How do different metapaths affect the MetaTrust model?
- **RQ4**: How do the parameter settings in different meta-paths affect the MetaTrust model?

5.1. Datasets

Three publicly available datasets are selected to accomplish the task of MetaTrust Model. The statistics of the datasets are described as shown in Table 2.

Epinions and Ciao. Epinions and Ciao datasets [8] are derived from two product review sites, which contain three types of interactions, i.e., trust interactions between users, rating interactions between users and items, and category interactions between items. Ratings of interacted items by users from 1 to 5. In addition, to eliminate the effect of noisy data as much as possible, we keep at least 15 items that users have interacted with and at least 10 ratings of items in Epinions dataset.

Yelp. Yelp dataset [46] contains ratings given by users to local businesses and their attribute information. Each user maintains a list of friends. Ratings of business rated by users are from 1 to 5. Additionally, we take into account that: (1) a business is equal to an item, and it is a notion that is frequently used in recommender systems; (2) if two users are friends, then two users trust each other equivalently. In our experiments, we extract part of the data from the original dataset.

5.2. Experimental setup

The implementation of our model is on pytorch.[1] Specifically, we cross-validate the embedding dimension in the metapath embedding layer as {16, 32, 64, 96, 128}, and the results of the pre-training indicate that the model achieves the best performance when the embedding size is set to 64. In the meta-path embedding layer, we set F as 300 and G as 300. In the metapath feature extraction layer, the horizontal depth of the LSTM network is set as 2, and the vertical depth is variable depending on the number of categories based on metapaths

[1] https://pytorch.org/.

Table 3
Effectiveness experiments on three datasets. A higher precision or F1 value indicates better performance. To facilitate the reading of the experimental results, we also present the improvement of MetaTrust approach over the MemTrust model. The larger improvement ratio indicates a better performance.

Dataset	Training size	Metric	JMF	STNE	LINE	Guardian	MemTrust	MetaTrust	Improv.
Epinions	5%	Precision	0.5140	0.4999	0.5806	0.4997	0.8924	0.9251	3.66%
		F1	0.4921	0.6648	0.7346	0.6654	0.8361	0.9528	13.96%
	30%	Precision	0.5002	0.5000	0.5742	0.5004	0.9549	0.9928	3.97%
		F1	0.4356	0.6662	0.7295	0.6667	0.9157	0.9855	7.62%
	50%	Precision	0.4987	0.8977	0.5715	0.5001	0.9680	0.9950	2.79%
		F1	0.4369	0.9508	0.7943	0.6668	0.9063	0.9810	8.24%
	70%	Precision	0.5018	0.8505	0.5710	0.5006	0.9081	0.9906	9.09%
		F1	0.4340	0.9519	0.7942	0.6655	0.9057	0.9779	7.97%
	90%	Precision	0.5042	0.9039	0.8711	0.5033	0.9512	0.9953	4.64%
		F1	0.4371	0.9432	0.794	0.6685	0.9183	0.9801	6.73%
Ciao	5%	Precision	0.5254	0.4856	0.4920	0.4890	0.8280	0.8320	0.48%
		F1	0.3416	0.6344	0.6580	0.6481	0.7368	0.7911	7.37%
	30%	Precision	0.5217	0.4876	0.4899	0.4858	0.9355	0.9365	0.11%
		F1	0.3433	0.6494	0.6562	0.6480	0.6917	0.8115	17.32%
	50%	Precision	0.5560	0.9156	0.4867	0.4878	0.9567	0.9678	1.16%
		F1	0.3497	0.8982	0.6529	0.6460	0.8972	0.8988	0.18%
	70%	Precision	0.5200	0.9223	0.4846	0.4899	0.9114	0.9515	4.4%
		F1	0.3574	0.9006	0.6513	0.6443	0.9138	0.9237	1.08%
	90%	Precision	0.5198	0.9116	0.4861	0.4867	0.8882	0.9821	10.57%
		F1	0.3579	0.8927	0.6523	0.6485	0.8926	0.8965	0.48%
Yelp	5%	Precision	0.5028	0.7214	0.7223	0.7214	0.6583	0.9030	37.17%
		F1	0.476	0.8381	0.5915	0.8381	0.7876	0.9231	16.96%
	30%	Precision	0.5259	0.7215	0.7216	0.7215	0.8151	0.9383	15.11%
		F1	0.5262	0.8382	0.5905	0.8382	0.8666	0.9393	8.39%
	50%	Precision	0.5090	0.7214	0.7215	0.7214	0.9344	0.9653	3.31%
		F1	0.5083	0.8381	0.5909	0.8381	0.9169	0.9603	4.73%
	70%	Precision	0.4904	0.7228	0.7213	0.7217	0.9312	0.9675	3.90%
		F1	0.4524	0.8371	0.5904	0.8373	0.9171	0.9396	2.42%
	90%	Precision	0.4940	0.7236	0.7221	0.7223	0.9526	0.9799	2.87%
		F1	0.4573	0.8365	0.6004	0.8381	0.9279	0.9708	4.62%

