

Research Article

Location-Aware POI Recommendation for Indoor Space by Exploiting WiFi Logs

Zengwei Zheng,¹ Yuanyi Chen,² Sinong Chen,¹ Lin Sun,¹ and Dan Chen¹

¹Hangzhou Key Laboratory for IoT Technology & Application, Zhejiang University City College, Hangzhou, China

²Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

Correspondence should be addressed to Yuanyi Chen; cyyxz@sjtu.edu.cn

Received 11 March 2017; Revised 28 May 2017; Accepted 14 June 2017; Published 31 July 2017

Academic Editor: Michele Ruta

Copyright © 2017 Zengwei Zheng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Indoor shopping trajectories provide us with a new approach to understanding user's behaviour pattern in urban shopping mall, which can be derived from user-generated WiFi logs using indoor localization technology. In this paper, we propose a location-aware Point-of-Interest (POI) recommendation service in urban shopping mall that offers a user a set of indoor POIs by considering both personal interest and location preference. The POI recommendation service cannot only improve user's shopping experience but also help the store owner better understand user's shopping preference and intent. Specifically, the proposed method consists of two phases: offline modelling and online recommendation. The offline modelling phase is designed to learn user preference by mining his/her historical shopping trajectories. The online recommendation phase automatically produces top- k recommended POIs based on the learnt preference. To demonstrate the utility of our proposed approach, we have performed a comprehensive experiment evaluation on a real-world dataset collected by 468 users over 33 days. The experimental results show that the proposed recommendation service achieves much better recommendation performance than several existing benchmark methods.

1. Introduction

Indoor location-based services, such as shopping flow monitoring, mobile location-based advertisement, and POI recommendation, are expected to witness a significant growth in the next decade due to the popularity of mobile devices and the development of indoor positioning technologies. Previous studies about this topic mainly focus on providing some basic services, such as indoor positioning [1], indoor navigation [2], or indoor tracking [3]. By contrast, few studies aim to perform in-depth analysis and utilize user's location information in indoor environment, which is a fundamental context for location-based services. For instance, the retention time customers spend in visiting shops, the way customers come to these shops (e.g., go directly to a shop or just hanging out in the shopping mall), and the frequency of customers checking in shops, something like that, are useful for understanding user's behaviour pattern and preference.

Similar to online behaviour analysis in E-commerce, this kind of in-depth location analysis is also called physical analytics [4], which is demonstrated as a revolutionary

new technology for connecting consumers with shop brand. Specifically, physical analytics in a shopping mall can provide three sides benefit: (1) For users side, some context-aware personal services (e.g., personalized recommendation and optimal shopping route) can be provided based on their preference deriving from physical analytics. (2) For shop owner side, physical analytics are beneficial to targeted advertising since potential consumers can be found based on their preference. (3) For shopping mall manager side, physical analytics can monitor traffic flow in real time and discover correlation between shops and users, which is useful for optimizing shopping mall layout. For E-commerce recommendation, website cookies are sufficient to learn user's preference. However, physical analytics will suffer serious challenge due to the difficulty of obtaining user's behaviour information, such as user's shopping trajectory. Even worse, user's location information in indoor environment is usually incomplete and scattered. Fortunately, WiFi check-in logs provide a new platform to generate user's trajectory in indoor environment since free WiFi is increasing available for many indoor spaces, such as urban shopping mall and

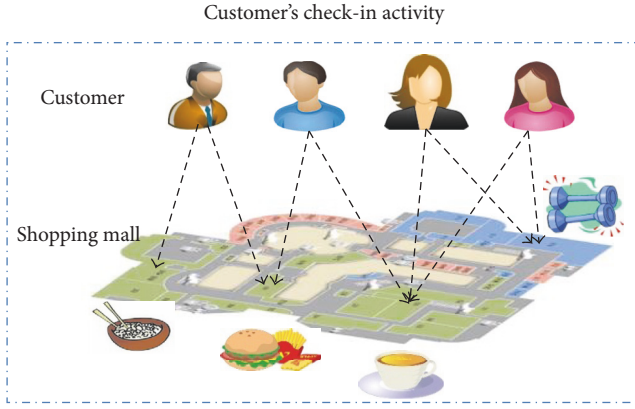


FIGURE 1: An example of customer's check-in activities in urban shopping mall.

museum. Additionally, customer's check-in activities imply their preference; since most people have a finite amount of resources (e.g., time and money), they tend to visit their favourite shops (as shown in Figure 1).

