IoT Service Recommendation Scheme Based on Matter Diffusion

PINGQUAN WANG [ID]

School of Computer and Information Engineering, Inner Mongolia University of Finance and Economics, Hohhot 010070, China
Hohhot Minzu College, Hohhot 010051, China

e-mail: wangpq@imufe.edu.cn

ABSTRACT With the development of Internet of Things(IoT), more and more smart devices and services appeared in people's life. These services make life more convenient, while it is difficult for people, especially the elderly, to find them in time, because they are widely distributed in great numbers with small sizes. In order to solve this problem, the service recommendation scheme, which is becoming increasingly important, needs to be used in IoT. However, the traditional web service recommendation schemes are not suitable for the IoT, because they rely more on the historical information rather than the energy state or the user's habit attributes, which are important for IoT. We research the service recommendation scheme, finding that the service recommendation process in IoT is a process of matter flows, which follows the conservation of matter. Therefore, we propose the service recommendation scheme based on tripartite graph with matter diffusion and use the habit feature as a dynamic tag. Based on the balance of matter, we use the positive and negative matter diffusion results on the tripartite as the recommendation results. The results of our evaluation show that the performance of the service recommendation scheme is improved in the precision and recall.

INDEX TERMS Service recommendation, Internet of things, matter diffusion, smart home, tripartite graph.

I. INTRODUCTION

Nowadays, a large number of intelligent devices are integrated with sensors, thus enabling them to detect real-time user behavior features and environment information. In our previous research, we used the smart switch, environment sensor and smart bracelet to record user's daily activities and extract the habits of user' using electrical equipment [1]. We discovered that the habits of users' using electrical appliance could be classified into the fixed-length habit and the timing habit for the first time. In addition, there were different relationships between habits, such as *selection*, *sequence*, *cross*, *parallel*, *inclusion*, *beginning*, and *finish*. In addition, we translated the habits into the service rules by SWRL, and realized the automatic control of the smart home equipment based on the service rules of habits [2]. The good habits could be spread as a culture in the smart home, so the services based on the habits could be recommended to others as well. For example, in the smart ward the patient with good habits

likes going to bed early and getting up early would achieve better rehabilitation results. This habit is associated with turning light on and off, therefore, the services of turning light on and off on time could be recommended to others. All in all, we researched the service recommendation scheme for such services to help people find the services in time.

The service in IoT is different from the traditional web service in user characteristics, constraint of energy, trust correlation, social relationship, etc. The IoT services come from the habits of users [2], and they will have user characteristics such as department, age, name, etc. The services in IoT usually rely on smart devices with limited power, but they may stop providing service content when the energy is exhausted, therefore they are called energy-constrained. In the IoT, the user prefers to use the trusted services. The services trusted by most users would have high trust value, therefore they are called trust correlation. The services would have social relations such as leader-member relationship, ownership relationship, etc. These relationships come from their owners, thus they have social relationship.

The associate editor coordinating the review of this manuscript and approving it for publication was Miguel Jesus Torres Ruiz.

Therefore, the service recommendation scheme should take account of these new features of service in IoT. However, the traditional web service recommendation schemes [3]–[6] like content-based filtering approach, collaborative filtering method and hybrid system approach are not suitable for the IoT. The traditional service recommendation schemes are mainly aimed at the Internet environment, and recommend the corresponding items to the users based on their previous evaluations, access records or preferences. However, in the IoT environment, the users rarely evaluate the service after using the corresponding IoT device services, therefore, the services cannot be recommended through the evaluation records of the previous users.

A. RESEARCH CONTRIBUTIONS

In this paper, we consider the service recommendation process in IoT as a process of matter diffusion. The energy state is regarded as the matter. The services with more matter would be easier to be recommended to users. The main contributions to the research for this work are summarized as follows.

- This paper proposes a IoT service recommendation scheme based on tripartite graph with matter diffusion. We treat the energy as the matter and believe that the IoT service recommendation process follows the conservation of matter for the first time.
- We chose the users' habit behavioral features about how they used the IoT device service as the dynamic tag, and we further built the complete tripartite graph network with matter diffusion based on the dynamic tags, the users and the IoT services. In the tripartite graph network, we could get the service's matter value more efficiently by the calculation of positive and backward matter diffusion value.
- The performance of the proposed technique was evaluated and compared with the existing model with respect to various performance metrics.

B. ORGANIZATION

The rest of the paper is structured as follows: Section II discusses the related work; Section III presents the service recommendation framework; Section IV presents the evaluation of the proposed model based on different performance metrics; Section V presents the conclusions of this research.

II. RELATED WORK

A. SERVICE RECOMMENDATION SCHEME IN WEB SERVICE

The service recommendation system, which is a tool automatically connecting users and items, recommends the service to the interested user, and is widely used in e-commerce, video websites, social networking, etc. There are three main types of recommendation algorithms in web service, namely content-based recommendation algorithm [7]–[9], collaborative filtering algorithm(CF) [10]–[14], and graph-based recommendation algorithm [15]–[17]. CF techniques use a database of user preferences for items to predict additional

topics or products a new user might like [18]. [19] used the user rating data to calculate the similarity of users or items and made the predictions or recommendations according to the similarity values. The memory-based CF methods are notably deployed into commercial systems such as http://www.amazon.com/, because they are easy to implement effectively [20], [21]. Customization of CF systems for each user decreases the search effort for users. It also promises a greater customer loyalty, higher sales, more advertising revenues, and the benefit of targeted promotions [21]. However, the CF needs to update the dataset of user's hobbies in real time, which is difficult to be achieved in the IoT, because it is difficult to obtain the user's interest in time.

The content-based recommendation is based on the text of the item and the historical access record [9], and it recommends the items that are similar to the items previously used by the user. The major difference between CF and content-based recommendation systems is that CF only uses the user-item ratings data to make predictions and recommendations, while content-based recommendation systems rely on the features of users and items for predictions [22]. The content-based recommendation algorithm needs to calculate the item similarity, so it is not suitable for IoT because the time complexity is too high.

The graph based on recommendation algorithm has been widely used in recommendation system. The PageRank algorithm of Google, which is based on graph [23], achieves a better effect on sorting web pages. The tripartite graph is built in [24] which is based on user, resource and tag. It uses the unified tagging information to reveal user's behaviors and preferences, extracts the latent semantic relations among items to make the recommendations. The tag could include different types of message such as position, environment, hobby, etc. These information could comprehensively describe the context information which is important for the recommendation results. The graph based on recommendation is also used in the [25]–[27], and gets better recommendation effects. The IoT service recommendation needs more attributes than traditional web service recommendation, such as trust, social relationship, energy, geography position, etc, which are seldom used in web service recommendation, however, they are important in IoT service recommendation.