UICIU in particular dataset; the number of MLP layers is set as 2. In the Inter-metapath fusion layer, the group projected subspace in the multi-headed attention network is set as 4. For the whole network running, the training batch is set as 16 and the learning rate of the network is set as 1e−5. We train the model for a total of 20 epochs for the outcomes of each experiment.

Baseline method. We compare the MetaTrust model with several other cutting-edge methods, either through publicly available code or the code we implemented. From the perspective of network types, we choose two network embedding methods, one of which is a homogeneous network embedding type (STNE, LINE, Guardian) and the other is a heterogeneous network embedding method (MemTrust, JMF). Specifically,

- **STNE**. STNE [3] embeds user–user trust interaction. The multiple propagation paths between two users are fused by using social balance theory. Thus, the trust relationship is evaluated.
- **LINE**. LINE [47] models an objective function to preserve the interaction information of 1-hop and 2-hop trusted neighbors. Pairwise features are exploited to measure whether a trust relationship exists.
- **Guardian**. Guardian [13] converts user–user trust interactions into a graph structure. The co-weight are shared through GCN to learn the global trust properties of users.
- **MemTrust**. MemTrust [20] exploits heterogeneous information such as users' ratings of items to mine the behavioral features of users and then measure the trust relationship between two users.
- **JMF**. JMF [19] designs a joint manifold factorization to collectively learn the similar behavioral characteristics of users and the features of trust interaction between users are reserved simultaneously.

5.3. The performance on trust prediction (RQ1)

To validate the effectiveness of the MetaTust model, we compared it with other competitive methods in terms of different training scales. As shown in Table 3, by observing the variations of F1 values, MetaTrust technique outperformed STNE, LINE, and Guardian methods by an average of 16.77%, 22.06%, and 46.34% in the Epinions dataset. The MetaTrust technique improved on average by 8.71%, 32.13%, 33.59% and 12.54%, 59.02%, 12.49% in the Ciao and Yelp datasets. These homogeneous network embedding-based methods only consider trust interactions between users and ignore the behavioral characteristics of users, which do not capture the trust-oriented features comprehensively. Compared to the MemTrust and JMF methods, the MetaTrust approach improved on average 8.81%, 118.15% on F1 values in Epinions dataset. In the Ciao and Yelp datasets, the MetaTrust method improved on average 4.59%, 14.7%, 6.73% and 94.74% about F1 values synthetically. This is because the MemTrust approach utilizes LSTM networks to extract behavioral features from heterogeneous information, which overlooks the search for the user's trusted neighbors. The JMF leverages matrix decomposition to extract behavioral patterns that are comparable to those found in the trust graph by integrating a large number of parameters. Since personalized trust features are not accurately captured by this comprehensive decomposition of each user's records, accuracy is decreased.

When the data amount ranges from 50% to 90%, all approaches operate quite consistently. This is due to the fact that they can mine enough beneficial features to foster user trust at 50% of the data. The performance of the other approaches declines when the data density is less than 50%, while MetaTrust maintains good performance.

