# Deep Learning-Embedded Social Internet of Things for Ambiguity-Aware Social Recommendations

Zhiwei Guo, *Member, IEEE,* Keping Yu, *Member, IEEE,* Yu Li, Gautam Srivastava, *Senior Member, IEEE* and Jerry Chun-Wei Lin, *Senior Member, IEEE*

***Abstract*—With the increasing demand of users for personalized social services, social recommendation (SR) has been an important concern in academia. However, current research on SR universally faces two main challenges. On the one hand, SR lacks the considerable ability of robust online data management. On the other hand, SR fails to take the ambiguity of preference feedback into consideration. To bridge these gaps, a deep learning-embedded social Internet of Things (IoT) is proposed for ambiguity-aware SR (SIoT-SR). Specifically, a social IoT architecture is developed for social computing scenarios to guarantee reliable data management. A deep learning-based graph neural network model that can be embedded into the model is proposed as the core algorithm to perform ambiguity-aware SR. This design not only provides proper online data sensing and management but also overcomes the preference ambiguity problem in SR. To evaluate the performance of the proposed SIoT-SR, two real-world datasets are selected to establish experimental scenarios. The method is assessed using three different metrics, selecting five typical methods as benchmarks. The experimental results show that the proposed SIoT-SR performs better than the benchmark methods by at least 10% and has good robustness.**

***Index Terms*—Social IoT, social computing, deep learning, graph neural networks.**

## I. Introduction

THE past decade has witnessed the rapid development of communication networks, bringing profound changes to society [1], [2]. Through a variety of social media platforms, the Internet facilitates closer connections among people [3]. Simultaneously, the continuous enrichment of spiritual and cultural content urges people to pursue more personalized social services [4], [33]. To meet demand, social computing has become a newly emerging research area [5]. Social computing exploits advanced computational technologies to deeply analyze various rules, characteristics, and patterns in social networks to discover the hidden social needs of users [6]. Among these areas, social recommendation (SR) is an important area of focus [7]. SR aims to suggest appropriate items to users by integrating the preference features of users as well as the contextual characteristics of social networks [8]. The most straightforward form of preference feedback is ratings, which can reflect a preference for items [9]. As a social network can be thought of as a heterogeneous graph network, the contextual characteristics include information on attributes and relations [10]. Currently, efficient social recommendations have become the key point in the operation of social network platforms [11], [34].

During the past few years, research related to SR has received considerable attention globally. Accordingly, several representative technical approaches have been proposed in the area of SR [12]-[27]. Jiang *et al.* [12] inferred inter-domain and intra-domain correlations from tagging space, and constructed a modified matrix factorization model. Rafailidis *et al.* [25] formulated the learning of both user preferences and social influence as a joint optimization problem, so as to set up fine-grained feature spaces. Fan *et al.* [27] simultaneously captured interactions and opinions in the user-item graph, and proposed a graph neural network for SR. As SR relies on fruitful source data to efficiently capture social characteristics, the main solutions used in existing research look to extend feature spaces by inferring unknown relationships from known relationships [10].

It can be found from related works that exploration of effective SR remains hard. Although positive strides have been made in the area of SR, two main challenges remain. First, existing methods are highly reliable on offline data while training models, which is not sufficient. Large-scale online data are required for SR as support, meaning that recommendation models continuously adapt to time-varying contextual modes. Second, almost all of the existing research assumes that the preference feedback of users is clear. In other words, users are only allowed to give one rating or grade to clearly express their preference degrees. However, preference feedback is influenced by multiple factors and is thus ambiguous, which is illustrated by a typical example in Fig. 1. For a given movie, users may have different comments on its different

This work was supported in part by the State Language Commission Research Program of China under grant YB135-121, in part by the Chongqing Natural Science Foundation of China under grant cstc2019jcyj-msxmX0747, in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under grant KJQN202000805, in part by the Japan Society for the Promotion of Science (JSPS) Grants-in-Aid for Scientific Research (KAKENHI) under grant JP18K18044, in part by the National Natural Science Foundation of China under grant 61901067, and in part by the Key Research Project of Chongqing Technology and Business University under grant ZDPTTD201917 and grant KFJJ2018071. (*Corresponding author: Keping Yu*)

Zhiwei Guo and Yu Li are with the School of Artificial Intelligence, National Research Base of Intelligent Manufacturing Service, Chongqing Technology and Business University, Chongqing 400067, China (e-mail: zwguo@ctbu.edu.cn; liyu@ctbu.edu.cn).

Keping Yu is with the Global Information and Telecommunication Institute, Waseda University, Shinjuku, Tokyo 169-8050, Japan (e-mail: keping.yu@aoni.waseda.jp).

Gautam Srivastava is with the Department of Computer Science, Brandon University, Brandon, MB R7A 6A9, Canada as well as the Research Centre for Internueral Computing, China Medical University, Taichung, Taiwan (e-mail: srivastavag@brandonu.ca).

Jerry Chun-Wei Lin is with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway (e-mail: jerrylin@ieee.org).

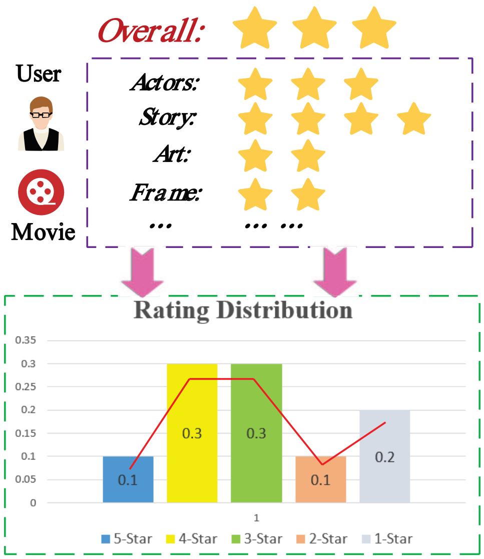



Fig. 1: An example of preference ambiguity and a rating distribution.

components, such as the actors, story, and frame. Users preference degrees regarding different components may be diverse, constituting a preference distribution. Naturally, single rating values cannot clearly reflect comprehensive preference feedback and will surely lead to some noise during modeling.

Fortunately, the Internet of Things (IoT) acquires process information that must be monitored through various sensors, constructing network media that connects people, machines, and things. Due to its resilient ability to perform data collection and arrangement, it has been utilized in many realistic problem scenarios, producing many derivatives such as the industrial IoT, medical IoT, and financial IoT. Hence, it can be introduced to solve the problem in this research area via two steps. First, a social IoT architecture is specifically developed for the issue of social recommendation, guaranteeing a reliable source of training data. It is expected to utilize flexible network plug-ins to replace physical sensors so that multisource information can be acquired more conveniently. On this basis, intelligent algorithms can be embedded into the social IoT architecture. As is shown in Fig. 1, preference feedback on an item can be viewed as the distribution of all the possible rating values, rather than a single rating value. Such multivariate prediction results in a form of rating distribution that can be output by neural network methods. As for users and items, their attributes and internal correlations can be encoded via deep representation schemes, forming fine-grained feature spaces. Thus, this paper proposes a deep learning-embedded social IoT for ambiguity-aware social recommendations (SIoT-SR). This proposal not only guarantees online data sensing and management but also overcomes the preference ambiguity problem in SR. To the best of our knowledge, this research is the first to investigate the ambiguity-aware SR issue and develop a deep learning-embedded social IoT for this purpose. The main contributions of this paper can be summarized as follows:

- It is recognized that existing research on SR still faces two types of difficulties: online data management and preference ambiguity.
- A deep learning-embedded social IoT architecture named SIoT-SR is proposed for ambiguity-aware SR. The social IoT is developed as a base of support and deep learning is introduced to build the core algorithms.
- Experiments are conducted on real-world datasets to evaluate the performance of the proposed SIoT-SR.

The remainder of this paper is organized as follows: Section II introduces the problem scenarios and gives basic definitions. In Section III, the mathematical process of the SIoT-SR is described in detail. The experimental settings, results, and analysis are given in Section IV. We conclude this paper in Section V.

## II. System Model

This research puts forward a specific social IoT to address issues of social recommendation, especially ambiguity-based scenarios. This section first describes the architecture of the designed social IoT and then describes the proposed deep learning-based recommendation algorithm that is embedded into the social IoT.

### A. Architecture of the Social IoT

To provide a resilient environment for data management and scheduling, this research proposes a social IoT whose architecture is illustrated in Fig. 2. The designed social IoT contains four layers: a persistence layer, representation layer, processing layer, and application layer. As different parts of the social IoT, they play different roles in collaboratively implementing social computing tasks. The main roles and effects of these layers are as follows:

- The persistence layer is mainly responsible for some preprocessing work on the source data, including data acquisition, data transmission, and data cleaning. First, the initial data can be from several sources, such as Internet plug-ins and sensing devices. Then, the transmission control protocol is set as the standard for data transmission inside the whole social IoT. After all the data have been stored, this layer should classify them and complete the missing content. Only through these steps can proper data management be realized.
- The representation layer abstracts the whole social network as a hybrid social graph that contains a user subgraph and an item subgraph. The features of the nodes and edges inside both subgraphs can be jointly encoded into two representative vectors separately. As for the nodes, a unified encoding scheme is set up for all the attribute features according to their types. Regarding encoding edges, they are unobserved and need to be modeled by distinguishing different types.
- The processing layer mainly implements the deep learning-based recommendation algorithm that is embedded in the social IoT. It integrates the feature vectors encoded in the representation layer to construct the robust recommendation model. It employs up-to-date source data to continuously update the model parameters so