B. SERVICE RECOMMENDATION SCHEME IN IoT

At present, the researches on IoT service recommendation are still in its early stage. The IoT service has some new characteristics such as user characteristics, constraint of energy, and trust correlation. The IoT service recommendation scheme should take these characteristics into consideration. The IoT service recommendation framework was proposed in [28], [29], using the similarity calculation results of users and smart devices. The service recommendation system based on the formal model was proposed in [30]. As the first attempt, this paper puts forward a kind of hyper graph model, which is used to model the time and space correlation of the IoT, in which each edge connects the user,
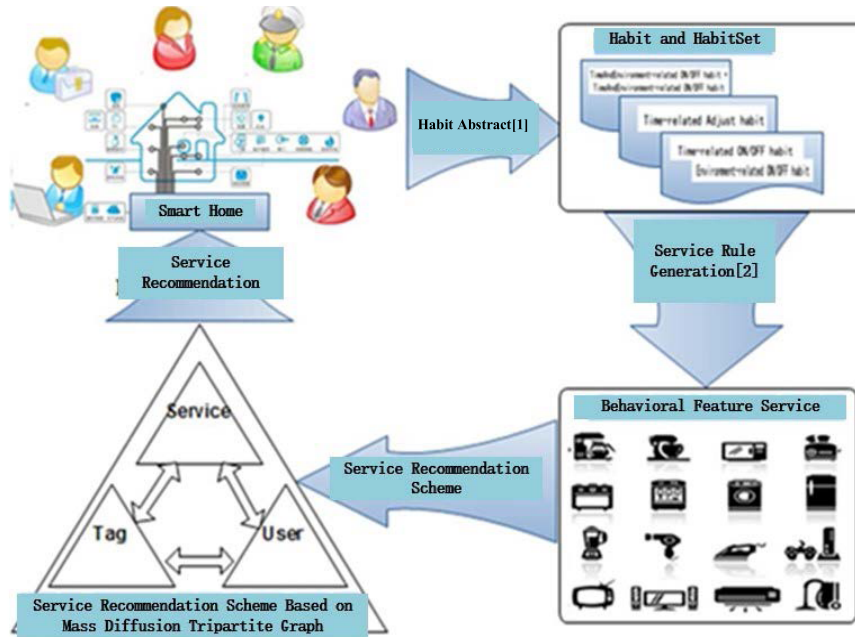
FIGURE 1. Service recommendation framework.

the objects and the service. However, the paper does not consider things' own properties, such as the energy state, the time and spatial relevance of things. The attribute relationship and energy-aware mechanism were first used in [31] for the service recommendation, in which each node is a provider of information/services as well as a requester or presenter of the information service. Requesting a particular service, the node should broadcast the request information on the network and then select the appropriate node. Each node maintains a transactional history and brief profiles. The node with high trust relationship and energy state will be recommended to the user. In [32], the authors use cognitive reasoning into the social IoT for the recommendation of everyday tasks. Tag is used to describe the different kinds of information in the Internet service recommendation system, such as user's topic preferences as well as the content of items [33]. In [37]–[39], the authors use the lightweight situation aware for the service composition. In the IoT service recommendation, tag could have many definitions such as human feature, environment feature and situation, but it has seldom been applied in IoT service recommendation.

III. SERVICE RECOMMENDATION FRAMEWORK

In our previous researches, we extracted the user's habits of using electrical equipment [1] in smart home, and translated them into the IoT services[2] which realized the automatic control of the smart home. For example, the TV set service could turn on TV according to the user's habits. In order to recommend these services to other persons in the smart home, we propose the service recommendation framework as shown in Fig.1. The framework mainly includes three parts, namely habit abstract, service rule generation, and service recommendation scheme. The user's daily behaviors

are recorded by different kinds of sensors. The habit abstract module extracts the user's habits and habit set by the scheme in [1], and translates them into the service rules in service rule generation module by the scheme in [2]. The behavioral feature services which are based on the habit and habit set would be recommended to users in service recommendation scheme module by the service recommendation scheme based on matter diffusion on tripartite graph.

A. BEHAVIORAL FEATURE SERVICE

The service which is consistent with user's habit is called behavioral feature service, which is very common in the smart home. For example, the user is used to turning the light on at eight o' clock every day, and we could collect this habit and translate it into a kind of behavioral feature service [1], [2]. The formal definition of a behavioral feature service status at time t can be represented as follows.

Definition 1 (Behavioral Feature Service): The behavioral feature service is based on the user's habit. Given a behavioral feature service S_k, its status at time t is $S_k(t) \equiv < D_k, H_k, T_k(t) >$, in which $D_k \subset DevSet$, $DevSet$ is the device set in the smart space, $H_k \subset HabitSet$, and HabitSet is the habits which make up the services. $T_k(t)$ is the trust status of the service at time t.

The habit feature is obtained from the *on/off* or time series of the activity adjustment, which is shown in Fig.2. An activity is a pair of on/off operation or an operation regulation. In Fig.2, the capital letters without being underlined stand for the *on* operation, the lowercase letter stands for the *off* operation, and the underlined capital letters stand for the

$$A_{1\text{-}1} \ \underline{B}_{1\text{-}2} \ C_{1\text{-}3} \ D_{1\text{-}4} \ c_{1\text{-}5} \ a_{1\text{-}6} \ C_{1\text{-}7} \ A_{1\text{-}8} \ c_{1\text{-}9} \ d_{1\text{-}10} \ C_{1\text{-}11}$$

FIGURE 2. On/Off/ activity adjustment.