Here, we concentrate on the scenario in which there is a 5% training set size where the data is extremely sparse. The other five methods perform very poorly, while MetaTrust still maintains high accuracy. Two factors are responsible for MetaTrust's success: (1) Through metapath *UICIU*, the information search from users interacting with the

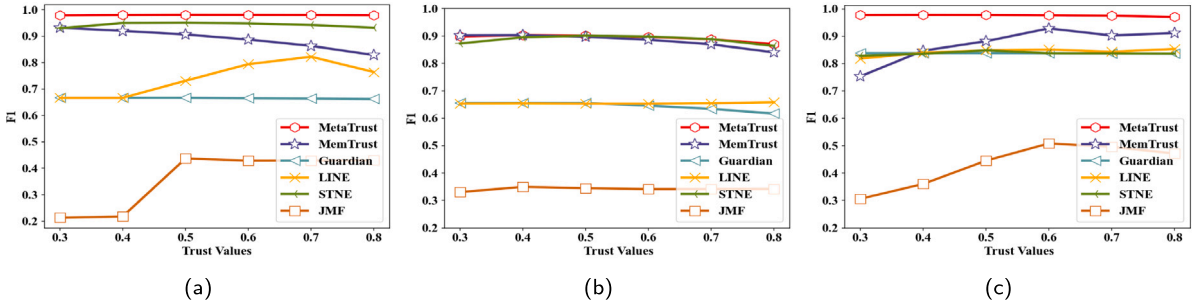

Fig. 4. The effect of trust values on Epinions (a), Ciao (b) and Yelp (c).

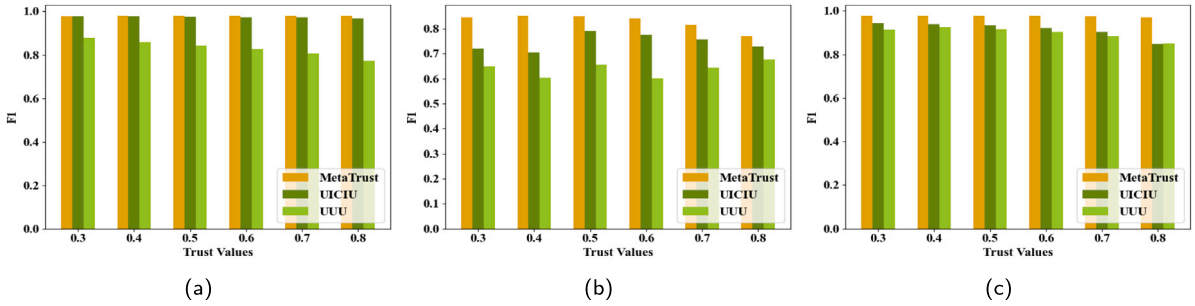

Fig. 5. The effect of different metapath on Epinions (a), Ciao (b) and Yelp (c).

same items is expanded to the information search of items with the same category, broadening the way of interactive information search and obtaining more powerful features. Moreover, the behavioral characteristics of user sharing are explored. Furthermore, the interaction between users' trust neighbors is expressly expanded by the metapath *UUU* and it is simpler to build a trust relationship when two users have more trust neighbors in common. (2) Multi-headed attention networks guide the extraction of metapath features through LSTM and MLP networks, facilitating accurate metapath feature generation.

5.4. The effect of trust values (RQ2)

When the trust value threshold is set higher, a more stringent metapath-guided user feature matching condition is acquired. Thus, gradually increasing trust values are set as {0.3, 0.4, 0.5, 0.6, 0.7, 0.8} from which the robustness of each approach is validated. As shown in Fig. 4, MetaTrust has the best performance in comparison with the other five methods. Among the homogeneous network embedding methods (Guardian, LINE, STNE), Guardian performs relatively poorly because it uses GCN to aggregate the user's outward/inward trust neighbor features to represent the matching features for user trust building. The development of trust-oriented user features is hampered by joining the calculation and sharing the weights with the high-order trust neighbors. As a result, user features with noisy information are obtained. With increasing trust value thresholds, Guardian has no access to more accurate user features.

LINE aggregates the features of multi-hop trust neighbors, and a large number of less relevant 2-hop trust neighbors are aggregated into user features, which reduces the accuracy of matching user trust features. Due to the fact that STNE uses the modified Skip-Gram model to identify users' latent features, it performs nearly as well as the MetaTrust solution. It is able to capture trust-directed features effectively.

Given varying trust levels, MetaTrust has maintained a generally outstanding and seamless performance. More accurate interaction information is filtered out through the metapath as the threshold of trust value rises. Additionally, the multi-headed attention mechanism allows it to efficiently extract correlated features and contribute significantly to trust prediction.