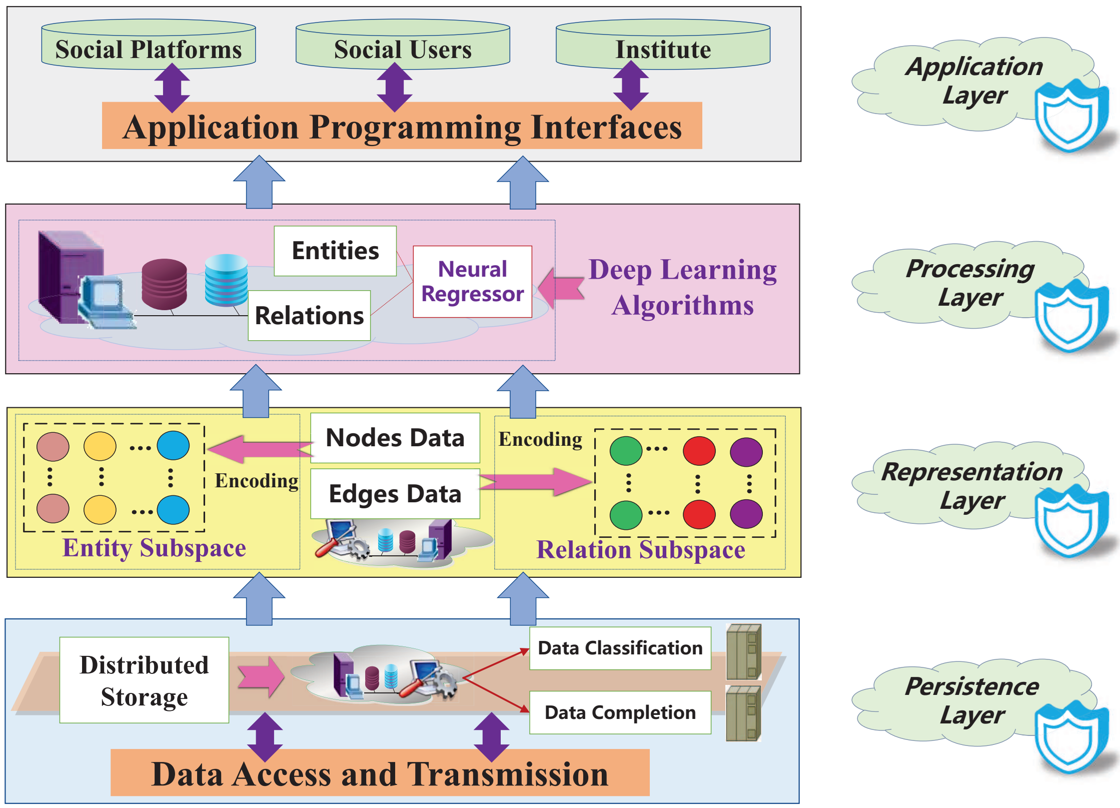



Fig. 2: Architecture of the developed social IoT.

that the recommendation model can fit the time-varying characteristics of social networks. In addition, the recommendation model obtained by training can be used to predict unknown multivariant preference feedback results of users regarding items.

- The application layer connects the core functions of the social IoT to the final users, including platform operators, social users, etc. It provides a content presentation platform for the final users, in which application programming interfaces with unified protocol standards are established to give access to the final users. They can request personalized services or research data with the aid of the proposed social IoT.

The recommendation algorithms are embedded into the middle two layers: the representation layer and processing layer. The next subsection states the research problem and describes the mathematical process in detail.

### *B. Workflow of SIoT-SR*

Let $u_i\,(i=1,2,\cdots,|u|)$ denote the set of $|u|$ users, and let $v_j\,(j=1,2,\cdots,|v|)$ denote the set of $|v|$ items. Each user has the chance to express his preference feedback regarding the items. In this research, preference feedback takes the form of a rating distribution instead of single rating values. In other words, the dimension of preference feedback equals the number of possible rating values. The value of each element is the degree to which the preference feedback is associated with the corresponding rating value. Here, the preference feedback is denoted as $Y_{i,j}^{(\tau)}$, which is a multidimensional vector. With $\tau$ ranging from 1 to $q$, each $Y_{i,j}^{(\tau)}$ is composed of $q$ elements. As the users are not likely to rate all the items, it is certain that the ratings of some items for each user are absent. Given some known preference feedback between the users and items as historical data, this method should predict the unknown preference feedback.

To achieve this, a mapping from features to preference feedback is established with the aid of historical data. Assuming that $D_i$ and $D_j$ denote features of user $u_i$ and item $v_j$, respectively, this process can be denoted as:

$$F\left(D_i, D_j\right) \rightarrow Y_{i,j}^{(\tau)}, \tag{1}$$

where $F\left(\cdot,\cdot\right)$ is a mapping from features to preference feedback and $Y_{i,j}^{(\tau)}$ is a $q$-dimensional vector. Before determining this mapping, all the diverse features must be encoded as representative vectors from the perspective of graph networks. Specifically, the whole social network is abstracted as a hybrid graph network, which is denoted as $\mathcal{G}\left(E,R\right)$. $E$ denotes the features of all the nodes, and $R$ denotes the features of all the edges. $\mathcal{G}\left(E,R\right)$ is composed of a user subgraph $SG\left(u_i\right)$ and an item subgraph $SG\left(v_j\right)$. The rest of this subsection briefly introduces the encoding procedure for $\mathcal{G}\left(E,R\right)$.

For the user subgraph $SG\left(u_i\right)$, the users are the nodes and their relations are the edges. As for the item subgraph $SG\left(v_j\right)$, the items are the nodes, and their relations are the edges. Concatenating the nodes of the two subgraphs into a novel node set and the edges of the two subgraphs into a novel edge set, a new composite graph network $NG(U, G)$ can be formulated. Inside the network, the nodes are mainly features of users and items, and they can be classified into three types according to their forms. A unified encoding principle is proposed for them to transform them into representative vectors. Regarding user $u_i$, all of her feature components are concatenated into a representative vector $E_i$, and all of her relation features are encoded as $R_{i,z}$. $z$ is the index number of all the users except user $u_i$. Similarly, for item $v_j$, all of its feature components are concatenated into a representative vector $E_j$, and all of its relation features are encoded as a representative vector $R_{j,l}$. $l$ is the index number of all the items except item $v_j$. Then, $E_i$ and $E_j$ are concatenated into a novel representative vector $E_{i,j}$, and $R_{i,z}$ and $R_{j,l}$ are concatenated into a novel representative vector $R_{i,j}$. On this basis, a graph neural network (GNN) structure is formulated to obtain a mapping from the features to multivariant preference feedback results. The whole workflow of the proposed SIoT-SR is illustrated in Fig. 3.

## III. Deep Learning-based Social Recommendation

A whole social network can be viewed as a large hybrid graph network including three types of subgraphs: a user-user subgraph, an item-item subgraph, and a user-item subgraph. Inside the former two subgraphs, the users and items are regarded as nodes, and the relations among them are regarded as edges. The latter subgraph is a combination of the other two subgraphs, as the pairs of a user and an item are viewed as nodes, and the relations between a user and an item are viewed as edges. Specifically, for the whole social graph, the nodes and edges of the three subgraphs are aggregated into its nodes and edges. In summary, the whole feature space in this work can be classified into two parts: the entity subspace and relation subspace. The first two subsections below describe the encoding of the feature space implemented by the GNN structure. The third subsection describes the final neural regressor structure, which outputs multivariant preference prediction results in the form of rating distributions.

### A. Encoding of the Node Features

The main goal of this subsection is to model the internal features of all the nodes in the hybrid social graph. According to the different subgraphs, the nodes are divided into two types: user nodes and item nodes. The features of user nodes mainly refer to the personal profiles of users, denoting information on their status and identity. The features of item nodes mainly refer to the internal metadata of items, which reveal the attributes and parameters of items. As the initial profiles cannot be substituted into mathematical models for calculation, a uniform encoding scheme is established that fits the feature contents of different kinds of nodes. Following one of our previously published works, all the profile attributes can be classified into three types: the numerical type, categorical type, and textual type. This section first lists specific feature names for the three types of nodes, then puts forward encoding principles for them and finally concatenates all the encoding components into a total representative vector for node features.

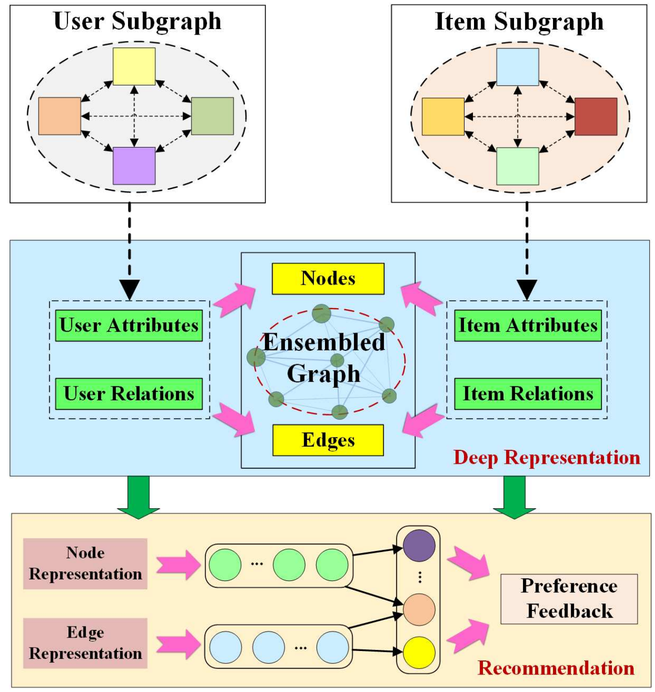



Fig. 3: Workflow of the proposed SIoT-SR.

*1) Feature Overview:* As shown in Fig. 4, eight and ten features are selected for users and items, respectively. Users are relatively universal objects in social networks, as they consume things or services. However, items diversify with changes in the social scenario, as they may be movies, music, news, etc. Here, movies are selected as example items for listing features. Regarding other social scenarios, the corresponding attributes can be changed accordingly. This subsection briefly describes these features in terms of user entities and item entities separately.

For each user entity, the categorical features include authentication status and location, the numerical features include age, registration time, user level, number of ratings, and number of friends, and the textual features include personal tags. As for the item entities, the categorical features include genre, director, 1st actor, country, language, and production company, numerical features include the year, budget, and number of received ratings, and textual features include reviews from users.

