Graph-based service recommendation in Social Internet of Things

Yuanyi Chen, Yanyun Tao, Zengwei Zheng and Dan Chen

Abstract
While it is well understood that the emerging Social Internet of Things offers the capability of effectively integrating and managing massive heterogeneous IoT objects, it also presents new challenges for suggesting useful objects with certain service for users due to complex relationships in Social Internet of Things, such as user’s object usage pattern and various social relationships among Social Internet of Things objects. In this study, we focus on the problem of service recommendation in Social Internet of Things, which is very important for many applications such as urban computing, smart cities, and health care. We propose a graph-based service recommendation framework by jointly considering social relationships of heterogeneous objects in Social Internet of Things and user’s preferences. More exactly, we learn user’s preference from his or her object usage events with a latent variable model. Then, we model users, objects, and their relationships with a knowledge graph and regard Social Internet of Things service recommendation as a knowledge graph completion problem, where the “like” property that connects users to services needs to be predicted. To demonstrate the utility of the proposed model, we have built a Social Internet of Things testbed to validate our approach and the experimental results demonstrate its feasibility and effectiveness.

Keywords
Graph embedding, object usage event, service recommendation, Social Internet of Things

Date received: 18 February 2021; accepted: 23 February 2021

Handling Editor: Yanjiao Chen

Introduction
With the rapid development and popularization of low-cost embedded sensing devices and mobile computing technologies, numerous physical things are being interconnected to form an Internet of Things (IoT). The physical things are embedded with sensing, communication, and computing capabilities, such as microwaves, furniture, and smart electromechanical devices. The IoT will gradually transform people’s work and living environment into an intelligent space, providing the ability to integrate the physical world and the cyber world, and also promote the emergence of many new applications (e.g. things search, semantic annotation, anomaly detection, and indoor localization). While the IoT promises to be a source of great benefits to our lives, it still faces a few significant challenges with increasing heterogeneous things participate in sensing and communicating, such as effectively sensing and transmitting data, safely discovering services and objects based on user’s preference, and context-aware browsing and caching. For example, numerous smart objects are required to register their information in a centralized server before searching, and the search...
process cannot benefit from the information about the objects’ context relationships. Obviously, these challenges caused by the extremely high complexity of the IoT environments cannot be solved by even very smart objects singularly. Recently, a few studies\(^9,10\) proposed a new paradigm named Social Internet of Things (SIoT) by integrating social networks and IoT, which applies the theory of social networks into different levels of the IoT and provides new opportunities to address these challenges. Given the increasing number of heterogeneous IoT objects, the SIoT make the world of trillions of objects manageable when facing service and information discovery and lay the ground for autonomous interactions among objects for the benefit of the human user, thus can facilitate a few valuable services,\(^1\) such as (1) are able to interact with other objects autonomously with respect to the owners; (2) can easily crawl the IoT made of billions of objects to discover services and information in a trust-oriented way; and (3) can advertise their presence to provide services to the rest of the network.

SIoT adopts a service-oriented architecture where heterogeneous IoT objects can offer or request autonomous services and collaborate in terms of their relationships. Therefore, recommending useful objects with specific services to better meet user’s requests in SIoT environment is a fundamental research challenge for many applications (e.g. urban computing, smart cities, and health care), and attracts more and more attention.\(^12,13\) However, SIoT service recommendation is more spontaneous, context-dependence, and monopolistic compared with traditional web service recommendations due to a few unique characteristics:

- **Complex social relationships:** According to Atzori et al.\(^9,11\) the SIoT includes five basic social relationships, that is, parental object relationship, co-location object relationship, co-work object relationship, ownership object relationship, and social object relationship. Without considering these complex social relationships and interdependencies among SIoT, objects may result in inappropriate recommendation results.
- **Heterogeneous SIoT objects:** In SIoT, objects are heterogeneous in terms of functionality, and attributes thus cannot be represented in a uniform feature space. Besides, the text descriptions of objects are usually incompletely. As a result, both the feature-based method and text-based method cannot be utilized to measure an object’s similarity or find an object with a specific service when recommending SIoT service.
- **Implicit user preferences:** For a traditional recommendation system such as recommending clothes or music, the usefulness of a product or service is generally represented by a “rating,” which indicates how much a given user likes a particular item. Thus, content-based filtering and collaborative filtering based on the user’s rating can be used to recommend clothes or music. However, user’s ratings or preference for services are usually implicit in SIoT, which is a major challenge for SIoT service recommendation.

In light of the preceding challenges, we present a graph-based service recommendation framework to address the service recommendation problem in SIoT. As shown in Figure 1, to fuse information from heterogeneous SIoT objects and their complex social relationships, we represent users and SIoT objects as entities of a knowledge graph. Then we model the SIoT service recommendation problem as a knowledge graph completion problem, where the “like” property depends on the user’s preference, which connects users to a specific service that they like, has to be predicted. However, unlike e-commerce sites or location-based social network, the “like” property between users and objects is implicit is SIoT. Fortunately, user’s object usage events can be easily recorded and obtained using ubiquitous sensing technologies, such as Radio-frequency identification (RFID) and sensor readings. These object usage events, which embedded with rich spatial-temporal information and implicitly imply the regularities of users, provide rich information to uncover user’s preference.\(^10,14,15\) For example, how frequently and how long a user uses an object with a specific service can reflect their preference to some extent. For predicting missing “like” relations in knowledge graph, existing graph-based methods\(^16,17\) may suffer from robustness caused by the incompleteness of the graphs. To address these challenges, we utilize recent advances in graph embedding learning techniques and propose a novel framework Specifically, we first jointly embed users, objects, services, and their corresponding relations into a shared lower-dimensional space. Then, we decompose the SIoT service recommendation into a link prediction process while considering the user’s preference and object’s social relationships. In a nutshell, the contributions of our research are three-fold:

- We uncover the “like” property between user and service by modeling the user’s preference toward an object as the hidden factor of his or her object usage events with a latent variable model.
- We construct a knowledge graph to capture complex social relationships of SIoT objects and the “like” property between users and services. Then we model the SIoT service recommendation problem as a knowledge graph completion problem and propose a graph embedding-based SIoT service recommendation approach.
We evaluate our approach based on three real-world data sets and a synthetic data set, which are collected by 242 SIoT objects and more than 500 users. The results show the advantages of our approach beyond multiple baseline algorithms.