S1: behavioral feature service
T1: dynamic tag
U1: user

(a) An example of tripartite graph with dynamic tag

(b). correlated subgraph 1

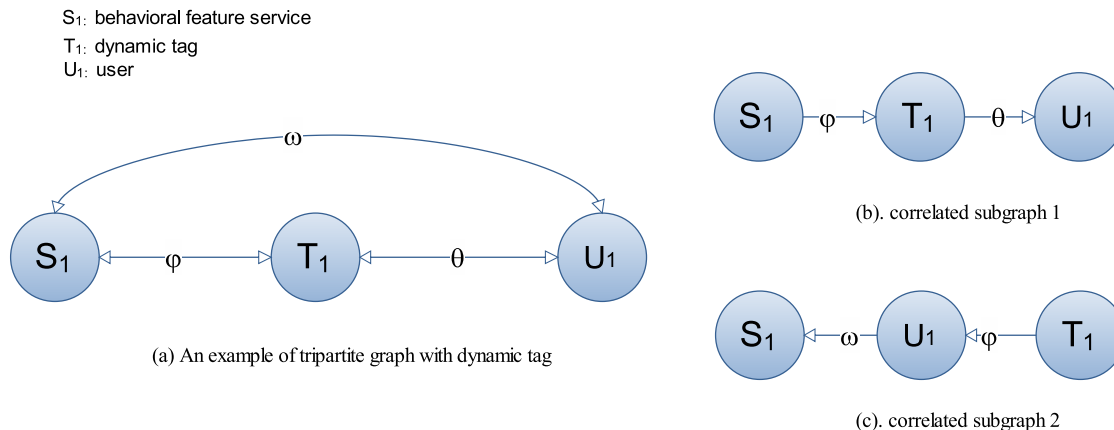(c). correlated subgraph 2

FIGURE 3. Tripartite graph with dynamic tag.

operation regulation. The subscript represents the monitoring period and time.

B. TRIPARTITE GRAPH WITH DYNAMIC TAG

The behavioral feature services use different kinds of habits as the service rules. The habits contain the information about the activities such as type of people, location, time, operation, etc. These information is useful to IoT service recommendation. For example, the person's early rising habit usually happens in the morning in the bedroom with the light on. When a new person come into the smart home, we could recommend the behavioral feature service to him based on the location, age, time, etc. These information is seldom used in the tradition web service recommendation, but is important for IoT service recommendation. The tag could carry more information, and it could be used to describe different kinds of information in the Internet service recommendation system, such as user's topic preferences as well as the content of items [33]. In order to recommend the popular behavioral feature service to the user, we could label the service with the habit and recommend the service with tag based on service recommendation scheme.

We use the habit as a kind of dynamic tag, and construct a tripartite graph of *user-dynamic tag-service*. The dynamic tag of habit could well describe the time and environment characteristics which are important for Internet of things. Furthermore, the user's habit usually changes with time and environment, and the dynamic tag could describe it well. The formal definition of dynamic tag could be represented as follows.

Definition 2 (Dynamic Tag): The dynamic tag is based on the habit and it could be expressed as a triple which is $< L_s >:= (H_s, O_s, D_s)|t_s, e_s$, in which $H_k \subset HabitSet$, which is the habits which make up the tag L_s, O_s is the operation sequence of the habit H_s, $D_s \subset DevSet$, which is the device set in the smart space, t_s is the time characteristic of tag L_s, and e_s is the environment characteristic of tag L_s.

We could see from the definition 2 that when the time and environment meet the condition of t_s and e_s, the recommendation system will show the dynamic tag L_s. For example, when the humidity is higher than 60% between 7 o'clock and 8 o'clock in the morning, the recommendation system will show the dynamic tag of querying the humidity.

Based on the definition of dynamic tag, we further define the tripartite graph with dynamic tag as shown in Fig.3(a).

The tripartite consists of three bipartite graphs, including user-service, service-dynamic tag and dynamic tag-user. The user-service bipartite stands for the user' using the service directly, the service-dynamic tag bipartite stands for the service labeled by the dynamic tag, and the dynamic tag-user stands for the dynamic tag used by the user. The tripartite graph has two directions including service->dynamic tag->user bipartite graph as shown in Fig.3(b) and service->user->dynamic tag as shown in Fig.3(c). There are some relationships among users, services and dynamic tag, such as the usage and the annotation. To describe these relationships, we propose the edge weight ω, φ and θ shown in Fig.3.

The edge weight is related to the historical usage record. For example, ω is the edge weight between service and user, standing for the state that the service is used by the user.

$$w(M_i) \equiv < u_i, h > |M \in \{\omega, \varphi, \theta, \} \tag{1}$$

where u_i is the history records between the two objects T_i and S_i or U_i in the tripartite, h is the total records between the two objects T and S or U.

In order to evaluate the weight value, we further use the delta rule [34] which is based on gradient descent to get the edge weight. The edge weight calculation algorithm is shown in Algorithm 1, which chooses the random weight vector $\vec{w}$ as the initial value, and gets the training samples by the $o(\vec{x}) = \vec{w} \bullet \vec{x}$. We get Δw_i by Formula (2), where D is the training sample set, t_d is the target output of the training sample d, o_d is the output of the linear element to the training samples, x_{id} is the input component x_i of the linear element d, and η is the learning rate.

$$\Delta w_i = \eta \sum_{d \in D} (t_d - o_d)x_{id} \tag{2}$$

Algorithm 1 Get the Edge Weight

Input: The training sample is $< \vec{x}, t >$, where $\vec{x}$ is the input vector, t is the target output value, t_d is the target output of the training sample d, o_d is the output of the training sample d, η is the learning rate, *and z* is the global minimum error.

Output: The weight vector ω

1: initialize ω_i by the random small number
2: initialize $\Delta\omega$ by 0
3: initialize the error rate k=100
4: **while** $k > z$ **do**
5: do the following operation in the training sample $< \vec{x}, t >$:
6: $o(\vec{x}) = \vec{w} \bullet \vec{x}$
7: $\Delta\omega_i \leftarrow \Delta\omega_i + \eta(t_d - o_d)x_i$
8: $\omega_i \leftarrow \omega_i + \Delta\omega_i$
9: $k = k/e, e > 1$
10: **end while**
11: **return** ω

C. THE CACLULATION OF MATTER

The IoT devices which provided the data for the services are usually powered by the batteries. Therefore the energy is very important for the IoT service. We regard the energy as a kind of matter which is calculated by the energy surplus (*ES*) and the energy consumption (*EC*). The *ES* of the device c_i could be described as Formula (3).

$$ES(c_i) = CE(c_i) - M(c_i) \tag{3}$$

$CE(c_i)$ stands for the current energy state of c_i, and $M(c_i)$ stands for the minimum energy of c_i. The *EC* of the device c_i could be described as Formula (4).

$$EC(c_i) = SEC(c_i) \bullet (c_i) \tag{4}$$

$SEC(c_i)$ stands for the energy consumption rate of device c_i, and $T(c_i)$ stands for the execution time of device c_i.