*2) Multisource Data Encoding:* Numerical data refers to features or attributes whose contents are numerical values, such as age. They can be employed directly for computation, without excessive processing operations. As the value range of numerical features is flexible, a large gap may exist among the feature values with respect to different entities. To avoid such issues, it is necessary to normalize the values of each numerical feature one by one. In other words, all the numerical values of the different entities need to be mapped to the range $[0, 1]$ through the following process:

$$\bar{\Phi}_{\xi} = \frac{\Phi_{max} - \Phi_{\xi}}{\Phi_{max} - \Phi_{min}}, \tag{2}$$

| Entities | Feature Name | Type | Symbol |
|---|---|---|---|
| Users | Authentication Status | *Categorical* | $A_{u1}$ |
| | Location | *Categorical* | $A_{u2}$ |
| | Age | *Numerical* | $A_{u3}$ |
| | Registration Time | *Numerical* | $A_{u4}$ |
| | User Level | *Numerical* | $A_{u5}$ |
| | Number of Ratings | *Numerical* | $A_{u6}$ |
| | Number of Friends | *Numerical* | $A_{u7}$ |
| | Personal Tags | *Textual* | $A_{u8}$ |
| Items | Genre | *Categorical* | $A_{v1}$ |
| | Director | *Categorical* | $A_{v2}$ |
| | 1st Actor | *Categorical* | $A_{v3}$ |
| | Country | *Categorical* | $A_{v4}$ |
| | Language | *Categorical* | $A_{v5}$ |
| | Production Company | *Categorical* | $A_{v6}$ |
| | Year | *Numerical* | $A_{v7}$ |
| | Budge | *Numerical* | $A_{v8}$ |
| | Number of Received Ratings | *Numerical* | $A_{v9}$ |
| | Reviews | *Textual* | $A_{v10}$ |

Fig. 4: Specification of User Features and Item Features.

where $\xi$ denotes the index number of feature values, $\bar{\Phi}_\xi$ and $\Phi_\xi$ denote the normalized and initial results of the feature values, and $\Phi_{max}$ and $\Phi_{min}$ denote the maximum and minimum of all the feature values.

Categorical data refers to features or attributes whose contents are one or multiple fixed values chosen from multiple options. A typical example of such structured data is the location, as the location a user is in is fixed out of all the possible locations. Structured attributes or features can be represented via one-hot encoding (OHE) which is an encoding scheme that uses binary representations. In detail, all the possible value choices for a feature are initially set to zero, and the zero corresponding to the correct option is changed to one. For a vector obtained by one-hot encoding, the number of possible choices equals its dimension, in which the element corresponding to the correct option is set to one. In summary, the categorical type of data can be represented as vectors composed of several zeros or ones, which can be expressed in the following format:

$$\left[ \underbrace{0, 0, \cdots, 0}_{\text{general options}}, 1 \right]. \tag{3}$$

Textual data refers to unstructured text that can be set arbitrarily by users without structural constraints, such as tags or descriptions. Theoretically, there are unlimited possibilities for the content of such attributes. To transform them into structured types that can be easily encoded, topic indicators need to be introduced to represent unstructured text. As descriptive texts are generally short in social networks, the Twitter-LDA algorithm [32] is adopted to extract a topic indicator for each piece of text. Note that the topic indicators are latent rather than topics with clear meanings such as sports or politics. Assuming that $K$ topic indicators are involved, each piece of text is assigned a topic indicator from among all $K$ ones of them. Then, OHE can be utilized to transform each piece of text into a vector concerning the topic indicators. The dimension of the vector is $K$, and all the $K$ elements are the membership degrees for the corresponding topic indicators. The element concerning the correct topic indicator is set to one, and the other elements are set to zero.

*3) Concatenation:* As shown in Fig. 4, the feature vectors for user entities and item entities can be obtained by concatenating all the feature components. Let $E_i$ and $E_j$ denote feature vectors for user $u_i$ and item $v_j$, respectively. They are represented as the following two formulas:

$$E_i = \left[ E_i^{(1)} \oplus E_i^{(2)} \oplus \cdots \oplus E_i^{(8)} \right] \tag{4}$$

$$E_j = \left[ E_j^{(1)} \oplus E_j^{(2)} \oplus \cdots \oplus E_j^{(10)} \right]. \tag{5}$$

Thus, the representative vector for a pair of entities, user $u_i$ and item $v_j$, is denoted as $E_{i,j}$ and is computed as:

$$E_{i,j} = \sigma_1 \left( W_{E,i} \cdot E_i + W_{E,j} \cdot E_j + b_E \right), \tag{6}$$

where $\sigma_1(\cdot)$ is the sigmoid activation function, $W_{E,i}$ and $W_{E,j}$ are weight parameters that fuse two representative vectors with different dimensions, and $b_E$ is the bias parameter. Thus far, the representative vector for nodes has been built.

## B. Encoding of the Relation Features

The main goal of this subsection is to model external relations among different nodes in a hybrid social graph. From the perspective of subgraph construction, the relations can be divided into two classes: user-user relations and item-item relations. From the perspective of perceptibility, the types of relations can be divided into stable relations and potential relations. Stable relations are those that clearly exist in social networks and can be perceived directly, such as social relationships. Potential relations refer to those that exist latently but cannot be perceived directly. These relations include both stable relations and potential relations. This subsection formulates a vectorized representation for these relations and concatenates them into a total representative vector as the edges of the hybrid social graph.

*1) User-User Relations:* Let $u_z$ denote another user, different from user $u_i$. Enumerating $i$ with $z$ changing from 1 to $|u|$, $R_{i,z}$ denotes the set of relations among all pairs of users. This type of relation contains two parts: one is the relationships among friends that can be observed directly, and the other is the latent relations among users. Therefore, the two parts are represented as $R_{i,z}^{(fri)}$ and $R_{i,z}^{(lat)}$.

The first part is mainly related to the social relationship status between user $u_i$ and user $u_z$. Hence, $R_{i,z}^{(fri)}$ is actually a one-dimensional vector with only one element to reveal the accurate friendship between the users. It is represented as:

$$R_{i,z}^{(fri)} = \begin{cases} [1], & \text{friendship exists} \\ [0], & \text{otherwise} \end{cases} \tag{7}$$

The above formula shows that $R_{i,z}^{(fri)}$ equals the identity vector if a friendship exists between user $u_i$ and user $u_z$ and that it equals the zero vector if a friendship does not exist.

The second part is mainly determined by the feature correlations between user $u_i$ and user $u_z$. The features of the two users are encoded as two vectors, and their similarity is calculated to measure the relevance. Specifically, t he feature vectors for user $u_i$ and user $u_z$ are denoted as $E_i$ and $E_z$. The correlation between them is measured as the following formula:

$$R_{i,z}^{(lat)} = \phi_{i,z} \cdot \|E_i - E_z\|_F^2 , \tag{8}$$

where $\|\cdot\|_F^2$ denotes the second-order Frobenius norm and $\phi_{iz}$ is the relevance weight between user $u_i$ and user $u_z$, computed as [35]:

$$\phi_{i,z} = \frac{\exp\left[\psi_u\left(u_i \& u_z\right)/\psi_u\left(u_z\right)\right]}{\sum\limits_{\substack{\gamma=1 \\ \gamma\neq i,z}}^{|u|} \exp\left[\psi_u\left(u_i \& u_\gamma\right)/\psi_u\left(u_\gamma\right)\right]}, \tag{9}$$

where $\gamma$ is the index number of the users other than users $u_i$ and $u_z$, $\psi_u(u_z)$ and $\psi_u(u_\gamma)$ count the number of items rated by users $u_z$ and $u_\gamma$, $\psi_u(u_i \& u_z)$ counts the number of items commonly rated by users $u_i$ and $u_z$, and $\psi_u(u_i \& u_\gamma)$ counts the number of items commonly rated by users $u_i$ and $u_\gamma$.

Therefore, the final relevance between users $u_i$ and $u_z$ can be obtained through the following operation:

$$R_{i,z} = \left[R_{i,z}^{(fri)} \oplus R_{i,z}^{(lat)}\right]. \tag{10}$$

*2) Item-Item Relations:* Let $v_l$ denote another item, different from item $v_j$. Enumerating $j$ and $l$ from 1 to $|v|$, $R_{j,l}$ denotes the set of relations among all pairs of items. This type of relation is related to two types of factors: the similarity of interactions and the latent relations among items. Neither of them can be observed directly.

The first part is the interaction status between an item and all the users. For a pair of items and users, an interaction occurs between them if the user has ever rated the item and does not occur otherwise. Regarding item $v_j$, its interaction status can be represented via the OHE, yielding a vector $E_j^{(int)}$. This is a $|u|$-dimensional vector whose elements denote the existence of interactions between item $v_j$ and all the $|u|$ users. An element equals 1 if the corresponding interaction exists and 0 if the corresponding interaction does not exist. The second part is mainly determined by the internal features of the items. For item $v_j$, its features are encoded into a vector $E_j$, which was mentioned previously.

Thus, the correlation vector between item $v_j$ and item $v_l$ is calculated as:

$$R_{j,l} = \phi_{jl} \cdot \left\|\left[E_j \oplus E_j^{(\text{int})}\right] - \left[E_l \oplus E_l^{(\text{int})}\right]\right\|_F^2 , \tag{11}$$

where $\oplus$ denotes the concatenation operation and $\phi_{j,l}$ denotes the relevance weight between items $v_j$ and $v_l$. $\phi_{j,l}$ is determined by the similarity of their received ratings. Because the number of received ratings for each item may be different, the ratios of the received ratings are utilized as alternatives to direct ratings. Of all the ratings received for item $v_j$, the distribution ratios of all the possible rating values can be calculated and are denoted as $\varphi_j^{(\tau)}$, where $\tau$ is the index number of possible rating values and ranges from 1 to $q$. Naturally, $q$ is the number of possible ratings. $\phi_{j,l}$ is calculated via cosine similarity as:

$$\phi_{j,l} = \frac{\sum\limits_{\tau=1}^{q}\left[\varphi_j^{(\tau)} \cdot \varphi_l^{(\tau)}\right]}{\sum\limits_{\tau=1}^{q}\sqrt{\left[\varphi_j^{(\tau)}\right]^2} \cdot \sum\limits_{\tau=1}^{q}\sqrt{\left[\varphi_l^{(\tau)}\right]^2}}. \tag{12}$$

Thus far, the correlation vector between items $v_j$ and $v_l$ has been deduced.