The remainder of the article is organized as follows: section “Related work” surveys related work on service recommendation in SIoT. Section “Problem definition” presents the problem definition of SIoT service recommendation. Section “The proposed service recommendation model in SIoT” describes the proposed service recommendation model in SIoT. Section “Experiment evaluation” reports and discusses the experimental results. Finally, we present our conclusion and future work in section “Conclusion.”

Related work

In this section, we first survey related work on service recommendation in SIoT and then discuss how these works differ from our work.

A few studies have been proposed to perform SIoT service recommendation using semantic web technologies. The idea of semantic-based service recommendation is that first define a metadata model to describe characteristics (e.g. geophysical, functional, and non-functional) of an object, then find services that meet the user’s requirement via semantic similarity computation. For example, Rasch and Hussein et al. presented smart assistance via recommending services in smart homes by considering user’s context information (e.g. current situation, requirements, preferences, and habits). Through exploiting user’s interests and preferences from multi-modal information (e.g. travel sequence, actions, motivations, opinions, and demographic information), the study proposed a health-centric recommendation system in IoT based on the ontological knowledge base and tailored filtering mechanisms.

To recommend services among various SIoT applications, AGILE proposed two recommendation models for IoT edge devices with specific services, which utilize collaborative recommendation approach for suggesting additional applications for the current gateway and content-based recommendation. However, the proposed recommendation framework needs to figure out the contexts it makes sense to provide basic recommendation functionalities for the gateway. Based on RESTful principles and ontologies, Beltran et al. first proposed a system to manage SIoT service’s profile and interconnect the service with the rest of the knowledge database, then recommended possible automatic actions to users using the service’s ontology database and actual data. Mashal et al. introduced an undirected tripartite graph–based model with weight spreading to recommend services in IoT, which generates recommendation list by jointly considering information of services and ambient resource. While these semantic-based service recommendation models are effective in scenarios that are pre-programmed by domain experts, they will suffer a few limitations in SIoT environment, such as (1) new services need to be discovered across diverse and rapidly changing environments; (2) the challenge to model the usage and temporal changes of a SIoT service using predefined information.

To model the usage and temporal changes of an SIoT service, several literatures focus on SIoT service recommendation based on the user–object interaction behaviors. SIoT objects with specific services are discrete without explicit connection with users, but users and objects will interact in daily activities, and these interactions can provide rich information (e.g. activity, location, and time) for uncovering the regularities of users. For instance, Jeong et al. proposed a rule-based service recommendation system by analyzing both object’s usage patterns of a user and the common usage patterns of the user who is living in a similar context. However, the model lacks quality of services such as the service price and trust value. To combine object’s properties when making service recommendation,
Wang et al.\textsuperscript{26} proposed an IoT service recommendation model based on mass diffusion dynamic tag tripartite graph, which constructs the tripartite graph by extracting user’s habit features by regarding the SIoT service as the dynamic tag. The recommendation list is generated by first calculating the resource value of the service with positive, then performs backward mass diffusion on the tripartite graph, and finally jointly measures the energy state, the user similarity, the bearing capacity and the service resource value of the mass diffusion results. For modeling various interaction relationships in IoT entities, Yao et al.\textsuperscript{15} further proposed a unified probabilistic factor–based framework for IoT recommendation by fusing relations across heterogeneous entities of IoT, such as user–object relations, user–user relations, and object–object relations. However, these methods generate SIoT service recommendation without considering various social relationships in SIoT, which is important for selecting services to better meet user’s requirement.

To our best knowledge, only few studies\textsuperscript{27–29} addressed the problem of service recommendation in SIoT with considering object’s social relationships. For example, Saleem et al.\textsuperscript{27} utilized people–objects relationship and objects–objects relationship in SIoT to recommend services among various IoT applications. For QoS-aware service recommendation for IoT Mashup applications, Cao et al.\textsuperscript{28} captured the relationships among mashup, services, and their links with relational topic model and exploited factorization machines to train the latent topics for recommending services. Since user groups’ preferences for IoT-based services differ significantly from individual users, Lee et al.\textsuperscript{29} proposed a service recommendation using user-based collaborative filtering, which determines neighbor user groups by considering several member organization–based group similarity metrics such as the group size–based, common member–based, and member preference–based metrics. Khelloufi et al.\textsuperscript{30} proposed a service recommendation system by considering the social relationships between device owners, as incorporating user’s social features can enable context-aware recommendation and increase the efficiency of the recommended services. To fuse multiple types of service data and their logical relations for personalized service recommendation methods, Wang et al.\textsuperscript{31} proposed domain-oriented user and service interaction knowledge graph. As privacy-preserving schemes are very important for service recommendation in SIoT, Ferrag et al.\textsuperscript{32} surveyed recent research literatures on research trends in privacy-preserving schemes for ad hoc social networks. However, user’s ratings or preference for SIoT services are usually implicit, thus traditional recommendation models (such as content-based filtering and collaborative filtering) based on user’s rating or preference cannot be used to service recommendation in SIoT.