5.5. The effect of different metapath (RQ3)

As shown in Fig. 5, there are two metapaths in MetaTrust model. To further check the impact of different metapaths on the model, we validated the F1 performance of different metapaths under varying trust values. In comparison to the approach that only employs metapath *UUU*, the method that only utilizes the metapath *UICIU* performs better. Here, the metapath *UICIU* method can locate records of user interactions focused on "category" and can gather user behavior characteristics. The outcomes of this study also support the social theory that users who have comparable behavioral characteristics are more likely to form trust relationships. Furthermore, the multi-layer LSTM network can effectively capture the association features of multiple *UICIU* paths.

The metapath *UUU*-based approach aggregates the information of 1-hop and 2-hop trusted neighbors. The MLP network can be used to extract the feature representation of multi-hop trusted neighbors. This approach reflects the trust interaction of users and can influence the establishment of trust relationships between users to some degree, but it falls short of sufficiently generalizing the necessary conditions for the establishment of trust relationships. Therefore, this results in rather low accuracy.

By combining these two approaches, MetaTrust succeeds in capturing the metapath-guided trust features. The use of redundant user–item interactions in the proposed MetaTrust model improves trust prediction accuracy by addressing the data sparsity problem. On the one hand, the traditional trust prediction approaches that heavily rely on user–user trust interactions, data sparsity arises when there are limited trust interactions between users. This lack of sufficient data can lead to inaccurate trust predictions, especially for users with few or no direct trust interactions. On the other hand, by generating redundant user–item interactions through category-aware metapaths, the model creates additional trust-related data points. These interactions are based on common item categories that users have interacted with, effectively capturing trust patterns influenced by the types of items users engage with. The four metapath-guided features for paired users are aggregated and automatically assigned weights by using a multi-headed attention network, which can further obtain trust-oriented integrated features.

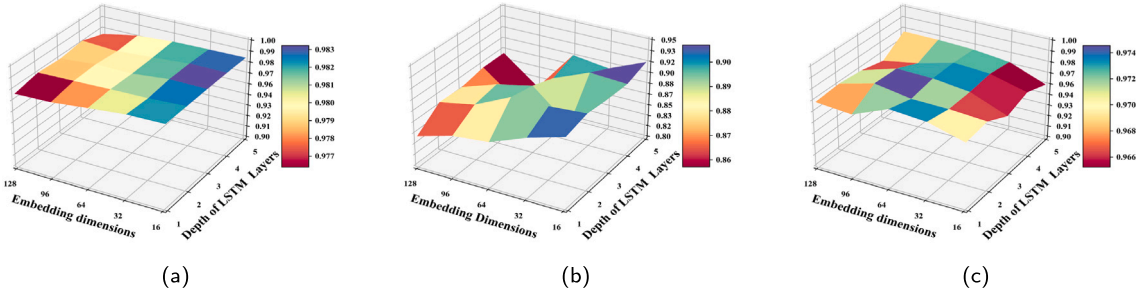

Fig. 6. The effect of network parameters on Epinions (a), Ciao (b) and Yelp (c).

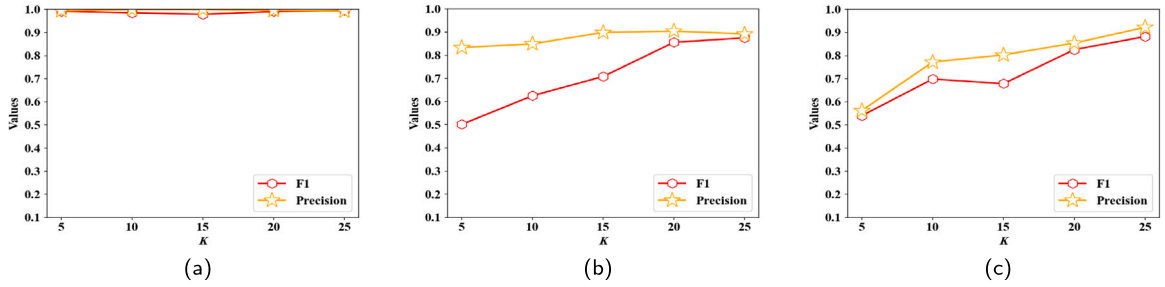

Fig. 7. The effect of different number of categories on Epinions (a), Ciao (b) and Yelp (c).