*3) Concatenation:* Having calculated the relation vectors inside two subgraphs, the user-user subgraph and item-item subgraph, they are aggregated into representative vectors for all pairs of a user and an item. As for the pair of user $u_i$ and item $v_j$, the total relevance vector for them, $R_{i,j}$, is determined by $R_{i,z}$ and $R_{j,l}$. Enumerating $z$ and $l$ from 1 to their maximum values, the two vectors $R_{i,z}$ and $R_{j,l}$ can be aggregated into two vectors to denote the total relation representations of user $u_i$ and item $v_j$. This aggregation process for the two vectors can be expressed as the following two formulas:

$$R_i = \frac{1}{|u-1|} \cdot \sum_{z=1;z\neq i}^{|u|} \tanh\left(\eta_z\right) \cdot R_{i,z} \tag{13}$$

$$R_j = \frac{1}{|v-1|} \cdot \sum_{l=1;l\neq j}^{|v|} \tanh\left(\eta_l\right) \cdot R_{j,l}, \tag{14}$$

where $\tanh(\eta_z)$ and $\tanh(\eta_l)$ are the attention weights for the users and items. Therefore, $R_{i,j}$ is computed as:

$$R_{i,j} = \sigma_1\left(W_{R,i} \cdot R_i + W_{R,j} \cdot R_j + b_R\right), \tag{15}$$

where $W_{R,i}$ and $W_{R,j}$ are the weight parameters that fuse two representative vectors with different dimensions and $b_R$ is the bias parameter. Hence, the representative vector for the edges of the hybrid social graph $R_{i,j}$ is obtained.

### C. Recommendation

Having deduced representative vectors for the nodes and edges of the hybrid social graph, a regressor of the GNN must be built. However, the two representative vectors obtained are the initial expressions that have not undergone iterative processes. Some updating procedures must be conducted to reoptimize them. Here, this operation is implemented through multiple rounds of crossing iterations between the representative vectors of nodes and those of edges. As the dimension of preference feedback equals $q$, the two initial representative vectors are mapped to $q$-dimensional matrices. The two matrices obtained are the matrices in the initial iterative round and will be updated for a number of rounds. The multilayer perception (MLP) network is introduced for this transformation from vectors to matrices, and it is represented as:

$$ME_{i,j}^{(0)} = Q_E\left(E_{i,j}\right) \tag{16}$$

$$MR_{i,j}^{(0)} = Q_R\left(R_{i,j}\right), \tag{17}$$

where $Q_E(\cdot)$ and $Q_R(\cdot)$ are two different MLPs for $E_{i,j}$ and $R_{i,j}$. Naturally, the effect of the two MLPs can be summarized as extending the dimensions of $E_{i,j}$ and $R_{i,j}$ from one to $q$. Next, this subsection will describe the updating procedures for nodes and edges separately and formulate the objective function for generating multivariant preference results.

During the transition from the $k$-th round to the $(k+1)$-th round, the updating of the representative vector for nodes is expressed by the following formula:

$$ME_{i,j}^{(k+1)} = \sigma_2 \left[ ME_{i,j}^{(k)} + \alpha_{i,j} \cdot H_{i,j}^{(k)} \right], \tag{18}$$

where $\sigma_2(\cdot)$ is the sigmoid activation function, $H_{i,j}^{(k)}$ is the hidden state at the $k$-th round, and $\alpha_{i,j}$ is the transition weight for the pair of user $u_i$ and item $v_j$. The hidden state $H_{i,j}^{(k)}$ can be calculated via:

$$H_{i,j}^{(k)} = \sigma_1 \left\{ W_H \cdot \left[ ME_{i,j}^{(k)} \oplus MR_{i,j}^{(k)} \right] + b_H \right\}, \tag{19}$$

where $W_H$ is the weight parameter and $b_H$ is the bias parameter.

Similarly, the updating of the representative vector for the edges is expressed through the following formulas:

$$MR_{i,j}^{(k+1)} = \sigma_2 \left[ MR_{i,j}^{(k)} + \beta_{i,j} \cdot D_{i,j}^{(k)} \right] \tag{20}$$

$$D_{i,j}^{(k)} = \sigma_1 \left\{ W_D \cdot \left[ ME_{i,j}^{(k)} \cdot U_{i,j} + MR_{i,j}^{(k)} \cdot U'_{i,j} \right] + b_D \right\}, \tag{21}$$

where $U_{i,j}$ and $U'_{i,j}$ are the transition matrices that unify the dimensions of the two vectors $ME_{i,j}^{(k+1)}$ and $MR_{i,j}^{(k+1)}$, $W_H$ is the weight parameter and $b_H$ is the bias parameter.

After all $K$ rounds of updating procedures, two representative matrices $ME_{i,j}^{(K)}$ and $MR_{i,j}^{(K)}$ are obtained. As they both originated from two initial representative vectors, two inverse MLPs must be defined to compress $ME_{i,j}^{(K)}$ and $MR_{i,j}^{(K)}$ into two one-dimensional vectors. The inverse transformation process can be represented as:

$$E_{i,j}^{(K)} = \tilde{Q}_E \left[ ME_{i,j}^{(K)} \right] \tag{22}$$

$$R_{i,j}^{(K)} = \tilde{Q}_R \left[ MR_{i,j}^{(K)} \right], \tag{23}$$

where $\tilde{Q}_E(\cdot)$ and $\tilde{Q}_R(\cdot)$ are two different inverse MLPs for $ME_{i,j}^{(K)}$ and $MR_{i,j}^{(K)}$. They can be utilized directly for outputting the multivariant preference feedback results.

It is assumed that the dimension of the preference feedback results is $q$ and that the index number of each dimension is $\tau$. In other words, $\tau$ ranges from 1 to $q$. As for the pair of user $u_i$ and item $v_j$, the $\tau$-th dimension of the preference feedback results is represented as:

$$\hat{Y}_{i,j}^{(\tau)} = \frac{1}{1 + \exp\left[ T_{i,j}^{(\tau)} \right]} \tag{24}$$

and $T_{i,j}^{(\tau)}$ is computed as:

$$T_{i,j}^{(\tau)} = \lambda_1 \cdot w_1^{(\tau)} \cdot \left[ E_{i,j}^{(K)} \right]^T + \lambda_2 \cdot w_2^{(\tau)} \cdot \left[ R_{i,j}^{(K)} \right]^T + b_1, \tag{25}$$

where $w_1^{(\tau)}$ and $w_2^{(\tau)}$ are the $\tau$-th weight components with respect to the nodes and edges, $b_1$ is the bias parameter, and $\lambda_1$ and $\lambda_2$ are trade-off parameters that sum to 1. Naturally, the above two formulas refine each dimension of the preference feedback results to the range of $(0,1)$. Enumerating $\tau$ from 1 to $q$, the sum of all the $\hat{Y}_{i,j}^{(\tau)}$ values should equal 1. Thus, it is assumed that the normalization operation has been carried out before obtaining $\hat{Y}_{i,j}^{(\tau)}$. In other words, all the elements of $\hat{Y}_{i,j}^{(\tau)}$ satisfy the following condition:

TABLE I: Statistics of the experimental datasets

| Attribute | ***Douban*** | ***Netflix*** |
|---|---|---|
| Number of Users | 12309 | 31052 |
| Number of Items | 17547 | 12463 |
| Number of Ratings | 963514 | 2785357 |
| Number of Social Links | 127531 | 567570 |
| Number of Attributes of Each Item | 10 | 10 |
| Number of Attributes of Each User | 8 | 8 |
| Rating Density | 0.446% | 0.720% |
| Social Density | 0.084% | 0.059% |

$$\sum_{\tau=1}^{q} \hat{Y}_{i,j}^{(\tau)} = 1. \tag{26}$$

The real preference feedback results are denoted as $Y_{i,j}^{(\tau)}$. The learning goal of this work can be summarized as minimizing the distance between $Y_{i,j}^{(\tau)}$ and $\hat{Y}_{i,j}^{(\tau)}$. For the pair of user $u_i$ and item $v_j$, the fundamental objective function for this purpose is formulated as:

$$S_{i,j}^{(\tau)} = \lambda_3 \cdot \left\| \hat{Y}_{i,j}^{(\tau)} - Y_{i,j}^{(\tau)} \right\|_F^2 + \lambda_4 \cdot \left[ \left\| w_1^{(\tau)} \right\|_F^2 + \left\| w_2^{(\tau)} \right\|_F^2 \right], \tag{27}$$

where $\lambda_3$ and $\lambda_4$ are the trade-off parameters used to adjust the weights of the two parts. Similarly, their sum is 1. Extended to all the pairs of users and items, the total objective function is formulated as:

$$\min \sum_{i=1}^{|u|} \sum_{j=1}^{|v|} \left\{ \lambda_5 \cdot \left[ \sum_{\tau=1}^{q} S_{i,j}^{(\tau)} \right] + \lambda_6 \cdot \|\Theta\|_F^2 \right\}, \tag{28}$$

where $\Theta$ denotes the set of all the parameters except $w_1^{(\tau)}$ and $w_2^{(\tau)}$, $\lambda_5$ and $\lambda_6$ are trade-off parameters used to adjust the weights of the two parts, and their sum equals 1. Finally, the stochastic gradient descent (SGD) method [31] is selected as the optimizer to obtain approximate solutions for the above objective function. Let $\Omega$ denote the set of all the parameters, which is expressed as:

$$\Omega = \left\{ \Theta, w_1^{(\tau)}, w_2^{(\tau)} \right\}. \tag{29}$$

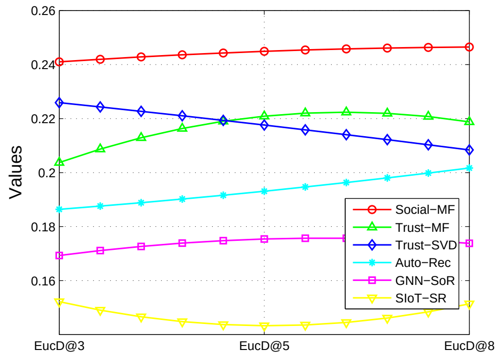
Therefore, the SGD process can be expressed abstractly as:

$$\Omega^{(t+1)} = \Omega^{(t)} - a \cdot \left[ \lambda_5 \cdot \sum_{\tau=1}^{q} \frac{\partial S_{i,j}^{(\tau)}}{\partial \Omega^{(t)}} + 2\lambda_6 \cdot \Theta \right], \tag{30}$$

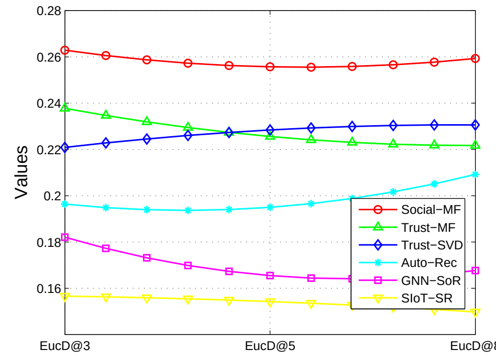
where $t$ is the index of the iterative rounds and $a$ is the learning rate.