Our proposed approach differs from existing works in the following two aspects: (1) we consider that both the usage frequency and usage duration of an object to characterize user’s preference, then model the “like” property between user and service with a latent variable model and (2) we perform service recommendation in SIoT by using a knowledge graph to capture complex social relationships of heterogeneous SIoT objects and user’s “like” property.

**Problem definition**

In this section, we first define the key data structures and notations then present the problem definition of service recommendation in SIoT.

**Object**

$oi$ denote with a tuple $<ID_i, S_i>$, where $ID_i$ is the identifier and $S_i = \{s_1', ..., s_j', ...\}$ is the service set.

**Example 1.** Considering an object $o$ named multi-functional coffee machine (https://item.jd.com/42882380807.html/) (as shown in Figure 2), which is a physical machine but embedded in several sensors (e.g. water flow sensor, temperature, and built-in WiFi). For this smart object, its service set $S_o$ consists of $<s_1$: making coffee $>$, $<s_2$: making juice $>$, $<s_3$: making milk tea $>$, $<s_4$: making soy milk $>$. According to the object’s definition, we denote this object $o$ as $<192.168.1.111, S_o>$ using the IP address as object’s identifier.

**Object usage event**

An object usage event between $u_i$ and $o_j$, denote by $y \in Y^{(ij)} = \{y_1^{(ij)}, y_2^{(ij)}, ..., y_H^{(ij)}\} = \{<u_i, o_j, s_{ij}, t_s, t_e>|u_i, o_j, s_{ij}, u_i \in U \land o_j \in O \land s_{ij} \in S \land t_s, t_e \in TS\}$, indicates that a user $u_i$ used service $s_{ij}$ of object $o_j$ from timestamp $t_s$ to timestamp $t_e$, where $U, O, TS$ denote the set of users, objects, and timestamps, respectively.

**User-object preference**

Denote by an adjacent matrix $I$, where $I^{(ij)}$ indicates the preference strength of user $u_i$ for object $o_j$.

**SIoT knowledge graph**

Denote by $G = \{E, R, A\}$ where $E$ is the set of entities, $R \subseteq E \times \mathbb{R} \times E$ is a set of typed relations among entities, and $A$ is an ontology that denotes the set of relation types, that is, properties $\mathbb{R}$. Entities include users
Observed positive feedback between a user and a service is described by a special property, which we name “like.” We finally add the “like” property as triples \((u_i, \text{like}, s_j)\) if the preference strength of user \(u_i\) for service \(s_j\) is larger than a threshold.

With the aforementioned definitions, the problem of service recommendation in SIoT can be formally stated as follows:

Giving a set of objects and the service set that objects can provide:

\[
O = \{<ID_1, S_1>, <ID_2, S_2>, \ldots\},
\]

and the history object usage events of users:

\[
Y^{(u)} = \{y^{(u)}_1, y^{(u)}_2, \ldots, y^{(u)}_H\},
\]

the problem of service recommendation in SIoT aims to find a ranking function \(r(u_i, s_j)\) that assigns a score to any user–service pair \((u_i, s_j) \in U \times S\) and sort the service according to \(r(u_i, s_j)\), then generate a set of \(k\) recommendation services \(S_{\text{recommend}}\) according to the rank score.

The proposed service recommendation model in SIoT

In this study, we regard service recommendation in SIoT as a knowledge graph completion problem by modeling users, objects, and services as entities of a knowledge graph, where the “like” property between users and services has to be predicted. As shown in Figure 3, our proposed recommendation framework produces top-\(k\) recommended services by two phases: (1) learning user preference from object usage event; (2) service recommendation via graph embedding (details of service recommendation will be introduced in section “Experiment evaluation”). The phase produces top-\(k\) recommended services by solving knowledge graph completion problem based on graph embedding.

### Learning user preference from object usage event

The idea behind our approach is that users’ object usage events imply their preference, as most people have a finite amount of resources, they tend to select service from preferred service providers (i.e. objects). Thus, we learn the user’s preference by the following two steps.

1. **Collect user’s object usage event with ubiquitous sensing technologies.** User’s object usage events, which embed with rich spatial-temporal information and implicitly imply behavior habits of a user, provide us a new approach to uncover his or her preference. In addition, object usage events are easy to record and collect using ubiquitous sensing technologies (such as state-change sensors) in SIoT. In our experiment, we utilize state-change sensors to identify and record usage information.

2. **Learn user’s preference.**
of service providers and further develop a context-aware service (CoAE) application for recording users’ service usage records. During the study, each participant was given an Android device to run CoAE application and uses the application to record the service what he or she is doing, such as the start and end time of this event.

Figure 4 shows an example of collecting object usage event using the state-change sensors and CoAE application. As shown in Figure 4(a), the starting and ending time of service “make coffee” is recorded by CoAE application. We further extract the participated objects of the service by observing the sensor activations during the service duration (as shown in Figure 4(b)). Finally, we observe the service is completed with five service providers: filter cup, microwave, coffee machine, electronic scale, and kettle, thus consider there is an object usage event between user and each object during this event.

**Mining user-service preference via latent variable model.** The principle of mining User-Service preference is that homophily is ubiquitous in SIoT, which suggests the likelihood that an object usage event will take place relate positively to user’s preference to the object.33 Clearly, users have a finite amount of resources, they tend to select their favorite service providers. In this way, we model user’s preference toward an object as the hidden factor of his or her object usage event with a latent variable model, details of the modeling process can be found in our previous work.3,14

**Generate recommendation list via graph embedding**

After modeling users’ service preference from object usage event, we combine the learned user preference and various social relationships among SIoT objects to construct a knowledge graph. Then we formulate the SIoT service recommendation problem as one of knowledge graph completion problem, where the “like” property between users and services needs to be predicted. For predicting missing “like” property in the SIoT knowledge graph, we map entities and various social relationships among SIoT objects to a low-dimensional semantic space by graph embedding.