Based on the definition of $ES(c_i)$ and $EC(c_i)$, we further define the energy assessment of c_i as Formula (5),

$$EP(c_i) = ES(c_i)/EC(c_i) \tag{5}$$

We can see from the Definition 1 of behavioral feature service that one service always consists of several devices. If one service consists of n device, the energy assessment could be expressed as Formula (6):

$$EP(s_j) = min\{EP(C_1), EP(C_2), \ldots EP(C_n)\} \tag{6}$$

We can see from the Formula (6) that the service energy state depends on the device energy state which has minimum energy assessment.

D. SERVICE RECOMMENDATION SCHEME BASED ON MATTER DIFFUSION

We can see from the Fig.3 that the tripartite graph with dynamic tag has two directions, namely *service → User →*

Dynamic Tag bipartite graph as shown in Fig.3(c) and *Service → Dynamic Tag → User* as shown in Fig.3(b). We name the direction form service to user in the bipartite as forward direction, and the direction from user to dynamic tag as backward direction. The process of service recommendation is accompanied by the flow of matter, such as the energy. Therefore, the recommendation process satisfies the law of mass conservation. Based on the edge weight Formula (1) and the tripartite in Fig.3, we could construct the matrix of user-service A_{PS}, the matrix of user-dynamic tag A_{PK}, and the matrix of dynamic-service A_{KS}. In the tripartite, if there is an association between two entities, the weight should be showed in the matrix. For example, in the matrix A_{PS} as shown in Fig.4, if the user P chooses the service S, the matrix element $a_{ps} = \omega_{p,s}$, otherwise $a_{ps} = 0$.

$$\begin{array}{c} \\ P_1 \\ P_2 \\ \ldots \\ P_n \end{array} \begin{array}{ccccc} S_1 & S_2 & S_3 & \ldots & S_m \end{array} \\ \left[\begin{array}{ccccc} \omega_{1,1} & \omega_{1,2} & 0 & \ldots & 0 \\ 0 & 0 & \omega_{2,3} & \ldots & \omega_{2,m} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \omega_{n,2} & \omega_{n,3} & \ldots & 0 \end{array} \right]$$

FIGURE 4. The matrix of A_{PS}.

The backward diffusion is the process: *Service → User → Dynamic Tag*. The backward diffusion process should first find the user who used the service, then the dynamic tags the user used, and finally the service labeled by the dynamic tag. The forward diffusion is the process: *Service → Dynamic Tag → User*. The forward diffusion process should first search for the service labeled by the dynamic tag, then the owner of the dynamic tag, and finally the service which the user used.

1) MATTER IN THE BACKWARD DIFFUSION

We assume the service set is $S = \{S_1, S_2, \ldots, S_x\}$, the user set is $U = \{U_1, U_1, \ldots, U_y\}$, and the dynamic tag set is $T = \{T_1, T_1, \ldots, T_z\}$, so the matter in the backward diffusion process could be expressed as follows:

1. The initial matter of service S_i in the backward matter diffusion is f_{S_i}; after the matter is diffused from the service to the user, the matter of the user U_k is f_{U_k}; the service set which is directly connected with the user U_k is $S' = \{s_1, s_2, \ldots, s_M\}$, and $S' \in S$; the out-degree of service S_i in the tripartite graph is N_{S_i}, of which $S_i \in S'$. The matter value of U_k is:

$$f_{U_k} = \sum_{S_i \in S'} \frac{f_{S_i}}{N_{S_i}} Q_{s_i U_k}, \tag{7}$$

where $Q_{s_i U_k}$ is the weight between service S_i and user U_k in the matrix of service-user A_{PS}. f_{S_i} is the matter value of S_i in the backward diffusion.

2. After the matter diffusion of user-dynamic tag, the matter value of dynamic tag T_L is f_{T_L}, the user set

directly connected with dynamic tag T_L is $U' = \{U_1, U_2, \ldots, U_L\}$, of which $U' \in U$, and the out-degree of user U_j is N_{U_j}, of which $U_j \in U'$. The matter value of dynamic tag T_L is:

$$f_{T_L} = \sum_{U_j \in U'} \frac{f_{U_j}}{N_{U_j}} Q_{T_L U_j}, \qquad (8)$$

where $Q_{T_L U_j}$ is the weight value between user U_j and dynamic tag T_L in the matrix of A_{PK}. f_{U_j} is the matter value of user U_j.

3. After the matter diffusion of dynamic tag-service, the matter value of the service S_i is f_{S_i}, and the dynamic tag set directly connected with service S_i is $T' = \{T_1, T_2, \ldots, T_P\}$, of which $T' \in T$, and the out-degree of dynamic is N_{T_r}, of which $T_r \in T'$. The matter value of service S_i is:

$$F_{S_i} = \sum_{T_r \in T'} \frac{f_{T_r}}{N_{T_r}} Q_{T_r S_i}, \qquad (9)$$

where $Q_{T_r S_i}$ is the weight value between dynamic tag T_r and service S_i in the matrix of A_{KS}. f_{T_r} is the matter value of the dynamic tag T_r.

2) MATTER IN THE FORWARD DIFFUSION

We assume the service set is $S = \{S_1, S_2, \ldots, S_x\}$, the user set is $U = \{U_1, U_1, \ldots, U_y\}$, and the dynamic tag set is $T = \{T_1, T_1, \ldots, T_z\}$, so the matter in the forward diffusion process could be expressed as follows:

1. The initial matter of service S_i in the forward matter diffusion is z_{S_i}, but after the matter diffusion of service-dynamic tag, the matter of dynamic tag T_k is Z_{T_k}; the service set which is directly connected with the dynamic tag T_k is $S'' = \{S_1, S_2, \ldots, S_W\}$, of which $S'' \in S$; the out-degree of service S_i in the tripartite graph is N_{S_i}, of which $S_i \in S''$.
The matter value of T_k is:

$$Z_{T_k} = \sum_{S_i \in S''} \frac{z_{S_i}}{N_{S_i}} Q_{s_i T_k}, \qquad (10)$$

where $Q_{s_i T_k}$ is the weight between dynamic tag T_k and service S_i in the matrix of dynamic tag-service A_{KS}.

2. After the matter diffusion of dynamic tag-user, the matter value of user U_L is Z_{U_L}, the dynamic set directly connected with user U_L is $T'' = \{T_1, T_2, \ldots, T_O\}$, of which $T'' \in T$; the out-degree of dynamic tag T_y is N_{T_y}, of which $T_y \in T''$.
The matter value of user U_L is:

$$Z_{U_L} = \sum_{T_y \in T''} \frac{Z_{T_y}}{N_{T_y}} Q_{T_y U_L}, \qquad (11)$$

where $Q_{T_y U_L}$ is the weight between user U_L and dynamic tag T_y in the matrix of dynamic tag-user A_{PK}.