## IV. Experiments and Analysis

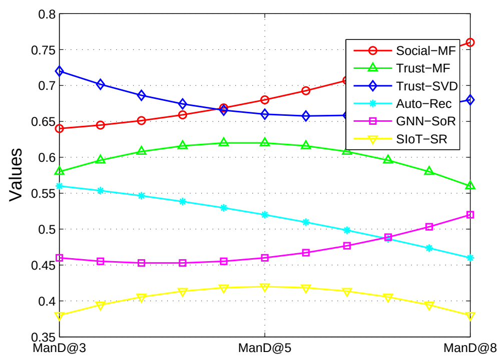
This section presents the detailed process of evaluating the performance of the proposed SIoT-SR on three real-world datasets of social networks.



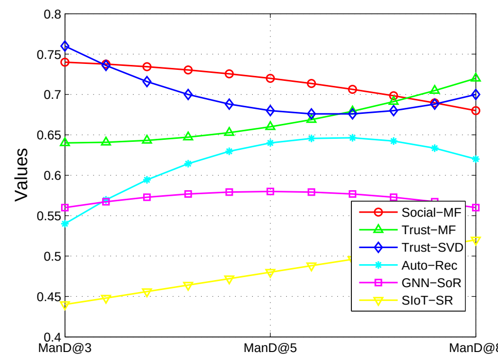

(a) EucD@C results under a proportion of training data of 60%




(b) EucD@C results under a proportion of training data of 70%




(c) ManD@C results under a proportion of training data of 60%




(d) ManD@C results under a proportion of training data of 70%

Fig. 5: EucD@C and Man@C results on the Douban dataset.

TABLE II: CheD@C results under different proportions of training data on the Douban dataset.

| Method | 60% of Data for Training | | | 70% of Data for Training | | |
|---|---|---|---|---|---|---|
| | CheD@3 | CheD@5 | CheD@8 | CheD@3 | CheD@5 | CheD@8 |
| Social-MF | 0.1700 | 0.1600 | 0.1800 | 0.1700 | 0.1400 | 0.1800 |
| Trust-MF | 0.1900 | 0.1800 | 0.1600 | 0.1500 | 0.1200 | 0.1400 |
| Trust-SVD | 0.1200 | 0.1500 | 0.1400 | 0.1300 | 0.1300 | 0.1600 |
| Auto-Rec | 0.1000 | 0.0900 | 0.1200 | 0.1200 | 0.1400 | 0.1300 |
| GNN-SoR | 0.0800 | 0.0800 | **0.0900** | 0.1000 | 0.1100 | 0.1100 |
| SIoT-SR | **0.0700** | **0.0900** | 0.1000 | **0.0800** | **0.0700** | **0.1000** |

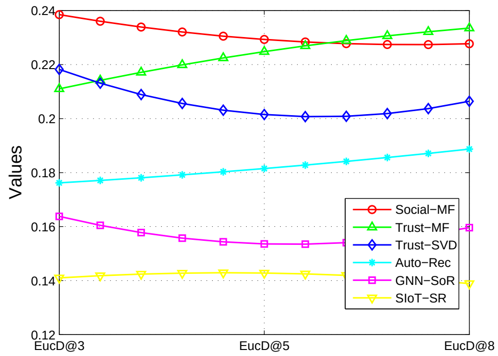
### *A. Datasets*

The construction of the experimental scenarios is derived from two publicly available datasets that are commonly used for such purposes: Douban and Netflix. The initial datasets, as well as some preprocessing operations, are described as follows:

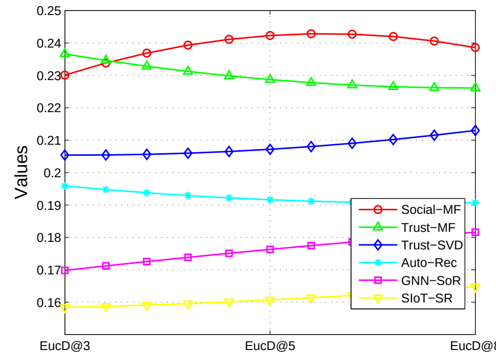
***Douban:*** Douban Movie [1] is a Chinese online community where users can discuss comments and share preferences about movies. The Douban dataset was collected from such websites by researchers to assist in investigating SoR and is updated at least once a year. The dataset contains not only the ratings of users on items but also rich information on the item attributes that can be exploited to generate correlation information. In addition, the dataset contains social relationship information that can be used in scenarios of SoR. To remove useless data, users with fewer than three ratings are filtered out.

***Netflix:*** Founded in 1997 and located in Los Angeles, Netflix is an online video rental provider that mainly provides large numbers of Netflix DVDs and delivers them free of charge. Netflix set up a data mining prize in 2006 and has released approximately one hundred million ratings of users for movies since then. Currently, the Netflix dataset has been the most successful and commonly used in the area of RS. As the full dataset is too large, we filter out items receiving fewer than five ratings and users with fewer than five rating records. To determine the attribute correlations of the items, we follow

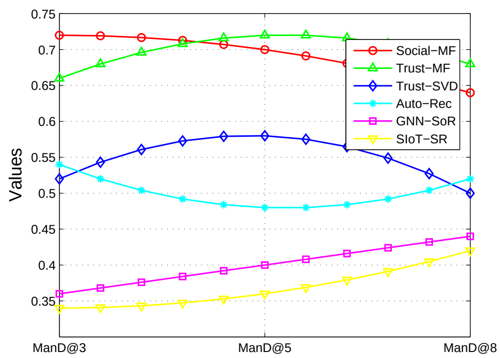
[1] https://movie.douban.com/



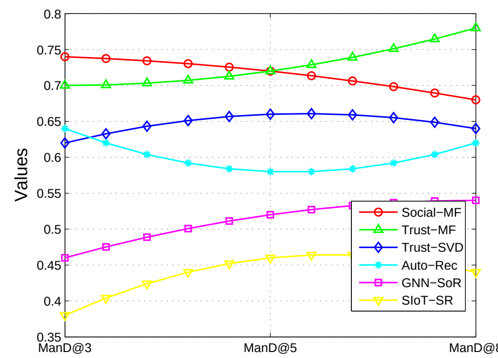

(a) EucD@C results under a proportion of training data of 60%




(b) EucD@C results under a proportion of training data of 70%




(c) ManD@C results under a proportion of training data of 60%




(d) ManD@C results under a proportion of training data of 70%

Fig. 6: EucD@C and Man@C results on the Netflix dataset.

the preprocessing method in [11] and extract metadata from the IMDb website. To address the lack of social information in the Netflix dataset, we randomly assign social relationships to some user pairs with a proportion of approximately 15%.

The statistics of the postprocessed datasets are listed in TABLE I.

### *B. Experimental Settings*

The data of each dataset are divided into two parts: a training set and a testing set. The former set is assumed to be historical records and is thus used for training models. The latter set is viewed as real data occurring in the future and is adopted to testify the efficiency of the recommendation results. The predicted preference feedback is compared with the real preference feedback to measure the effect of the prediction results. For the preference feedback, a major issue that needs to be addressed is how to transform single rating values into rating distributions. Here, a random number-based scheme is proposed to implement this transformation. For each preference feedback, the distribution value of the selected rating value is randomly set to a number drawn from a uniform distribution within the range $[0.7, 0.9]$. Then, the distribution values of the other $(q-1)$ rating values are set randomly, making the sum of all the distribution values equal to 1. Because all the preference feedback takes the form of rating distributions, it measures the distance between the predicted rating distributions and the real rating distributions. To measure the distance between two distributions, four different measurement metrics are introduced, which are briefly described as follows:

*a) Euclidean distance::* This is a universal metric to measure the distance between two multivariant vectors in Euclidean space and is abbreviated as EucD.

*b) Manhattan distance::* This is utilized to indicate the sum of the absolute wheelbases of two points in a standard coordinate system and is abbreviated as ManD.

*c) Chebyshev distance::* This is a metric to measure the distance between two multivariant vectors by computing the maximum value of the absolute value difference, and is abbreviated as CheD.

*d) Correlation distance::* This is a criterion that measures the degree of dependence between two vectors with equal dimensions and is abbreviated as CorD.

*e) Mean absolute error::* This is the average absolute value of the deviation between the predicted vectors and real vectors and is abbreviated as MAE.

*f) Mean-square error::* This is a metric to measure the difference between the predicted vectors and real vectors and is abbreviated as MSE.