**SIoT knowledge graph embedding.** As mentioned earlier, SIoT knowledge graph \( G = \{E, R, A\} \) is a multi-relational graph, entities \( E \) and relations \( R \) contain different types. For the “like” property between users and services, we add triples \((u, like, s)\) to train data set based on a threshold-based method, that is, the user’s service preference \( I(u, s) \) is larger than a threshold. We use bold letters \( h, r, t \) to represent the corresponding embedding representations of a triple \((h, l, t)\) to train data set.

As mentioned earlier, SIoT knowledge graph \( G \) is a multi-relational graph, entities \( E \) and relations \( R \) contain different types. For the “like” property between users and services, we add triples \((u, like, s)\) to train data set based on a threshold-based method, that is, the user’s service preference \( f(u,s) \) is larger than a threshold. We use bold letters \( h, r, t \) to represent the corresponding embedding representations of a triple \((h, l, t)\) to train data set.

The model parameters of TransH are as follows: (1) \( e_{i}^{\parallel} \), which represent all the entities’ embeddings; (2) \( (w_{r}, d_{r})_{r=1}^{\mid R\mid} \), which represent all the relations’
hyperplanes and translation vectors. To learn these model parameters, TransH exploits margin-based ranking loss as objective function to distinguish golden triplets and incorrect triplets as follows:

$$\Gamma = \sum_{(h, r, t) \in \Delta^+} \sum_{(h', r', t') \in \Delta^-} [f_r(h, t) + \gamma - f_r(h', t')]_+$$  \hspace{1cm} (2)

where $[x]_+ = \max(0, x)$ denotes the positive part of $x$, $\Delta^+$ is the set of positive triplets, $\Delta^-$ denotes the set of negative triplets constructed by corrupting $(h, r, t)$, $\gamma$ is the margin separating positive and negative triplets. The minimum value of this loss function can be obtained with stochastic gradient descent by randomly visiting the triplets from the knowledge graph multiple times.

**Service recommendation process.** In this subsection, we present how to recommend SIoT service based on the learned embeddings entities and relations of SIoT knowledge graph. The principle is using the negative score assigned to a triple $f(u, \text{like}, s)$ as the ranking function $r(u, s)$.

More exactly, we first split users’ triples set $D_u$ into a $D^p_u$ and $D^g_u$. Then, we learn the SIoT knowledge graph embeddings from $G$ and obtain the vector representations of each $e \in E$ and $r \in R$ (e.g. the “like” property). For users in test set $u \in U_{te}$, we generate recommend SIoT services for $u$ by sort all service candidates $s \in S_{\text{cand}}$ according to the ranking score $r(u, s) = -f(u, \text{like}, s)$ that can be estimated with equation (1), then select the service with the top-$k$ highest ranking scores as the recommendation list.

**Experiment evaluation**

In this section, we report on the results of a series of experiments conducted to evaluate the performance of the proposed recommendation model. We first describe the settings of experiments including data sets, comparative methods, and evaluation metrics. Then, we report and discuss the experimental results.

**Experimental settings**

**Data sets.** We set up a testbed includes one workspace and two smart houses, where 196 objects are tagged with RFID and various state-change sensors for collecting users’ object usage event. This task greatly benefits from our extensive experience in IoT things categorization service, and we have implemented a prototype that users can check and control things real time via an Android application. For obtaining completed event usage information, two types of information need to be recorded: (1) service information. To obtain the service information, each participant utilized a context-aware service application to record the services they are using, including the start time and end time; (2) temporal information. To map the object’s usage time to the corresponding timestamps, we split one day into 24 timestamps with 1 h as an interval. Thirteen volunteers participated in the data collection phase during 1 year, and more than 50,000 object usage events are collected in the experiment. However, the data set collected from our testbed is still small. We augmented our data set with two data sets that published by the AI group in MIT to make the data set big enough for experimental studies (Table 1).

For our collected data set, we extract objects involved in a service according to record information of context-aware service application by time window overlapping (as shown in Figure 4). Note that some objects may involve in different services, for example, oven participates in roast steak and pizza service. Table 2 is a summary of the statistics of the three data sets for SIoT service recommendation. Unfortunately, the 15 users of the augmented data set are still too few for the experiment study. As a consequence, we write a computer program to generate random data set that contains information about users and objects they own and objects usage event they participated. The data set contains 500 users, each user is assigned a different number of object usage events with a minimum of 100 and a maximum of 1000.

**Comparative methods.** We compare the proposed SIoT service recommendation model with the following four state-of-art baselines:

- Probabilistic Matrix Factorization–based Service Recommendation Model (PMFR): PMFR was originally proposed to perform smart object recommendation in IoT. This model performs recommendation by fusing various relations across heterogeneous entities of IoT, such as user–object relations, user–user relations, and object–object relations. More exactly, PMFR learns shared latent factors stemming from the probabilistic matrix factorization for fusing heterogeneous entities of IoT. We follow this work to predict the recommendation score of a SIoT service by predicting service usage value $y_{ij}$, which is

| Table 1. An example of daily activity in MIT data set. |
|-----------------|-----------------|
| **Daily activity** | **Going out to work, 4/1/2003, 12:11:26, 12:15:12** |
| **Things** | Closet, jewelry box, door |
| **Starting time** | 12:12:29, 12:13:27, 12:13:45 |
| **Ending time** | 12:13:00, 12:13:35, 12:13:48 |

Note: This table shows an example of daily activity in the MIT data set. The data includes going out to work on 4/1/2003, with starting and ending times at 12:11:26 and 12:15:12 respectively. The activities are accompanied by timestamps of 12:12:29, 12:13:27, and 12:13:45.