3. After the matter diffusion of user-service, the matter value of service S_i is Z_{S_i}; the user set directly connected with service S_i is $U'' = \{U_1, U_2, \ldots, U_V\}$, of which $U'' \in U$; the out-degree of user U_r is N_{U_r}, of which $U_r \in U''$.

The matter value of service S_i is:

$$Z_{S_i} = \sum_{U_r \in U''} \frac{f_{U_r}}{N_{U_r}} Q_{U_r S_i}, \qquad (12)$$

where $Q_{U_r S_i}$ is the weight between user U_r and service S_i in the matrix of service-user A_{PS}.

After the matter diffusion, the service S_i's matter value W_{s_i} could be obtained by the forward/backward diffusion parameter λ as shown in Formula (13),

$$W_{s_i} = \lambda F_{s_i} + (1 - \lambda) Z_{s_i}, \qquad (13)$$

Algorithm 2 Service Recommendation Based on Matter Diffusion Tripartite Graph

---

Input: The service set is $S = \{S1, S2, \ldots, Sx\}$; user set is $U = \{U1, U2, \ldots, UY\}$; dynamic tag set is $T = \{T1, T2, \ldots, Tz\}$; forward/backward diffusion parameter is λ; recommended list length is C; service-user matrix is A_{PS}; user-dynamic tag matrix is A_{PK}; dynamic tag-service matrix is A_{KS}.

Output: recommendation list reclist[C].

1: /*The matter value in backward diffusion */
2: **for** $U_k \in U$ **do**
3: Based on the Formula (7), get the U_k's matter value f_{U_k} in the backward diffusion.
4: **end for**
5: **for** $T_L \in T$ **do**
6: Based on the Formula (8), get the T_L's matter value f_{T_L} in the backward diffusion.
7: **end for**
8: **for** $S_i \in S$ **do**
9: Based on the Formula (9), get the S_i's matter value F_{S_i} in the backward diffusion.
10: **end for**
11: /* The matter value in the forward diffusion */
12: **for** $T_L \in T$ **do**
13: Based on the Formula (10), get the Z_{T_L}'s matter value Z_{T_L} in the forward diffusion.
14: **end for**
15: **for** $U_k \in U$ **do**
16: Based on the Formula (11), get the U_k's matter value Z_{U_k} in the forward diffusion.
17: **end for**
18: **for** $S_i \in S$ **do**
19: Based on the Formula (12), get the S_i's matter value Z_{S_i} in the forward diffusion.
20: **end for**
21: **for** $S_i \in S$ **do**
22: Based on the Formula (13), get the S_i's composite matter value in the forward/backward matter diffusion.
23: **end for**
24: **return** the recommendation list reclist[C]

---

where $\lambda \in [0, 1]$. The service recommendation result is based on the matter value, and the service which has more matter could be given priority in recommendation.

In the end, we further propose the service recommendation algorithm based on the matter diffusion tripartite graph as shown in Algorithm (2).

In the algorithm, S_i is described as $S_i(f_{S_i}, Z_{S_i}, N'_{S_i}, N''_{S_i}, T', T'')$, of which $1 \leq i \leq x$, f_{S_i} is the initial matter value of service S_i in the backward matter diffusion, Z_{S_i} is the initial matter value of service S_i in the forward matter diffusion, N'_{S_i} is the out-degree of S_i in the forward matter diffusion, T' is the tag set directly connected with S_i in the backward matter diffusion, and T'' is the tag set directly connected with S_i in the forward matter diffusion. The user U_k is described as $U_k(N'_{U_k}, N''_{U_k}, S', S'')$, of which $1 \leq k \leq Y$, N'_{U_k} is the out-degree of user U_k in the backward diffusion, N''_{U_k} is the out-degree of user U_k in the forward diffusion, S' is the service set directly connected with user U_k in the backward matter diffusion, and S'' is the service set directly connected with user U_k in the forward matter diffusion.

To further explain the matter diffusion process in the tripartite graph, we give an example shown in Fig.5. We assume the initial matter values of the services S_1, S_2, S_3 are all 1, the edge weights of the tripartite are all 1, and the forward/backward matter diffusion parameter λ is 0.5. After the forward matter diffusion, the matter values of S_1, S_2, S_3 are 7/8, 3/2 and 5/8, respectively. After the backward matter diffusion, the matter values of S_1, S_2, S_3 are 28/24, 31/24 and 13/24, respectively. We use Formula (13) to get the comprehensive values of S_1, S_2, S_3, which are 49/48, 67/48 and 28/48, respectively. If the recommendation list length is 1, we could choose S_2 as the candidate service.

IV. PERFORMANCE STUDY

In this section, first of all, we present the setup of our simulation. Then, we analyze the performance of service recommendation algorithm based on matter diffusion tripartite graph in different recommendation scenes, and use the precision/recall to measure the effectiveness of the service recommendation scheme. Finally, we find that the proposed algorithm could get good precision/recall when we recommend the service to the user who is familiar with the smart environment.

A. SIMULATION ENVIRONMENT

We build a smart home and the collect the operation records of the user using different electrical appliances. The deployment of electrical equipment and smart switch is shown in Fig.6. The smart home has eleven electrical appliances like light, air conditioner, printer, etc. Because these devices are unintelligent devices, we use the smart switches for on/off/adjustment operation. The switch's control core is microcontroller AT89C51, and the Wi-Fi model is ESP8266 which could be connected to the TXD port and RXD port of SCM for the serial communication. The electrical appliances could be controlled by the Android APP, and the app could communicate with the Wi-Fi model ESP8266.