The recommendation size, denoted as $C$, refers to the number of items that are recommended to users. The $C$ suggested

TABLE III: CorD@C results under different proportions of training data on the Douban dataset.

| Method | 60% of Data for Training | | | 70% of Data for Training | | |
|---|---|---|---|---|---|---|
| | CorD@3 | CorD@5 | CorD@8 | CorD@3 | CorD@5 | CorD@8 |
| Social-MF | 0.7534 | 0.7217 | 0.7762 | 0.7208 | 0.7469 | 0.7731 |
| Trust-MF | 0.6894 | 0.6623 | 0.7155 | 0.6305 | 0.6604 | 0.6712 |
| Trust-SVD | 0.6028 | 0.6157 | 0.5930 | 0.5619 | 0.5806 | 0.6008 |
| Auto-Rec | 0.4997 | 0.4964 | 0.5101 | 0.4850 | 0.4637 | 0.4505 |
| GNN-SoR | 0.4210 | 0.4878 | 0.4529 | 0.3873 | 0.3951 | 0.4042 |
| SIoT-SR | **0.3146** | **0.3351** | **0.3572** | **0.3016** | **0.3239** | **0.3310** |

TABLE IV: CheD@C results under different proportions of training data on the Netflix dataset.

| Method | 60% of Data for Training | | | 70% of Data for Training | | |
|---|---|---|---|---|---|---|
| | CheD@3 | CheD@5 | CheD@8 | CheD@3 | CheD@5 | CheD@8 |
| Social-MF | 0.1800 | 0.1900 | 0.1700 | 0.1900 | 0.1700 | 0.1600 |
| Trust-MF | 0.1400 | 0.1600 | 0.1700 | 0.1600 | 0.1700 | 0.1500 |
| Trust-SVD | 0.1500 | 0.1300 | 0.1500 | 0.1600 | 0.1500 | 0.1600 |
| Auto-Rec | 0.1100 | 0.1000 | 0.1100 | 0.1300 | 0.1200 | 0.1000 |
| GNN-SoR | 0.0800 | 0.0900 | 0.1000 | 0.0900 | 0.1000 | 0.1000 |
| SIoT-SR | **0.0800** | **0.0700** | **0.0900** | **0.0800** | **0.0600** | **0.0800** |

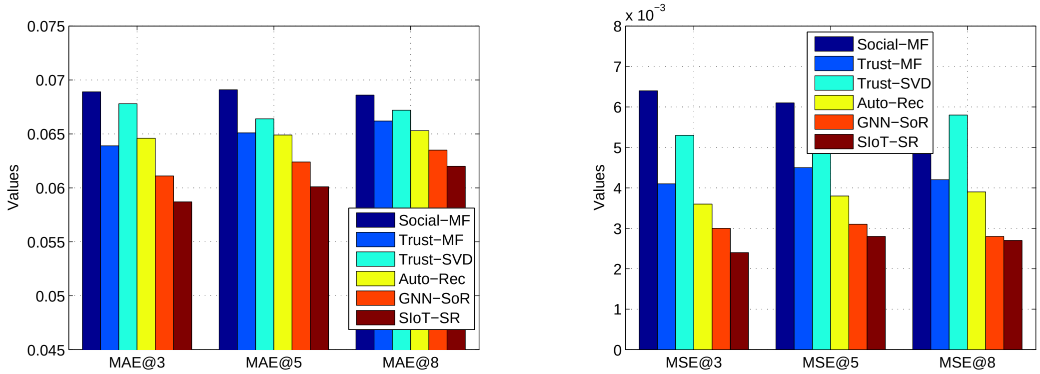
TABLE V: CorD@C results under different proportions of training data on the Netflix dataset.

| Method | 60% of Data for Training | | | 70% of Data for Training | | |
|---|---|---|---|---|---|---|
| | CorD@3 | CorD@5 | CorD@8 | CorD@3 | CorD@5 | CorD@8 |
| Social-MF | 0.6670 | 0.6849 | 0.6937 | 0.6562 | 0.6481 | 0.6869 |
| Trust-MF | 0.5874 | 0.6011 | 0.5995 | 0.5766 | 0.5952 | 0.6103 |
| Trust-SVD | 0.4891 | 0.5251 | 0.5146 | 0.4964 | 0.5058 | 0.5226 |
| Auto-Rec | 0.4270 | 0.4493 | 0.4516 | 0.4885 | 0.5029 | 0.5152 |
| GNN-SoR | 0.3857 | 0.4061 | 0.4058 | 0.4127 | 0.4358 | 0.4261 |
| SIoT-SR | **0.3185** | **0.3362** | **0.3514** | **0.3346** | **0.3762** | **0.3909** |

items are the top $C$ items in the list. This metric is closely related to the other metrics. For example, $MAE@C$ denotes the MAE value obtained when the top $C$ items are recommended to users. To verify the superiority of the proposed SIoT-SR compared to general social recommendation methods, several classical approaches for this purpose are selected as baselines. Similar to one of our previously published studies [10], the first four classical methods for SR are selected as Social-MF [28], Trust-MF [13], Trust-SVD [29] and Auto-Rec [30]. In addition, another GNN-based method that is proposed by our research team, named GNN-SoR [10], is selected as a baseline. Brief descriptions can be found in the corresponding literature. Among the five methods, the first three are based on conventional statistical learning theory, and AutoRec and GNN-SoR utilize deep learning theory to construct models. Note that all the baseline methods address preference feedback only in the form of single values. However, preference feedback takes the form of rating distributions in SIoT-SR. To unify the dimensions of preference feedback, these methods are specifically implemented for preference feedback in the form of rating distributions. In particular, they can be employed to generate all the dimension values one by one. For each piece of preference feedback, normalization operations need to be carried out on all the calculated distribution values, ensuring that their sum equals 1. Then, the predicted rating distributions are compared with the real rating distributions to measure the performance of SIoT-SR.

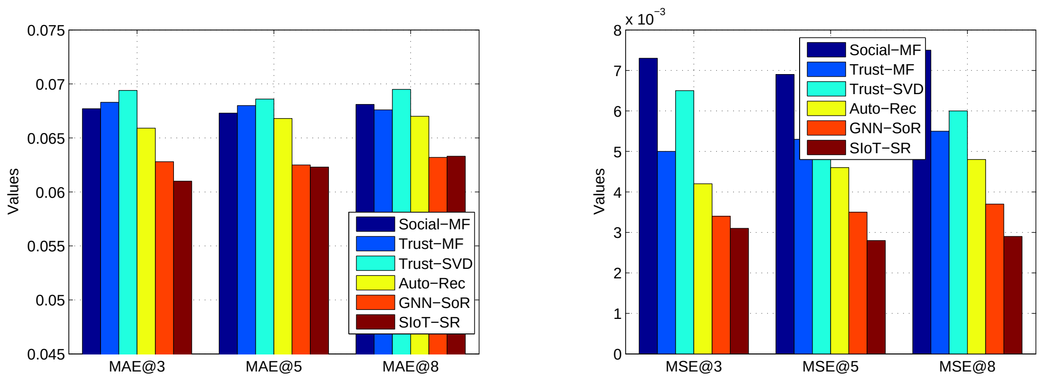
All the experiments are carried out in a deep learning working station with a 28-core CPU, 256-GB RAM, and a GPU (RTX-2080-Ti). The proposed Deep-PR is implemented with the assistance of TensorFlow [2]. The index number $k$ in Eq. (18) ranges from 1 to $K$, which is set as 10 in this work. Initially, $\lambda_1$ and $\lambda_2$ in Eq. (25) are both set to 0.5, $\lambda_3$ and $\lambda_4$ in Eq. (27) are set to 0.6 and 0.4, and $\lambda_5$ and $\lambda_6$ in Eq. (28) are set to 0.55 and 0.45. The learning rate of SIoT-SR is initially set to 0.001 and may be changed multiple times during experiments. As for Twitter-LDA algorithm involved in the SIoT-SR, setting of its major parameters utilize the setting in [32]. The parameters in the baselines are set to their default values and are omitted here due to text limitations. The recommendation size $C$ is set to 3, 5, and 8. Considering the number of recommended results, the proportion of training data is set to 60% and 70%.

[2] http://tensorflow.google.cn/




(a) MAE@C results (b) MSE@C results

Fig. 7: MAE@C and MSE@C results on the Douban dataset.




(a) MAE@C results (b) MSE@C results

Fig. 8: MAE@C and MSE@C results on the Netflix dataset.

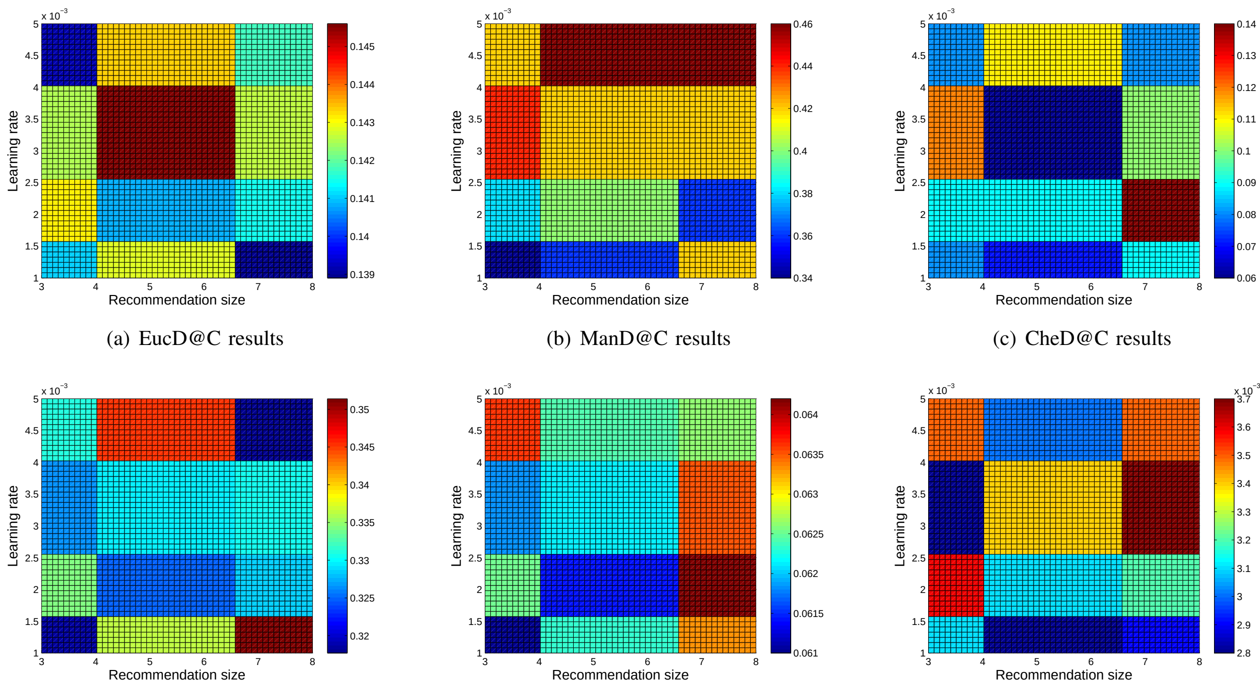
### *C. Results and Analysis*

This subsection presents two groups of experiments to reveal the efficiency and stability of the proposed SIoT-SR.