Among the three datasets, the MetaTrust approach performs best in the Epinions dataset.

5.6. The effect of parameter sensitivity (RQ4)

(1) *The Dimensions of Embedding*. We confirm the impact of the dimension of embedding on MetaTrust model from the standpoint of the network framework. The dimension of embedding is set to {16, 32, 64, 96, 128}. As shown in Fig. 6(a)(b), when the dimensions of the embedding is set to 16 and 64, the performance is relatively good in the Epinions and Ciao datasets. The F1 values of the model starts to drop when the dimension of embedding is steadily increased to 128. There are two main reasons. (a) The more elements of embedding used, the more irrelevant variables are mapped into the same data space, which causes the model to become overfit and degrade its performance. (b) The Epinions dataset has a higher density than the Ciao dataset. The dataset density is negatively correlated with the embedding dimension. In particular, the model performs better in the three datasets when the size of embedding is set to 64.

(2) *The Horizontal Depth of LSTM Network*. We investigate the impact of the horizontal depth of the LSTM network and the horizontal depth of LSTM network is set to {1, 2, 3, 4,5}. With the increasing horizontal depth of the LSTM network from 1 to 3, the accuracy of the model increases gradually as shown in Fig. 6. Thus, the metapaths can be successfully described by the appropriate number of LSTM layers. When the horizontal depth of the LSTM is set to 5, the model performs the poorest. This is because the extraction of the correlation of the features is hampered if the paths from input data to output features in LSTM network are too long.

In comparison to the three datasets, the MetaTrust model performs relatively unsmoothly in the Ciao dataset. There are two reasons for variations. (a) As the dimensionality of embedding gradually increases, the noise information is also incorporated into the embedding space. (b) The data sparsity in the Ciao dataset is large, which indicates that the interaction information selected randomly for each experiment is out of balance.

In summary, the MetaTrust model uses the embedding technique to convert the original path data to the same feature space. Then, the LSTM network and MLPs are utilized to analyze the user interaction features across various meta-paths in order to construct the user trust features. Indirectly, we demonstrate the effectiveness of obtaining user trust features by metapaths.

(3) *The Number of Categories*. As shown in Fig. 7, the number of path based on metapath *UICIU* depends on the number of categories of the items that the user has interacted with. In the three datasets, the number of categories is set as {5, 10, 15, 20, 25}. As the number of categories increases, the performance of MetaTrust method turns out to be better, i.e., the values of precision and F1 are slowly increasing on three datasets. The growing number of categories implies an expansion of behavioral records among users, which indicates that more behavioral characteristics of users can be explored. More importantly, the MetaTrust model still performs well even if there are few categories. In conclusion, these results show the robustness of our proposed approach.

6. Conclusion

To the best of our knowledge, this is the first attempt that proposes a metapath-guided trust prediction model in heterogeneous social networks. The behavioral characteristics and trust properties of users have an important impact on the establishment of trust relationships among users. Thus, we design two types of metapaths focused on behavioral interaction records and trust neighbors of users, which can effectively capture key information. Embedding techniques are utilized to capture the structural and semantic information of both metapaths. In order to efficiently extract the features of a single metapath, LSTM network and MLP network are employed to extract two metapath features with behavioral and trust features. To further discriminate the importance of the two metapath features, the multi-headed attention network can automatically calculate the relevance and assign different attention scores for the two type of metapath features. As a result, global trust features are subsequently obtained. Finally, the softmax function is applied to distinguish the trust value between any two users. Compared with other competitive approaches, our proposed trust model has superior and stable performance. In particular, in the case of extremely sparse data, MetaTrust is still able to tightly capture trust-oriented information and effectively characterize user features.

Understanding trust building in social networks has various potential applications. One such application is the improvement of recommendation systems. By incorporating trust information into recommendation algorithms, personalized and trustworthy recommendations can be provided to users based on the trust relationships. This can enhance user satisfaction and engagement with the platform. Another potential application is in the design of social network platforms. Understanding trust dynamics can help in developing more effective user engagement strategies. For example, identifying influential users with high trust ratings can facilitate targeted marketing campaigns and community management efforts.