Chen et al.
Table 2. Detailed information of the three data sets for SIoT service recommendation.

|                | MIT S1 | MIT S2 | Our data set |
|----------------|--------|--------|--------------|
| # of objects   | 76     | 70     | 196          |
| # of daily services | 21     | 23     | 93           |
| # of object usage events | 295   | 208    | 50,000       |
| # of participants | 1      | 1      | 13           |
| # of collecting period (days) | 14     | 14     | 365          |

reflected by the usage frequency of service. The service usage value depends on both the inner product of user and object latent factors and the bilinear product of user observable features $f_u$ and service observable features $f_s$ as follows

$$r(u, s) = u^T s + \sum_{mn} c d f_u^m f_s^n w_{mn}$$  \hspace{1cm} (3)$$

where $c$ and $d$ are the dimensionalities of users’ explicit features and objects explicit features, $m$ and $n$ are the number of users and services.

- Tripartite Graph-based Service Recommendation Model (GraphR): GraphR$^{26}$ performs SIoT service recommendation based on the mass diffusion dynamic tag tripartite graph, where the tripartite graph is built by extracting the users’ habit features of using the IoT device service as the dynamic tag. For generating recommendation list, GraphR performs the positive and backward mass diffusion on the tripartite graph, then calculates the energy state, the user similarity, the bearing capacity, and the service resource value of the mass diffusion results. Let $f_u^a$ denote the resource value, $R_{us}(t)$ and $T(t)$ denote the user similarity and energy left at current time, respectively, GraphR estimates the ranking score of service as follows

$$r(u, s) = k(t)(w_1 R_{us}(t) + w_2 T(t) + w_3 f_u^a)$$  \hspace{1cm} (4)$$

where $k(t) = 0$ if service reaches the upper limit of the load at the time $t$, otherwise $k(t) = 1$.

- Trust-based Service Recommendation Model (TrustR): TrustR$^{36}$ generates service recommendation list based on the integrated trust values. User selects one or multiple objects with high trustworthiness according to service requirements and interacts with them. In addition, TrustR estimated the trustworthiness value between user and service by considering high timeliness and corresponds to a precise time point as follows

$$r(u, s, t) = w_1 R_{us}(t) + w_2 S_{us}(t) + w_3 E_{us}(t)$$  \hspace{1cm} (5)$$

where $R_{us}(t)$, $S_{us}(t)$, and $E_{us}(t)$ denote the reputation, social relationship, and energy status of service $s$ from user $u$ perspective, respectively. $w_1$, $w_2$, and $w_3$ denote the normalized weight parameters for reputation evaluation, social relationship, and current energy status.

- Tensor-based Service Recommendation Model (TensorAR). Wang et al.$^{37}$ evaluated the recommendation score of IoT service with user attribute similarity and the attribute correlation of user and IoT service. TensorAR constructs a tensor-based linear regression model to capture the relation among the attributes of IoT services, the attributes of objects and the attributes of users. Let $f_u$ and $f_s$ represent user’s attribute vector and service’s attribute vector, $f_{ub}^a$ and $f_{sb}^b$ denote the $a_{th}$ attribute value of the user and the $b_{th}$ attribute value of the service, the recommendation score $r(u, s)$ is calculated as follows

$$r(u, s) = \sum_{a} C \sum_{b} D f_{ub}^a f_{sb}^b w_{ab}$$  \hspace{1cm} (6)$$

where $C$ and $D$, respectively, represent the number of the user attributes and the number of the service attributes, $w_{ab}$ is the correlation between the user attribute $f_{ub}^a$ and service attribute $f_{sb}^b$.

**Evaluation metric.** Following the work,$^{14,15}$ we adopt two widely used metrics: mean absolute error (MAE) and Recall @ $k$, as the measurement metrics to evaluate recommendation effectiveness, which are defined as:

- MAE, which is utilized to measure the accuracy of predicted recommendation score, and the lower the MAE, the better the performance. Specifically, MAE calculates the average of absolute difference between predicted recommendation score and actual recommendation score as the following

$$MAE = \frac{\sum_{ij} |\hat{r}(u_i, s_j) - r(u_i, s_j)|}{|TE|}$$  \hspace{1cm} (7)$$

where $r(u_i, s_j)$ represents the predicted recommendation score calculated by different recommendation models, $\hat{r}(u_i, s_j)$ is the actual recommendation score, $|TE|$ is the number of the predicted service usage values.

- Recall@$k$: The calculation process of Recall @ $k$ is as follows: (1) randomly select $N$ services that unused by user $u$, and compute the
recommendation score for service $s_i$ and the additional selected $N$ services; (2) form a ranked list by ordering all the $N + 1$ services according to their recommendation scores. Let $ind$ denote the rank of the test service $s_i$ within this list; (3) generate a top-$k$ recommendation list by picking the $k$-top ranked services from the list. Let $\#hit@k$ denote a single test case as either the value is 1 if the test item $s_i$ appears in the top-$k$ results, or else the value is 0. The overall $Recall@k$ are defined by averaging all test cases

$$Recall@k = \frac{\#hit@k}{|TE|} \tag{8}$$

where $k$ is the number of the recommendation results, $\#hit@k$ denotes the number of hits in the test set, and $|TE|$ is the number of all test cases.