*1) Efficiency:* When the recommendation size is set to 3, 5, and 8, the EucD@C results and ManD@C results under different recommendation sizes on the two datasets are as shown in Figs. 5 and 6. Both figures contain four subfigures, among which the first two are the EucD@C results concerning the two proportions of training data and the other two are the Man@C results concerning the two proportions of training data. In each subfigure, the X-axis represents the recommendation size $C$ as it changes from 3 to 8, and the Y-axis represents the values of the metrics. The results obtained on the Douban dataset fluctuate greatly, and those on the Netflix dataset fluctuate little. As for the performance comparison, the first two methods perform relatively worse, the next three are relatively better, and the other three deep learning-based methods achieve better results. Of all these approaches, the proposed SIoT-SR always performs better than the baselines. Even compared with two proper baselines, Auto-Rec and GNN-SoR, the proposed SIoT-SR still shows an improvement of at least 10%. The EucD@C results are approximately 10% better than those on the Douban dataset and 15% better than those on the Netflix dataset. The ManD@C results are approximately 12% better than those on the Douban dataset and 18% better than those on the Netflix dataset.

TABLES II and IV list the CheD@C results and CorD@C results on the Douban dataset under different proportions of training data. TABLES III and V list the CheD@C results and CorD@C results on the Netflix dataset under different proportions of training data. Each table has two clusters of data, corresponding to the results obtained under training data proportions of 60% and 70%. Each cluster has six lines that correspond to the results of the six methods and three rows that correspond to the results concerning the three different recommendation sizes. It can be observed from the four tables that the two deep learning-based methods perform better than the other methods and that the proposed SIoT-SR performs better than all the baseline methods. Concerning the CheD@8 results in TABLE II, SIoT-SR performs no better than GNN-SoR, which is a special example. But the overall tendency is that SIoT-SR shows the best performance.

It can be preliminarily concluded from previous experiments that the proportion of training data has little influence on the experimental results. Thus, the proportion of training data




(a) EucD@C results (b) ManD@C results (c) CheD@C results

(d) CorD@C results (e) MAE@C results (f) MSE@C results

Fig. 9: Parameter sensitivity results for the proposed SIoT-SR.

can be fixed at 60% in regard to the metrics MAE@C and MSE@C. Fig. 7 shows the MAE@C results and MSE@C results on the Douban dataset, and Fig. 8 shows the MAE@C results and MSE@C results on the Netflix dataset. Both of them have two subfigures, corresponding to the MAE@C results and MSE@C results, respectively. Each subfigure has three clusters of columns that correspond to the three different recommendation sizes: 3, 5, and 8. It can be naturally observed from these figures that SIoT-SR shows nearly the best performance in terms of all the scenarios of parameter settings. These methods seem to have better performance on the Netflix dataset than the Douban dataset. This phenomenon may be attributed to the fact that the Netflix dataset is a purer dataset and is more suitable for social computing tasks.

There are three reasons for the above results. First, SIoT-SR regards preference feedback as rating distributions rather than single rating values. This is the main novelty that distinguishes it from previous SR methods as well as its main advantage compared with others. Second, a deep representation strategy is introduced into the SIoT-SR to endow it with a stronger feature expression ability. With this property, the recommendation efficiency can be further improved. Thirdly, the whole social network is abstracted as a hybrid social graph, and the newly proposed GNN theory is introduced when establishing feature spaces. This insight can be used to extract more potential and latent relations within social networks so that the feature space in this method can be more fine-grained. The joint effect of the above three features contributes to the improvement of our proposal compared with the baseline methods.

*2) Robustness:* The main purpose of this subsection is to explore the robustness of the proposed SIoT-SR by testing its parameter sensibility. Specifically, when a combination of parameter settings change within certain ranges, the fluctuation tendency of the experimental results is visualized. If the experimental results do not fluctuate with great parameter changes, the corresponding method has proper robustness because it is not susceptible to parameter changes and has good stability. In this work, the combination of parameters is selected as the learning rate and recommendation size. The learning rate is set to four different values: 0.001, 0.002, 0.003 and 0.005. The recommendation size is set to three different values: 3, 5, and 8. Naturally, their combinations yield a total of twelve situations. Of the two datasets, the more representative Netflix dataset is selected as the main experimental environment. The proportion of training data is set to 60%. In addition, only the performance of SIoT-SR itself is evaluated, without comparing it with others. The metrics are the six evaluation distance metrics. The parameter sensitivity results of SIoT-SR are demonstrated in Fig. 9. This figure is composed of six subfigures that correspond to the results of the six evaluation metrics. Inside each subfigure, the X-axis denotes the recommendation size as it changes from 3 to 8, while the Y-axis denotes the learning rate as it changes from 0.001 to 0.005. All the blocks in color display metric values under different situations of the parameter settings. The smaller the chromaticity difference between two blocks, the less the performance of the algorithm is affected by the parameter change. It can be seen from these subfigures that fluctuation of the metric values in each subfigure is quite gentle, demonstrating the proper stability of the proposed SIoT-SR. Fluctuations that are not too great occur with changing parameters. These experimental results can be attributed to two reasons. On the one hand, as rating distributions instead of single rating values are introduced as the forms of preference feedback, more comprehensive preference features can be obtained. Through such operations, the proposed SIoT-SR is not easily influenced by noise or error from ambiguous preference feedback. On the other hand, the whole feature space is divided into two subspaces, and deep representations are adopted to encode them. Through its strong abilities of feature abstraction and extraction, this scheme can produce more robust feature spaces, which in turn promote the stability of the recommendation models. The collaborative effect of the above two reasons leads to better performance.

## V. Conclusion

Social networks have been widely regarded as an indispensable part of human society. Among the most typical problems of social network analysis, social recommendation is well worth investigation. However, current research on SR faces two challenges: a lack of online data management and ignorance of preference ambiguity. Undoubtedly, current circumstances have greatly limited the progress of approaches for SR. To deal with these drawbacks, this paper proposes SIoT-SR, a deep learning-embedded social Internet of Things. First, a social IoT architecture is specifically developed for social computing scenarios to guarantee reliable data management. Then, a deep learning-based graph neural network model that can be embedded into the social IoT is proposed as the core algorithm to obtain ambiguity-aware SR. The design of the social IoT addresses the online data management issue, the utilization of rating distributions as preference feedback provides novel insights for the construction of recommendation models, and deep learning improves the modeling efficiency. Empirically, two real-world datasets are selected to establish experimental scenarios. Six different metrics and five typical methods are selected for assessment. The experimental results show that the proposed SIoT-SR performs better than the benchmark methods by at least 10% and that it has proper parameter sensitivity.

It is also noted that this work has some limitations. In particular, the label space is assumed to be a distribution of all the possible label values. The label enhancement operation brings novel insights regarding recommendation methods but may also lead to some noise. In real-world scenarios, a label space in the form of distributions is not easy to obtain. To deal with this challenge, a label adaptation scheme is naturally required. Additionally, reducing the noise brought by the label adaptation operation is a significant issue. These points are the primary future directions for our research team.

## References


[1] C. Hsu, M. Yeh and S. Lin, "A General Framework for Implicit and Explicit Social Recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 12, pp. 2228-2241, 1 Dec. 2018.

[2] L. Zhen, A. K. Bashir, K. Yu, Y. D. Al-Otaibi, C. H. Foh and P. Xiao, "Energy-Efficient Random Access for LEO Satellite-Assisted 6G Internet of Remote Things", *IEEE Internet Things J.*, doi:10.1109/JIOT.2020.3030856.

[3] Z. Guo, L. Tang, T. Guo, K. Yu, M. Alazab and A. Shalaginov, "Deep Graph neural network-based spammer detection under the perspective of heterogeneous cyberspace," *Future Gener. Comput. Syst.*, vol. 117, pp. 205-218, 2021.

[4] K. Yu, L. Lin, M. Alazab, L. Tan and B. Gu, "Deep Learning-Based Traffic Safety Solution for a Mixture of Autonomous and Manual Vehicles in a 5G-Enabled Intelligent Transportation System," *IEEE Trans. Intell. Transp. Syst.*, doi: 10.1109/TITS.2020.3042504.

[5] X. Zhou, W. Liang, S. Huang and M. Fu, "Social Recommendation With Large-Scale Group Decision-Making for Cyber-Enabled Online Service," *IEEE Trans. Comput. Soc. Syst.*, vol. 6, no. 5, pp. 1073-1082, Oct. 2019.

[6] M. Gao, J. Zhang, J. Yu, J. Li, J. Wen and Q. Xiong, "Recommender systems based on generative adversarial networks: A problem-driven perspective," *Inf. Sci.*, vol. 546, pp. 1166-1185, 2021.

[7] S. Deng, L. Huang, G. Xu, X. Wu and Z. Wu, "On Deep Learning for Trust-Aware Recommendations in Social Networks," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 28, no. 5, pp. 1164-1177, May 2017.

[8] Z. Guo, K. Yu, T. Guo, A. K. Bashir, M. Imran, M. Guizani, "Implicit Feedback-based Group Recommender System for Internet of Things Applications," in *Proc. of 2020 IEEE Global Communications Conferences*, accepted on August 17, 2020.

[9] Z. Zhao et al., "Social-Aware Movie Recommendation via Multimodal Network Learning," *IEEE Trans. Multim.*, vol. 20, no. 2, pp. 430-440, Feb. 2018.

[10] Z. Guo and H. Wang, "A Deep Graph Neural Network-based Mechanism for Social Recommendations," *IEEE Trans. Ind. Informatics*, doi: 10.1109/TII.2020.2986316.

[11] Z. Guo et al., "Robust Spammer Detection Using Collaborative Neural Network in Internet of Thing Applications," in *IEEE Internet Things J.*, doi: 10.1109/JIOT.2020.3003802.