In the future, our model can be extended to marketing, recommender systems and distributed platform cooperation [48,49]. Since behavioral preferences of users are variable, and not all preferences are assigned equal attention, a trust model guided by the distribution of users' preferences is necessary to be established. Additionally, a variety of social connections (such as friends and relatives) within heterogeneous social networks have an impact on how trust is developed. It is expected that the developed trust prediction model can seize and integrate crucial social relationships.

CRediT authorship contribution statement

Yanwei Xu: Conceptualization, Data curation, Formal analysis, Methodology, Writing – original draft, Writing – review & editing. **Zhiyong Feng:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Meng Xing:** Investigation, Visualization, Writing – review & editing. **Hongyue Wu:** Supervision, Writing – review & editing. **Shizhan Chen:** Writing – review & editing. **Xiao Xue:** Writing – review & editing. **Schahram Dustdar:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

References

[1] Paolo Massa, Paolo Avesani, Controversial users demand local trust metrics: An experimental study on epinions. com community, in: AAAI, Vol. 1, 2005, pp. 121–126.

[2] Paola Sapienza, Anna Toldra-Simats, Luigi Zingales, Understanding trust, Econ. J. 123 (573) (2013) 1313–1332.

[3] Pinghua Xu, Wenbin Hu, Jia Wu, Weiwei Liu, Bo Du, Jian Yang, Social trust network embedding, in: 2019 IEEE International Conference on Data Mining (ICDM), IEEE, 2019, pp. 678–687.

[4] Guangchi Liu, Qi Chen, Qing Yang, Binhai Zhu, Honggang Wang, Wei Wang, Opinionwalk: An efficient solution to massive trust assessment in online social networks, in: IEEE INFOCOM 2017-IEEE Conference on Computer Communications, IEEE, 2017, pp. 1–9.

[5] Guangchi Liu, Chenyu Li, Qing Yang, Neuralwalk: Trust assessment in online social networks with neural networks, in: IEEE INFOCOM 2019-IEEE Conference on Computer Communications, IEEE, 2019, pp. 1999–2007.

[6] Jianli Zhao, Wei Wang, Zipei Zhang, Qiuxia Sun, Huan Huo, Lijun Qu, Shidong Zheng, TrustTF: A tensor factorization model using user trust and implicit feedback for context-aware recommender systems, Knowl.-Based Syst. 209 (2020) 2–8.

[7] David Crandall, Dan Cosley, Daniel Huttenlocher, Jon Kleinberg, Siddharth Suri, Feedback effects between similarity and social influence in online communities, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 2008, pp. 160–168.

[8] Jiliang Tang, Huiji Gao, Xia Hu, Huan Liu, Exploiting homophily effect for trust prediction, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, 2013, pp. 53–62.

[9] Jiliang Tang, Huiji Gao, Huan Liu, Atish Das Sarma, eTrust: Understanding trust evolution in an online world, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2012, pp. 253–261.

[10] Xu Chen, Yuyu Yuan, Mehmet Ali Orgun, Lilei Lu, A topic-sensitive trust evaluation approach for users in online communities, Knowl.-Based Syst. 194 (2020) 1–16.

[11] Yanxin Xu, Zaiwu Gong, Jeffrey Yi-Lin Forrest, Enrique Herrera-Viedma, Trust propagation and trust network evaluation in social networks based on uncertainty theory, Knowl.-Based Syst. 234 (2021) 1–16.

[12] Ramanthan Guha, Ravi Kumar, Prabhakar Raghavan, Andrew Tomkins, Propagation of trust and distrust, in: Proceedings of the 13th International Conference on World Wide Web, 2004, pp. 403–412.

[13] Wanyu Lin, Zhaolin Gao, Baochun Li, Guardian: Evaluating trust in online social networks with graph convolutional networks, in: IEEE INFOCOM 2020-IEEE Conference on Computer Communications, IEEE, 2020, pp. 914–923.

[14] Wanyu Lin, Baochun Li, Medley: Predicting social trust in time-varying online social networks, in: IEEE INFOCOM 2021-IEEE Conference on Computer Communications, IEEE, 2021, pp. 1–10.

[15] Jiliang Tang, Huiji Gao, Huan Liu, mTrust: Discerning multi-faceted trust in a connected world, in: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, 2012, pp. 93–102.