**Experimental results**

In this sub-section, we first study the overall performance of different SIoT service recommendation models. Then we conduct two groups of fine-grained experiments and report their performance to validate the efficiency. The first group aims to evaluate the recommendation performance for all users and cold-start users. The second group investigates the impact of different training data ratios on recommendation performance.

**Recommendation effectiveness for all users and cold-start users.** We randomly select 70% of users’ object usage events as training set and use the rest of data as the test set. Then we test the performance of different recommendation models by setting the length of recommendation list as 7, and report the MAE in Table 3. From the table, we observe our method and PMFR show better performance consistently than GraphR, TrustR, and TensorAR as they learn user’s preference from object usage events. For instance, the MAE of PMFR is 1.74 while the MAE of TensorAR is 1.79. This clearly demonstrates the effectiveness of mining user’s service preference from interaction behavior between users and objects.

We further evaluate $Recall@k$ by varying the number of recommendation list from 1 to 12 in Figure 5, since a greater value of $k$ is usually ignored for a top-$k$ recommendation task. In this experiment, we randomly select 70% data as training data set and using the rest 30% as test data set. Figure 5(a) reports the performance of these service recommendation methods for all users. It is apparent that these methods have significant performance disparity in terms of top-$k$ recall. From this figure, we observe: (1) the performance of all recommendation models increases as the number of recommendation list increases. The reason for better performance is, increasing the number of recommendation results makes the data denser, leading to better recommendation; (2) our method and PMFR outperforms other competitor recommendation models (GraphR, TrustR and TensorAR) significantly, showing the advantages of learning users’ preference from object usage events. For example, the $Recall@k$ of our method is about 32.7% when $k = 8$, and the performance is improved by 8.7% and 16.43% compare with GraphR and TrustR, respectively. Similar results are also observed in top-$k$ recommendation of PMFR, taking $Recall@5$ as an example, the performance is improved: 3.75% (GraphR), 5.41% (TensorAR), and 8.15% (TrustR); (3) for recommendation algorithms that learn user’s preference from their objects usage events, our method achieves the best performance, showing the advantage of generating service recommendation by considering complex social relationships of heterogeneous SIoT objects and user preference with graph embedding. For instance, the recall of our method is about 36.72% for top-10 service recommendation, (i.e. the proposed recommendation model has a probability of 36.72% of selecting a service within target user’s service list in the top-10), while 32.75% for PMFR; (4) TrustR performs worst among all SIoT service recommendation algorithms, which suggests that only considering the trust value of a service is insufficient to reflect the level of user’s preference. Similarly, the results of GraphR and TensorAR suggest that only utilizing the similarity of user’s profiles and service’s

| Method         | Metric       | PMFR | GraphR | TrustR | TensorAR | Our method |
|----------------|--------------|------|--------|--------|----------|------------|
|                | MAE          |      |        |        |          |            |
|                | Recall@k     | 0.273| 0.225  | 0.154  | 0.192    | 0.30       |

PMFR: probabilistic matrix factorization–based service recommendation model; MAE: mean absolute error.
feature is insufficient to reflect the level of user’s preference.

To investigate the advantage of the proposed method for SIoT service recommendation, we further compare the recommendation performance of different algorithms for “cold-start” users in Figure 5(b). In this experiment, we regard users whose object usage events are less than 500 records as “cold-start” users. From the figure, we observe: (1) the recommendation performance of different service recommendation methods for cold-start users degrades significantly compared to all users, showing data sparsity caused by few objects usage events bring serious challenge for learning user’s preference. For instance, the Recall@10 of GraphR for “cold-start” users drops 11.85% compared with all users; (2) the proposed method performs much better than baseline algorithms (i.e. GraphR, TrustR, TensorAR, and PMFR), showing the advantage of SIoT service recommendation by capturing various social relationships among users, objects, and services with graph embedding. For instance, the performance of our method improved 5.37% compared with TrustR when \( k = 10 \). This is no surprise since our method can present new services and users by the learned vector representations of used records, thus the potential user-service links between cold-start services and users will be captured correspondingly. In other words, the proposed method can leverage not only the user-service links in SIoT Knowledge graph but also various social relationships when recommending cold-start users.

Recommendation effectiveness by using object’s social relationship. To illustrate one can generate better recommendation results by jointly considering both user’s preference and smart object’s social similarity, we compare our method with ItemCF, which generates the recommendation results by only considering the “like” property between user and service by modeling user’s preference toward an object with a latent variable model. Specifically, ItemCF predicts the recommendation score between users and objects without considering both the temporal influence and the object’s social similarity. The recommendation score \( r(u_i, s_j) \) between user \( u_i \) and smart object \( s_j \) is estimated as follows

\[
r(u_i, s_j) = \frac{\sum_{s_p \in u_i} \text{sim}(s_p, s_j) I^{(p)}}{\sum_{s_p \in u_i} \text{sim}(s_p, s_j)}
\]

where \( s_p \) is the object set that user \( u \) has used, \( \text{sim}(s_p, s_j) \) is the correlation-based similarity between object \( s_j \) and \( s_p \) that are estimated by the set of users who both rated \( s_j \) and \( s_p \), \( I^{(p)} \) is the preference of user \( u_i \) to object \( s_p \), the graphical model representation and inference of modeling \( I^{(p)} \) can be found in Chen et al.’s study.\(^3\)\(^1\)\(^4\)