In two months, we collected one volunteer's 1342 records of on/off/adjustment operations. Moreover, we generated 6530 records of other five users, which were similar to the volunteer's. We further extracted the users' habits and service rules by the methods in [1], [2]. In the same smart home, we chose another volunteer who was similar to the first volunteer, and the similarity is higher than the similar threshold $\propto$ [1]. In two months, we collected 2765 records of the second volunteer. We used the precision and recall to measure the effectiveness of the recommendations, and the definitions of precision and recall are shown in Formula (14), (15).

$$Recall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_u |T(u)|} \quad (14)$$

$$Precision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_u |R(u)|}, \quad (15)$$

where $R(u)$ is the recommendation list in the training set, and $T(u)$ is the operation list in the test set. The precision is the possibility that the user is interested in the recommendation
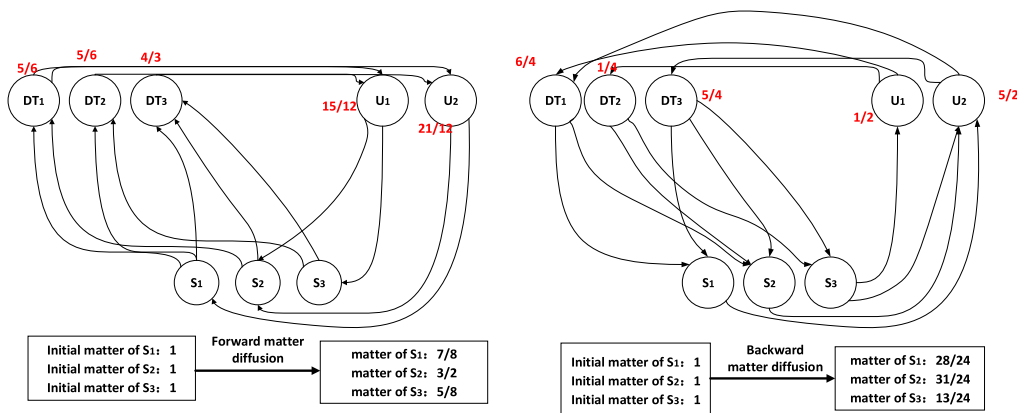

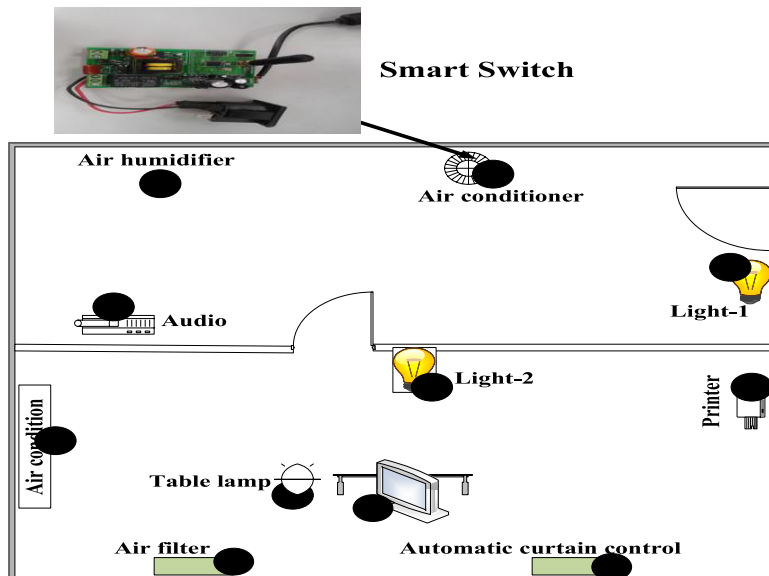**FIGURE 5.** Matter diffusion example in the tripartite graph.

FIGURE 6. The deployment of electrical equipment and smart switch.

service. The recall is the possibility that the user is interested in the service recommended.

In the experiment, we firstly determined the forward and backward diffusion parameter λ and the number of dynamic tags k. Then we used 7872 historical records as a training set, and chose the 2765 records of the second volunteer as the test set. We can see from histogram of precision, λ and k in Fig.7 that the recommendation precision is affected by λ and k. With the increasing of k, the recommendation precision is decreasing. When λ is 0.3 and k is 10, the recommendation precision reaches the maximum 35%. When λ is 0.1 and k is 20, the recommendation precision reaches the minimum 12%. Therefore, in the following experiments, we set λ to be 0.3 and k to be 10.

We divided the test set into two parts by month. In the same test set, we compared the service recommendation arithmetic
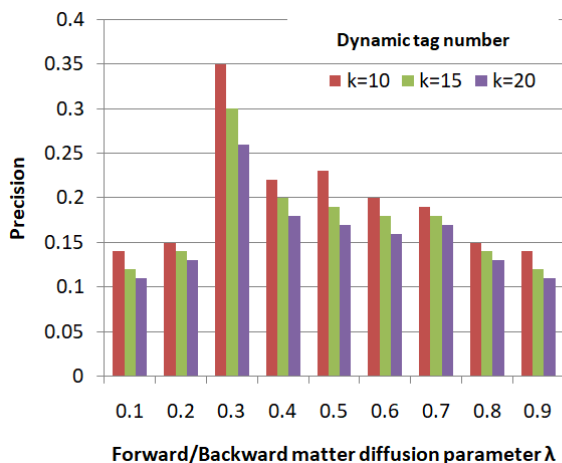
based on matter diffusion tripartite graph with attribute similarity algorithm [35], namely MPSUO and ServRank [36].

The similarities of the attributes between the two users are calculated in the form of the cosine similarity, as shown in Formula (16)

$$q_{ij} = \frac{\lambda |M(u_j) \cap M(s)|}{\sqrt{|M(u_j)| |M(s)|}}, \qquad (16)$$

λ represents the consistency parameter of the must-same attribute set. The similar attribute set of the user S is $M(s)$, and the similar attribute set of the user u_j is $M(u_j)$.

MPSUO is a popular service recommendation algorithm based on service, user and object, which could be described as Formula (17),

$$S(u, o) = \underset{s \in S}{\mathrm{argmax}}(\beta |Y_{s,u}| + (1 - \beta)|Y_{s,o}|) \qquad (17)$$

where $Y_{s,u}$ is the popular service between service and user, $Y_{s,o}$ is the popular service between service and object, and β is the weight. In the experiment, we set β to be 0.5.

ServRank arithmetic is originated from FolkRank which is a service recommendation arithmetic based on tripartite graph. ServRank could be described as Formula (18),

$$w = dAw + (1 - d)p, \qquad (18)$$

where w is the adjacent matrix in the tripartite graph, A is the random value of the normalized column of w, and p is the preference vector. We set d to be *0.7* in the experiment.