[16] Qi Wang, Weiliang Zhao, Jian Yang, Jia Wu, Shan Xue, Qianli Xing, S Yu Philip, C-DeepTrust: A context-aware deep trust prediction model in online social networks, IEEE Trans. Neural Netw. Learn. Syst. (2021) http://dx.doi.org/10.1109/TNNLS.2021.3107948.

[17] Xiaoming Zheng, Yan Wang, Mehmet A Orgun, Guanfeng Liu, Haibin Zhang, Social context-aware trust prediction in social networks, in: International Conference on Service-Oriented Computing, Springer, 2014, pp. 527–534.

[18] Qi Wang, Weiliang Zhao, Jian Yang, Jia Wu, Chuan Zhou, Qianli Xing, Atnetrust: Attributed trust network embedding for trust prediction in online social networks, in: 2020 IEEE International Conference on Data Mining (ICDM), IEEE, 2020, pp. 601–610.

[19] Jin Huang, Feiping Nie, Heng Huang, Yi-Cheng Tu, Trust prediction via aggregating heterogeneous social networks, in: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, 2012, pp. 1774–1778.

[20] Yanwei Xu, Zhiyong Feng, Xiao Xue, Shizhan Chen, Hongyue Wu, Xian Zhou, Meng Xing, Hongqi Chen, MemTrust: Find deep trust in your mind, in: 2021 IEEE International Conference on Web Services (ICWS), IEEE, 2021, pp. 598–607.

[21] Lianyong Qi, Wenmin Lin, Xuyun Zhang, Wanchun Dou, Xialong Xu, Jinjun Chen, A Correlation graph based approach for personalized and compatible web apis recommendation in mobile app development, IEEE Trans. Knowl. Data Eng. http://dx.doi.org/10.1109/TKDE.2022.3168611.

[22] Yang Xu, Ziming Liu, Cheng Zhang, Ju Ren, Yaoxue Zhang, Xuemin Shen, Blockchain-based trustworthy energy dispatching approach for high renewable energy penetrated power systems, IEEE Internet Things J. 9 (12) (2021) 10036–10047.

[23] Maryam Nooraei Abadeh, Mansooreh Mirzaie, A differential machine learning approach for trust prediction in signed social networks, J. Supercomput. (2023) 1–24.

[24] Jinbo Liu, Yunliang Chen, Xiaohui Huang, Jianxin Li, Geyong Min, GNN-based long and short term preference modeling for next-location prediction, Inform. Sci. 629 (2023) 1–14.

[25] Yan Jia, Zhaoquan Gu, Zhihao Jiang, Cuiyun Gao, Jianye Yang, Persistent graph stream summarization for real-time graph analytics, World Wide Web (2023) 1–21.

[26] Yang Xu, Cheng Zhang, Guojun Wang, Zheng Qin, Quanrun Zeng, A blockchain-enabled deduplicatable data auditing mechanism for network storage services, IEEE Trans. Emerg. Top. Comput. 1–12, http://dx.doi.org/10.1109/TETC.2020.3005610.

[27] Cheng Zhang, Yang Xu, Yupeng Hu, J Wu, Ju Ren, Yaoxue Zhang, A blockchain-based multi-cloud storage data auditing scheme to locate faults, IEEE Trans. Cloud Comput. http://dx.doi.org/10.1109/TCC.2021.3057771.

[28] Yang Xu, Ju Ren, Yan Zhang, Cheng Zhang, Bo Shen, Yaoxue Zhang, Blockchain empowered arbitrable data auditing scheme for network storage as a service, IEEE Trans. Serv. Comput. 13 (2) (2019) 289–300.

[29] Yuwen Liu, Zuolong Song, Xiaolong Xu, Wajid Rafique, Xuyun Zhang, Jun Shen, Mohammad R Khosravi, Lianyong Qi, Bidirectional GRU networks-based next POI category prediction for healthcare, Int. J. Intell. Syst. 37 (7) (2022) 4020–4040.

[30] Shuai Zhang, Lina Yao, Aixin Sun, Yi Tay, Deep learning based recommender system: A survey and new perspectives, ACM Comput. Surv. 52 (1) (2019) 1–38.

[31] Lina Yao, Quan Z Sheng, Anne HH Ngu, Jian Yu, Aviv Segev, Unified collaborative and content-based web service recommendation, IEEE Trans. Serv. Comput. 8 (3) (2014) 453–466.