We investigate the recommendation performance by considering object’s social relationship, the results of Recall@\( k \) are reported in Figure 6 by increasing the length of the recommendation list from 1 to 5. From this figure, we observe (1) our method performs better than ItemCF for different lengths of recommendation list. For all users, the Recall@\( k \) of our method is 0.261 when the length of recommendation list is 5, which improves 11.3% than ItemCF (0.235). The results show again the advantage of SIoT object recommendation by jointly considering user’s preference and object’s social similarity; (2) the performance of both our method and ItemCF increase as the length of recommendation list increases. Taking ItemCF of cold-start users as an example, the Recall@\( k \) is 0.132 and 0.151 when the length of recommendation list is 3 and 5, respectively. The reason is increasing the length of time slot makes the data denser, which leads more ground truth objects in each time slot thus leading to better recommendation.
Recommendation effectiveness with different training data ratios. We investigate the impact of different training data ratios to the performance of GraphR, TrustR, TensorAR, PMFR, and our method. As shown in Figure 7, the performance of all service recommendation methods degrade to some extent as the training data ratios decreases. Nevertheless, our method and PMFR show better performance consistently than the other three methods over all training data ratios as they learn user’s preference from object usage events. For example, the Recall@8 of our method is 19.25% when the training data ratio is 40%, while 11.2% of GraphR with the same training data ratio. This clearly demonstrates the effectiveness of modeling user’s service preference from their object usage events. We also observe the proposed method can achieve considerable performance even when the training data ratio is relatively low, while GraphR, TrustR, and TensorAR perform poorly with few samples to train model. For example, the Recall@9 of our method drops 12.6% when the training data ratio decreases from 40% to 20%, while the number is 35.6% of TensorAR with the same condition. This reason is TensorAR requires enough service profiles or obvious user spatial-temporal patterns for training model, while our method can uncover user’s preference from their object usage events. However, the performance improvement by mining user’s preference from object usage events is diverse for our method and PMFR. Taking Recall@5 for example, the performance improvement of PMFR is about 5.14% compare with GraphR when the training data ratio is 40%, while the performance improvement of our method is very obvious (about 9.84%). This is no surprise since PMFR generates service recommendation results by merely considering the user’s preference, leading the advantage can be ignored when users have few object usage events.
On the contrary, our method generates service recommendation results by jointly considering user’s preference and object’s various social relationships, and further present new services and users by the learned vector representations of used records, thus the potential user-service relationship will be modeled correspondingly even the user with few object usage events.

Conclusion

In this article, we propose a novel method to recommend SIoT service for users, especially for the users with few training data. The proposed method first learns user’s preference from objects usage events with a latent variable model. Then, we construct a high-quality knowledge graph by bridging diverse social relationships among SIoT objects and user’s preference. Specifically, the proposed method jointly embeds users, objects, services, and their corresponding relations into a shared lower-dimensional space. Finally, we model SIoT service recommendation as a knowledge graph completion problem, and use the graph embeddings to decompose the SIoT service recommendation into a linked prediction process while considering the user’s objects usage event and object’s social relationships. Extensive experiments on real-world data sets are conducted and the results demonstrate the effectiveness of the proposed method.

In future work, we will improve the SIoT service recommendation accuracy by considering more context information of users, such as the spatio-temporal behavior pattern and the quality of services.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by the Young Scientists Fund of the National Natural Science Foundation of China (grant no. 61802343), in part by the Zhejiang Provincial Natural Science Foundation of China (grant nos. LGF19F020019 and LGN20F020003), and in part by the Intelligent Plant Factory of Zhejiang Province Engineering Lab.

ORCID iDs

Yuanyi Chen https://orcid.org/0000-0002-8592-2033
Yanyun Tao https://orcid.org/0000-0002-5342-1699

References

1. Pattar S, Buyya R, Venugopal KR, et al. Searching for the IoT resources: fundamentals, requirements, comprehensive review, and future directions. IEEE Commun Surv Tut 2018; 20(3): 2101–2132.
2. Chen Y, Zhou J and Guo M. A context-aware search system for Internet of Things based on hierarchical context model. Telecommun Syst 2016; 62(1): 77–91.
3. Chen Y, Zhang J, Xu L, et al. Modeling latent relation to boost things categorization service. IEEE T Serv Comput 2020; 13: 915–929.
4. Chen Y, Zhou M, Zheng Z, et al. Toward practical crowdsourcing-based road anomaly detection with scale-invariant feature. IEEE Access 2019; 7: 67666–67678.
5. Li W, Chen Z, Gao X, et al. Multimodal framework for indoor localization under mobile edge computing environment. IEEE Internet Things 2019; 6(3): 4844–4853.
6. Li W, Xu H, Li H, et al. Complexity and algorithms for superposed data uploading problem in networks with smart devices. IEEE Internet Things 2020; 7(7): 5882–5891.
7. Zhang J, Zhong S, Wang T, et al. Blockchain-based systems and applications: a survey. J Internet Technol 2020; 21(1): 1–14.
8. Zhang Z, Wu C, Yang D, et al. HSCS: a hybrid shared cache scheduling scheme for multiprogrammed workloads. Front Comput Sci 2018; 12(6): 1090–1104.
9. Atzori L, Iera A, Morabito G, et al. The Social Internet of Things (SIoT)—when social networks meet the Internet of Things: concept, architecture and network characterization. Comput Netw 2012; 56(16): 3594–3608.
10. Chen Y, Zhou M, Zheng Z, et al. Time-aware smart object recommendation in Social Internet of Things. IEEE Internet Things 2020; 7: 2014–2027.
11. Atzori L, Iera A and Morabito G. SIoT: giving a social structure to the Internet of Things. IEEE Commun Lett 2011; 15(11): 1193–1195.
12. Chen IR, Guo J and Bao F. Trust management for SOA-based IoT and its application to service composition. IEEE T Serv Comput 2016; 9(3): 482–495.
13. Li J, Bai Y, Zaman N, et al. A decentralized trustworthy context and QoS-aware service discovery framework for the Internet of Things. IEEE Access 2017; 5: 19154–19166.
14. Chen Y, Zhang J, Guo M, et al. Learning user preference from heterogeneous information for store-type recommendation. IEEE T Serv Comput 2020; 13: 1100–1114.
15. Yao L, Sheng QZ, Ngu AHH, et al. Things of interest recommendation by leveraging heterogeneous relations in the internet of things. ACM T Internet Techn 2016; 16(2): 9.
16. Gardner M and Mitchell T. Efficient and expressive knowledge base completion using subgraph feature extraction. In: Proceedings of the 2015 conference on empirical methods in natural language processing, Lisbon, 17–21 September 2015, pp.1488–1498. Stroudsburg, PA: Association for Computational Linguistics.
17. Minervini P, d’Amato C, Fanizzi N, et al. Leveraging the schema in latent factor models for knowledge graph
completion. In: Proceedings of the 31st annual ACM symposium on applied computing, Pisa, 4–8 April 2016, pp.327–332. New York: ACM.