B. SIMULATION RESULTS

Fig.8 shows the result of the test set of the first month. When the recall is between 0.15 and 0.28, the attribute similarity algorithm has the highest precision. By analyzing the test set, we find that when the volunteer was just beginning to use the smart home, they tended to use the services similar
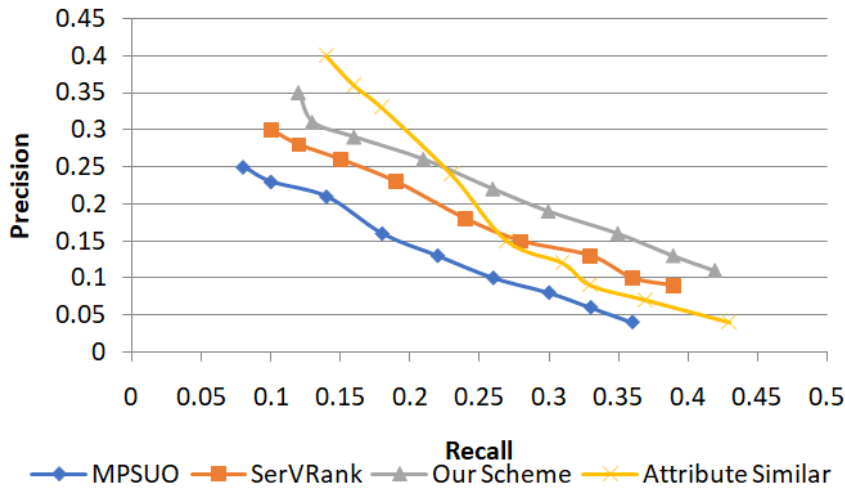

FIGURE 7. The histogram of precision, λ and k.

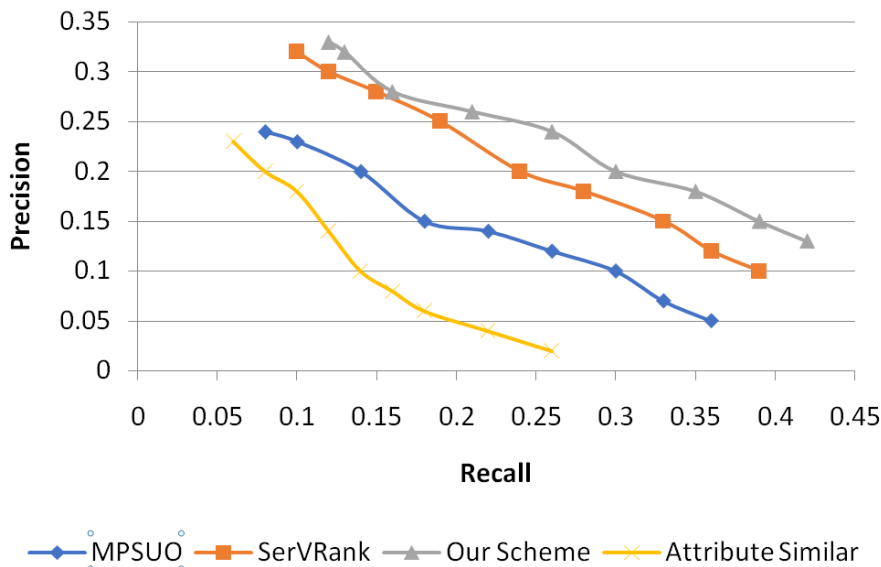**FIGURE 8.** Comparison of accuracy/recall curve of test set 1.

FIGURE 9. Comparison of precision/recall curve of test set 2.

to their own attributes, because he was not familiar with the smart environment. Therefore, the attribute similarity algorithm could achieve good recommendation effect in the early stage. As the volunteer was gradually familiar with the environment, he would select other services which were not similar to his own attributes. Therefore, the yellow curve has a big fluctuation when the recall is between 0.15 and 0.28. We can see from Fig.8 that the gray curve of service recommendation arithmetic based on matter diffusion tripartite graph is higher than MPSUO and SerVRank. Although the service recommendation arithmetic based on matter diffusion tripartite graph does not have better effect than the attribute similarity algorithm in the early stage, its recommended stability is better. The precision/recall of service recommendation arithmetic based on matter diffusion tripartite graph and SerVRank are better than MPSUO, and they are all based on

the tripartite graph. It also shows that the tripartite graph can comprehensively describe the relationship between users and services, realizing more accurate recommendations.

In order to further verify the conclusion, we did the same comparative experiment in the test set of the second month. The experimental result is shown in Fig.9. In the second month, with the recall increasing, the attribute similarity algorithm has the lowest precision compared with the other three recommendation algorithms. The recommendation arithmetic based on matter diffusion tripartite graph and SerVRank has better recommendation results. Therefore, we find that the recommendation algorithm of attribute similarity is more suitable for the new users who just start using smart space. Once the user is familiar with the smart service in the smart home, the recommendation arithmetic based on matter diffusion tripartite graph would have better effect.

V. CONCLUSION

This paper studied the IoT service recommendation strategy, and obtained the recommendation list through the comprehensive calculation of the energy state, the user similarity, the bearing capacity, and the service resource value of the mass diffusion results. The corresponding theory was validated in the real experimental environment. In the following work, the author will continue to study other recommendation methods of the IoT.

REFERENCES

[1] P. Q. Wang, H. Luo, X. M. Li, and Z. W. Zhao, "A new habit pattern learning scheme in smart home," *J. Tamkang Univ. Sci. Technol.*, vol. 19, no. 1, pp. 83–94, 2016.

[2] P. Wang, H. Luo, and Y. Sun, "A habit-based SWRL generation and reasoning approach in smart home," in *Proc. IEEE 21st Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2015, pp. 770–775.

[3] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "QoS-aware Web service recommendation by collaborative filtering," *IEEE Trans. Services Comput.*, vol. 4, no. 2, pp. 140–152, Apr. 2011.

[4] H. Sun, Z. Zheng, J. Chen, and M. R. Lyu, "Personalized Web service recommendation via normal recovery collaborative filtering," *IEEE Trans. Services Comput.*, vol. 6, no. 4, pp. 573–579, Oct. 2013.

[5] J. Liu, M. Tang, Z. Zheng, X. Liu, and S. Lyu, "Location-aware and personalized collaborative filtering for Web service recommendation," *IEEE Trans. Services Comput.*, vol. 9, no. 5, pp. 686–699, Sep. 2016.

[6] L. Yao, Q. Z. Sheng, A. H. H. Ngu, J. Yu, and A. Segev, "Unified collaborative and content-based Web service recommendation," *IEEE Trans. Services Comput.*, vol. 8, no. 3, pp. 453–466, May 2015.

[7] H. Li, F. Cai, and Z. Liao, "Content-based filtering recommendation algorithm using HMM," in *Proc. 4th Int. Conf. Comput. Inf. Sci.*, Aug. 2012, pp. 275–277.

[8] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 325–341.

[9] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 73–105.

[10] M. R. McLaughlin and J. L. Herlocker, "A collaborative filtering algorithm and evaluation metric that accurately model the user experience," in *Proc. 27th Annu. Int. Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2004, pp. 329–336.