[32] Xiaolong Xu, Qinting Jiang, Peiming Zhang, Xuefei Cao, Mohammad R Khosravi, Linss T Alex, Lianyong Qi, Wanchun Dou, Game theory for distributed IoV task offloading with fuzzy neural network in edge computing, IEEE Trans. Fuzzy Syst. (2022) http://dx.doi.org/10.1109/TFUZZ.2022.3158000.

[33] Lianyong Qi, Yihong Yang, Xiaokang Zhou, Wajid Rafique, Jianhua Ma, Fast anomaly identification based on multi-aspect data streams for intelligent intrusion detection toward secure industry 4.0, IEEE Trans. Ind. Inform. (2021) http://dx.doi.org/10.1109/TII.2021.3139363.

[34] Sumit Negi, Santanu Chaudhury, Link prediction in heterogeneous social networks, in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, 2016, pp. 609–617.

[35] Xiaolong Xu, Hao Tian, Xuyun Zhang, Lianyong Qi, Qiang He, Wanchun Dou, DisCOV: distributed COVID-19 detection on X-ray images with edge-cloud collaboration, IEEE Trans. Serv. Comput. (2022) http://dx.doi.org/10.1109/SERVICES55459.2022.00036.

[36] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, Yongliang Li, Metapath-guided heterogeneous graph neural network for intent recommendation, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 2478–2486.

[37] Haipeng Dai, Yun Xu, Guihai Chen, Wanchun Dou, Chen Tian, Xiaobing Wu, Tian He, ROSE: Robustly safe charging for wireless power transfer, IEEE Trans. Mob. Comput. 21 (6) (2020) 2180–2197.

[38] Rong Gu, Kai Zhang, Zhihao Xu, Yang Che, Bin Fan, Haojun Hou, Haipeng Dai, Li Yi, Yu Ding, Guihai Chen, et al., Fluid: dataset abstraction and elastic acceleration for cloud-native deep learning training jobs, in: 2022 IEEE 38th International Conference on Data Engineering (ICDE), IEEE, 2022, pp. 2182–2195.

[39] Florian Skopik, Daniel Schall, Schahram Dustdar, Start trusting strangers? bootstrapping and prediction of trust, in: International Conference on Web Information Systems Engineering, Springer, 2009, pp. 275–289.

[40] Florian Skopik, Daniel Schall, Schahram Dustdar, The cycle of trust in mixed service-oriented systems, in: 2009 35th Euromicro Conference on Software Engineering and Advanced Applications, IEEE, 2009, pp. 72–79.

[41] Yanwei Xu, Zhiyong Feng, Xian Zhou, Meng Xing, Hongyue Wu, Xiao Xue, Shizhan Chen, Chao Wang, Lianyong Qi, Attention-based neural networks for trust evaluation in online social networks, Inform. Sci. 630 (2023) 507–522.

[42] Bin Guo, Yasan Ding, Lina Yao, Yunji Liang, Zhiwen Yu, The future of false information detection on social media: New perspectives and trends, ACM Comput. Surv. 53 (4) (2020) 1–36.

[43] Yang Li, Guodong Long, Tao Shen, Tianyi Zhou, Lina Yao, Huan Huo, Jing Jiang, Self-attention enhanced selective gate with entity-aware embedding for distantly supervised relation extraction, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 8269–8276, 05.

[44] Zijun Zhang, Improved adam optimizer for deep neural networks, in: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), IEEE, 2018, pp. 1–2.

[45] Niklas Luhmann, Trust and Power, John Wiley & Sons, 2018.

[46] Chuan Shi, Binbin Hu, Wayne Xin Zhao, S. Yu Philip, Heterogeneous information network embedding for recommendation, IEEE Trans. Knowl. Data Eng. 31 (2) (2018) 357–370.

[47] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, Qiaozhu Mei, Line: Large-scale information network embedding, in: Proceedings of the 24th International Conference on World Wide Web, 2015, pp. 1067–1077.

[48] Florian Skopik, Daniel Schall, Schahram Dustdar, Modeling and Mining of Dynamic Trust in Complex Service-Oriented Systems, Springer, 2011.

[49] Florian Skopik, Daniel Schall, Schahram Dustdar, Trusted information sharing using SOA-based social overlay networks, Int. J. Comput. Sci. Appl. 10 (2) (2013).