18. Rasch K. Smart assistants for smart homes. PhD Thesis, KTH Royal Institute of Technology, Stockholm, 2013.

19. Hussein D, Han SN, Lee GM, et al. Social cloud-based cognitive reasoning for task-oriented recommendation. IEEE Cloud Comput 2015; 2(6): 10–19.

20. Subramaniyaswamy V, Manogaran G, Logesh R, et al. An ontology-driven personalized food recommendation in IoT-based healthcare system. J Supercomput 2019; 75: 3184–3216.

21. Felfernig A, Erdeniz SP, Jeran M, et al. Recommendation technologies for IoT edge devices. Procedia Comput Sci 2017; 110: 504–509.

22. Beltran V, Ortiz AM, Hussein D, et al. A semantic service creation platform for Social IoT. In: Proceedings of the 2014 IEEE world forum on Internet of Things (WF-IoT), Seoul, South Korea, 6–8 March 2014, pp.283–286. New York: IEEE.

23. Mashal I, Chung T-Y and Alsaryrah O. Toward service recommendation in Internet of Things. In: Proceedings of the 2015 7th international conference on ubiquitous and future networks, Sapporo, Japan, 7–10 July 2015, pp.328–331. New York: IEEE.

24. Jeong H, Park B, Park M, et al. Big data and rule-based recommendation system in Internet of Things. Cluster Comput 2019; 22: 1837–1846.

25. Wanigasekara N, Schmallfuss J, Carlson D, et al. A bandit approach for intelligent IoT service composition across heterogeneous smart spaces. In: Proceedings of the 6th international conference on the Internet of Things, Stuttgart, 7–9 November 2016, pp.121–129. New York: ACM.

26. Wang P, Luo H, Obaidat MS, et al. The Internet of Things service recommendation based on tripartite graph with mass diffusion. In: Proceedings of the 2018 IEEE international conference on communications workshops (ICC Workshops), Kansas City, MO, 20–24 May 2018, pp.1–6. New York: IEEE.

27. Saleem Y, Crespi N, Rehmani MH, et al. Exploitation of social IoT for recommendation services. In: Proceedings of the 2016 IEEE 3rd world forum on Internet of Things (WF-IoT). Reston, VA, 12–14 December 2016, pp.359–364. New York: IEEE.

28. Cao B, Liu J, Wen Y, et al. QoS-aware service recommendation based on relational topic model and factorization machines for IoT Mashup applications. J Parallel Distr Comput 2019; 132: 177–189.

29. Lee J-S and Ko I-Y. Service recommendation for user groups in Internet of Things environments using member organization-based group similarity measures. In: Proceedings of the 2016 IEEE international conference on web services (ICWS), San Francisco, CA, 27 June–2 July 2016, pp.276–283. New York: IEEE.

30. Khelloufi A, Ning H, Dhelim S, et al. A social-relationships-based service recommendation system for SIoT devices. IEEE Internet Things 2021; 8: 1859–1870.

31. Wang H, Wang Z, Hu S, et al. DUSKG: a fine-grained knowledge graph for effective personalized service recommendation. Future Gener Comp Sys 2019; 100: 600–617.

32. Ferrag MA, Maglaras L and Ahmim A. Privacy-preserving schemes for ad hoc social networks: a survey. IEEE Commun Surv Tut 2017; 19(4): 3015–3045.

33. Yao L, Sheng QZ, Ngu AHH, et al. Exploring recommendations in internet of things. In: Proceedings of the 37th international ACM SIGIR conference on research & development in information retrieval, Gold Coast, QLD, Australia, 6–11 July 2014, pp.855–858. New York: ACM.

34. Wang Z, Zhang J, Feng J, et al. Knowledge graph embedding by translating on hyperplanes. In: Proceedings of the 28th AAAI conference on artificial intelligence, Québec City, QC, Canada, 27–31 July 2014.

35. Tapia EM, Intille SS and Larson K. Activity recognition in the home using simple and ubiquitous sensors. In: Proceedings of the international conference on pervasive computing, Linz and Vienna, 21–23 April 2004, pp.158–175. Berlin; Heidelberg: Springer.

36. Chen Z, Ling R, Huang C-M, et al. A scheme of access service recommendation for the Social Internet of Things. Int J Commun Syst 2016; 29(4): 694–706.

37. Wang P, Luo H and Sun Y. IoT service recommendation strategy based on attribute relevance. In: Proceedings of the international conference on ubiquitous computing and ambient intelligence, Philadelphia, PA, 7–10 November 2017, pp.34–43. Cham: Springer.