[11] J. Cao, Z. Wu, Y. Wang, and Y. Zhuang, "Hybrid collaborative filtering algorithm for bidirectional Web service recommendation," *Knowl. Inf. Syst.*, vol. 36, no. 3, pp. 607–627, Sep. 2013.

[12] Y. Zhang and Y. Liu, "A collaborative filtering algorithm based on time period partition," in *Proc. 3rd Int. Symp. Intell. Inf. Technol. Secur. Informat.*, Apr. 2010, pp. 777–780.

[13] H. Wu, Y.-J. Wang, Z. Wang, X.-L. Wang, and S.-Z. Du, "Two-phase collaborative filtering algorithm based on co-clustering: Two-phase collaborative filtering algorithm based on co-clustering," *J. Softw.*, vol. 21, no. 5, pp. 1042–1054, May 2010.

[14] Y. Park, S. Park, W. Jung, and S.-G. Lee, "Reversed CF: A fast collaborative filtering algorithm using a k-nearest neighbor graph," *Expert Syst. Appl.*, vol. 42, no. 8, pp. 4022–4028, May 2015.

[15] S. Lo and C. Lin, "WMR–a graph-based algorithm for friend recommendation," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell.*, Dec. 2006, pp. 121–128.

[16] N. B. Silva, I.-R. Tsang, G. D. C. Cavalcanti, and I.-J. Tsang, "A graph-based friend recommendation system using genetic algorithm," in *Proc. IEEE Congr. Evol. Comput.*, Jul. 2010, pp. 1–7.

[17] Z. Wang, M. Zhang, Y. Tan, W. Wang, Y. Zhang, and L. Chen, "Recommendation algorithm based on graph-model considering user background information," in *Proc. 9th Int. Conf. Creating, Connecting Collaborating Through Comput.*, Jan. 2011, pp. 32–39.

[18] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, pp. 1–19, Oct. 2009.

[19] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperat. Work (CSCW)*. New York, NY, USA: ACM, 1994, pp. 175–186.

[20] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan. 2003.

[21] T. Hofmann, "Latent semantic models for collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 89–115, Jan. 2004.

[22] L. Si and R. Jin, "Flexible mixture model for collaborative filtering," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 704–711.

[23] A. Hotho *et al.*, "Information retrieval in Folksonomies: Search and ranking," in *Proc. Eur. Semantic Web Conf.* Berlin, Germany: Springer, 2006.

[24] Z.-K. Zhang, T. Zhou, and Y.-C. Zhang, "Tag-aware recommender systems: A State-of-the-Art survey," *J. Comput. Sci. Technol.*, vol. 26, no. 5, pp. 767–777, Sep. 2011.

[25] S. Hamouda and N. Wanas, "PUT-tag: Personalized user-centric tag recommendation for social bookmarking systems," *Social Netw. Anal. Mining*, vol. 1, no. 4, pp. 377–385, Nov. 2011.

[26] L. B. Marinho and L. Schmidt-Thieme, "Collaborative tag recommendations," in *Data Analysis, Machine Learning and Applications*. Berlin, Germany: Springer, 2008, pp. 533–540.

[27] S. Rendle and L. Schmidt-Thieme, "Pairwise interaction tensor factorization for personalized tag recommendation," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining (WSDM)*. New York, NY, USA: ACM, 2010, pp. 81–90.

[28] L. Yao, Q. Z. Sheng, A. H. H. Ngu, and X. Li, "Things of interest recommendation by leveraging heterogeneous relations in the Internet of Things," *ACM Trans. Internet Technol.*, vol. 16, no. 2, pp. 1–25, Mar. 2016.

[29] L. Yao, Q. Z. Sheng, A. H. H. Ngu, H. Ashman, and X. Li, "Exploring recommendations in Internet of Things," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*. New York, NY, USA: ACM, 2014, pp. 855–858.

[30] I. Mashal, O. Alsaryrah, and T.-Y. Chung, "Testing and evaluating recommendation algorithms in Internet of Things," *J. Ambient Intell. Humanized Comput.*, vol. 7, no. 6, pp. 889–900, 2016.

[31] Z. Chen, R. Ling, C.-M. Huang, and X. Zhu, "A scheme of access service recommendation for the Social Internet of Things," *Int. J. Commun. Syst.*, vol. 29, no. 4, pp. 694–706, 2016.

[32] D. Hussein, S. N. Han, G. M. Lee, and N. Crespi, "Social cloud-based cognitive reasoning for task-oriented recommendation," *IEEE Cloud Comput.*, vol. 2, no. 6, pp. 10–19, Nov. 2015.

[33] H. Liang, Y. Xu, Y. Li, R. Nayak, and X. Tao, "Connecting users and items with weighted tags for personalized item recommendations," in *Proc. 21st ACM Conf. Hypertext Hypermedia (HT)*. New York, NY, USA: ACM, 2010, pp. 51–60.

[34] R. Srikant and J. F. Naughton, *Fast Algorithms for Mining Association Rules and Sequential Patterns*. Madison, WI, USA: Diss. University of Wisconsin, 1996.

[35] P. Wang, H. Luo, and Y. Sun, "IOT service recommendation strategy based on attribute relevance," in *Proc. 11th Springer Int. Conf. Ubiquitous Comput. Ambient Intell. (UCAmI)*, 2017, pp. 34–43.

[36] I. Mashal, O. Alsaryrah, and T.-Y. Chung, "Analysis of recommendation algorithms for Internet of Things," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2016, pp. 1–6.

[37] B. Cheng, M. Wang, S. Zhao, Z. Zhai, D. Zhu, and J. Chen, "Situation-aware dynamic service coordination in an IoT environment," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2082–2095, Aug. 2017.

[38] B. Cheng, S. Zhao, J. Qian, Z. Zhai, and J. Chen, "Lightweight service mashup middleware with REST style architecture for IoT applications," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 3, pp. 1063–1075, Sep. 2018.

[39] B. Cheng, D. Zhu, S. Zhao, and J. Chen, "Situation-aware IoT service coordination using the event-driven SOA paradigm," *IEEE Trans. Netw. Service Manage.*, vol. 13, no. 2, pp. 349–361, Jun. 2016.

PINGQUAN WANG received the Ph.D. degree from the School of Computer Science, Beijing University of Posts and Telecommunications. He currently works as a Teaching Fellow with the School of Computer and Information Engineering, Inner Mongolia University of Finance and Economics. His research interests include the Internet of things, blockchain, and cloud computing.

• • •