[12] M. Jiang et al., "Social Recommendation with Cross-Domain Transferable Knowledge," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 11, pp. 3084-3097, 1 Nov. 2015.

[13] B. Yang, Y. Lei, J. Liu and W. Li, "Social Collaborative Filtering by Trust," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1633-1647, 1 Aug. 2017.

[14] H. Ma et al., "Recommender systems with social regularization," in *Proc. of the Fourth ACM International Conference on Web Search and Data Mining*, Hong Kong, China, 2011, pp. 287-296.

[15] W. Yao et al., "Modeling dual role preferences for trust-aware recommendation," in *Proc. of the 37th international ACM SIGIR conference on Research & development in information retrieval*, Gold Coast, Queensland, Australia, 2014, pp. 975-978.

[16] Y. Shen, R. Jin, "Learning personal+ social latent factor model for social recommendation," in *Proc. of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, Beijing, China, 2012, pp. 1303-1311.

[17] A. J. B. Chaney, D. M. Blei, T. Eliassi-Rad, "A probabilistic model for using social networks in personalized item recommendation," in *Proc. of the 9th ACM Conference on Recommender Systems*, Vienna, Austria, 2015, pp. 43-50.

[18] H. Fang, Y. Bao, J. Zhang, "Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation," in *Proc. of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, Qubec City, Qubec, Canada, 2014, pp. 30-36.

[19] X. Wang et al., "Learning personalized preference of strong and weak ties for social recommendation," in *Proc. of the 26th International Conference on World Wide Web*, Perth, Australia, 2017, pp. 1601-1610.

[20] Z. Qu et al., "An Efficient Recommendation Framework on Social Media Platforms Based on Deep Learning," in *Proc. of the 2018 IEEE International Conference on Big Data and Smart Computing*, Shanghai, China, 2018, pp. 599-602.

[21] H. Liu et al., "Collaborative social deep learning for celebrity recommendation," *Intelligent Data Analysis*, vol. 22, no.6, pp. 1375-1394, 2018.

[22] L. Xiao, et al., "A Neural Network Model for Social-Aware Recommendation," in *Proc. of the 2017 Asia Information Retrieval Symposium*, Jeju Island, Korea, 2017, pp. 125-137.

[23] O. Tal, Y. Liu, "TCENR: A Hybrid Neural Recommender for Location Based Social Networks," in *Proc. of the 2018 IEEE International Conference on Data Mining Workshops*, Singapore, 2018, pp. 1186-1191.

[24] M. Li, K. Tei, Y. Fukazawa, "An efficient co-Attention Neural Network for Social Recommendation," in *Proc. of the 2019 IEEE/WIC/ACM International Conference on Web Intelligence*, Thessaloniki, Greece, 2019, pp. 34-42.

[25] D. Rafailidis, F. Crestani, "Recommendation with social relationships via deep learning," in *Proc. of the ACM SIGIR International Conference on Theory of Information Retrieval*, Amsterdam, The Netherlands, 2017, pp. 151-158.

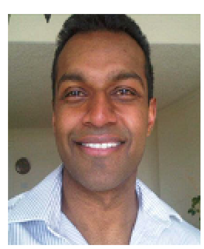
[26] W. Yuan et al., "Socialized healthcare service recommendation using deep learning," *Neural Computing and Applications*, vol. 30, no.7, pp. 2071-2082, 2018.

[27] W. Fan et al., "Graph Neural Networks for Social Recommendation," inemphProc. of the 28th World Wide Web Conference, San Francisco, CA, USA, 2019, pp. 417-426.
[28] M. Jamali, M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proc. of the fourth ACM conference on Recommender systems*, Barcelona, Spain, 2010, pp. 135-142.
[29] G. Guo, J. Zhang, N. Yorke-Smith, "TrustSVD: collaborative filtering with both the explicit and implicit influence of user trust and of item ratings," in *Proc. of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, Austin, Texas, USA, 2015, pp. 123-129.
[30] C. Zhang et al., "Collaborative User Network Embedding for Social Recommender Systems," in *Proc. of the 2017 SIAM International Conference on Data Mining*, Houston, Texas, USA, 2017, pp. 381-389.
[31] Y. Liu, W. Huangfu, H. Zhang and K. Long, "An Efficient Stochastic Gradient Descent Algorithm to Maximize the Coverage of Cellular Networks," *IEEE Trans. Wirel. Commun.*, vol. 18, no. 7, pp. 3424-3436, July 2019.
[32] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan and X. Li, "Comparing twitter and traditional media using topic models," *in Proc. of the 33rd European Conference on Advances in Information Retrieval*, Dublin, Ireland, pp. 338349, 2011.
[33] Y. Gong, L. Zhang, R. Liu, K. Yu, and G. Srivastava, "Non-linear MIMO for Industrial Internet of Things in Cyber-Physical Systems", *IEEE Trans. Ind. Informatics*, doi:10.1109/TII.2020.3024631.
[34] C. Feng, K. Yu, M. Aloqaily, M. Alazab, Z. Lv and S. Mumtaz, "Attribute-Based Encryption with Parallel Outsourced Decryption for Edge Intelligent IoV," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13784-13795, Nov. 2020.
[35] Z. Guo, K. Yu, A. Jolfaei, A. K. Bashir, A. O. Almagrabi and N. Kumar, "A Fuzzy Detection System for Rumors through Explainable Adaptive Learning," *IEEE Trans. Fuzzy Syst.*, accepted.

**Zhiwei Guo** received the B.E. degree from Zhengzhou University, Zhengzhou, China, in 2013, and the Ph.D. degree from Chongqing University, Chongqing, China, in 2018. He is currently an Associate Professor with Chongqing Technology and Business University, Chongqing, China. In 2017, he gave an oral presentation in the International Joint Conference on Artificial Intelligence (IJCAI 2017). His research interests focus on applications of data mining and pattern recognition in the Internet of Things.

**Keping Yu** (S11-M17) received the M.E. and Ph.D. degrees from the Graduate School of Global Information and Telecommunication Studies, Waseda University, Tokyo, Japan, in 2012 and 2016, respectively. He was a Research Associate and a Junior Researcher with the Global Information and Telecommunication Institute, Waseda University, from 2015 to 2019 and 2019 to 2020, respectively, where he is currently a Researcher. He is also a Visiting Professor with the College of Computer Science, Sichuan Normal University, Chengdu, China. He has hosted and participated in more than ten projects, is involved in many standardization activities organized by ITU-T and ICNRG of IRTF, and has contributed to ITU-T Standards Y.3071 and Supplement 35. His research interests include smart grids, information-centric networking, the Internet of Things, blockchain, and information security. Dr. Yu has been the general co-chair and publicity co-chair of the IEEE VTC2020-Spring 1st EBTSRA workshop, general co-chair of IEEE ICCC2020 2nd EBTSRA workshop, TPC co-chair of SCML2020, local chair of MONAMI 2020, Session Co-chair of CcS2020, and session chair of ITU Kaleidoscope 2016. Moreover, he has served as TPC member of more than ten international conferences, including the ITU Kaleidoscope, IEEE VTC, IEEE CCNC, IEEE WCNC, etc. He has been a Lead Guest Editor for Sensors, Peer-to-Peer Networking and Applications, Energies and Guest Editor for IEICE Transactions on Information and Systems, IET Intelligent Transport Systems, Computer Communications, Wireless Communications and Mobile Computing. He is an Editor of the IEEE OPEN JOURNAL OF VEHICULAR TECHNOLOGY (OJVT).

**Yu Li** received the B.E., M.E. and Ph.D. degrees from Chongqing University of Posts and Telecommunications, in 2011, 2014 and 2018, respectively. She is currently a lecturer with the Chongqing Technology and Business University. From December 2015 to December 2016, she was with the University of MassachusettsDartmouth, Dartmouth, MA, USA, as a Visiting Ph.D. Student. Her research interests include social networks and device-to-device communications.



**Gautam Srivastava** (Senior Member, IEEE) received the B.Sc. degree from Briar Cliff University, USA, in 2004, and the M.Sc. and Ph.D. degrees from the University of Victoria, Victoria, BC, Canada, in 2006 and 2011, respectively. He then taught for three years at the Department of Computer Science, University of Victoria, where he was regarded as one of the top undergraduate professors in the Computer Science Course Instruction at the University. In 2014, he joined a tenure-track position at Brandon University, Brandon, MB, Canada, where he currently is active in various professional and scholarly activities. He was promoted to the rank an Associate Professor in January 2018. He is popularly known as an active in research in the field of data mining and big data. In his eight-year academic career, he has published a total of 60 articles in high-impact conferences in many countries and in high-status journals (SCI and SCIE) and has also delivered invited guest lectures on big data, cloud computing, Internet of Things, and cryptography at some Czech universities. He received the Best Oral Presenter Award in FSDM 2017. He currently has active research projects with other academics in Singapore, Canada, Czech Republic, Poland, and USA. He is constantly looking for collaboration opportunities with foreign professors and students. He is an Editor of several international scientific research journals.

**Jerry Chun-Wei Lin** (Senior Member, IEEE) is currently a Full Professor with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway.,Dr. Lin has published more than 300 research articles in refereed journals (135 SCI articles, the IEEE Transactions on Knowledge and Data Engineering, the IEEE Transactions on Cybernetics, ACM TDS, and ACM TKDD) and international conferences. He has filed and held 33 patents including three U.S. patents. He is the co-leader of the well-known SPMF project, and also the founder and the leader of PPSF project. Moreover, he has been awarded as the Most Cited Chinese Researcher in 2018 by Elsevier. He is the IET Fellow, and ACM Senior Member. He is the Editor-in-Chief of Data Science and Pattern Recognition (DSPR) journal, an Editor or Associate Editor of the Journal of Network Intelligence, the Journal of Internet Technology, IEEE Access, the Journal of Network Intelligence, the International Journal of Interactive Multimedia and Artificial Intelligence, Intelligent Data Analysis, and the International Journal of Information Technology, Communications and Convergence. He has served as a TPC for more than 200 international conferences, and served as the Guest Editor for many SCI journals, such as the IEEE Transactions on Fuzzy Systems, Applied Sciences, Sensors, and IEEE Access.