As mobile devices and social media are becoming more and more pervasive, user-generated information (e.g., check-in records) from these platforms is providing rich information to in-depth understand user preference. Recently, a few studies [5, 6] on POI recommendation from location-based social network (LBSN) have been proposed; these approaches will suffer several challenges. The first challenge is that it is insufficient to learn customer preference by only utilizing check-in frequency; other pieces of context information (such as the residence time of visiting) can help better reflect the level of customer's interest towards a shop. The second challenge is data sparsity due to few check-ins in reality (most customers usually visit a shop only once in fact), so only utilizing customer's check-in frequency may not be sufficient to learn their preferences. Moreover, existing POI recommendation methods in LBSN cannot make recommendations for people who are not members of the LBSN.

To tackle these challenges, we propose an indoor POI recommendation method for urban shopping mall with the following three contributions:

- (i) It generates user's indoor spatial-temporal trajectory from user-generated WiFi logs. The proposed approach first recognizes POI entrances by utilizing the WiFi radio signal strength fluctuation of a small window size when passing a physical boundary point and then splits WiFi logs to a few subsequences according to these identified POI entrances. Finally, the approach maps each subsequence to POI based on indoor fingerprint-based localization.
- (ii) It utilizes a two-layer relation graph to capture multi-relation among users and POIs from user's historical spatiotemporal trajectories. Then it estimates *User-User* relation by utilizing a random walk-based propagation algorithm and performs indoor POI recommendation using user-based collaborative filtering.

- (iii) We evaluate our method on indoor POI recommendation using a real-world dataset collected by 468 users over 33 days, with performances outperforming six baselines.

Since it is insufficient to learn user's preference by merely utilizing their check-in frequency, our approach extracts various context information (such as the check-in frequency and check-in time) from user's indoor spatial-temporal trajectory for mining their preference. In addition, our approach collects user-generated WiFi logs by passive crowdsourcing, which is infrastructure-free and no user involvement. Therefore, our proposed approach can make recommendation for users who are not members of the LBSN. To deal with the data sparsity challenge, our approach infers *User-User* relation utilizing a random walk-based propagation algorithm. Random walk on graph can alleviate the sparsity problem in indoor POI recommendation by utilizing both *User-Store* relation and *User-User* relation. Typically, most users have few check-ins and tend to review a small number of POIs; thus the data of directly connected vertices (e.g., *User-POI* vertex pairs and *User-User* vertex pairs) are sparse. Fortunately, one vertex can reach another vertex through intermediate vertices (denote as hidden propagation path), which can better estimate the relation strengths between two vertices that are not directly connected with these hidden propagation paths.

The remainder of the paper is organized as follows. Section 2 surveys related work on mining indoor trajectories and indoor POI recommendation. Section 3 introduces our proposed approach for generating indoor trajectories using WiFi RSSI. Section 4 describes our indoor POI recommendation algorithm in detail. Section 5 reports the experimental results. Finally, we present our conclusion and future work in Section 6.

2. Related Work

In this section, we survey related works and discuss how these works differ from our research, including previous studies of generating indoor trajectory and indoor POI recommendation.

2.1. Generating Indoor Trajectory. We are aware of only a few works [7–10] which directly involve this issue that generates user's trajectory in indoor space. Specifically, Prentow et al. [7] proposed a bootstrapping approach to construct indoor trajectory for mitigating indoor positioning error biases. Werner et al. [8] utilized WiFi RSSI as an information source to infer indoor trajectory from a given trajectories database. Radaelli et al. [9] utilized sequential pattern mining to identify typical trajectories of indoor objects by exploiting presence sensors, that is, Bluetooth and RFID. Dakkak et al. [10] presented a method to generate indoor trajectories using classical predictors and digital fractional differentiation.

Fortunately, there are a number of works on indoor tracking which are relevant to generate indoor trajectories, as it is simple to obtain indoor trajectory from movement tracking data. In general, existing studies on indoor tracking can be divided into three categories: (1) Indoor tracking

by location fingerprinting-based approaches [2, 11]. The key idea is to obtain user's location using fingerprinting-based positioning algorithm, which usually consists of two phases: offline constructing fingerprint map and online positioning. (2) Indoor tracking by triangulation positioning approaches [3]. The main idea is to obtain user's location using at least three anchors for triangulation positioning. (3) Indoor tracking by dead reckoning approaches [12], which calculates user's current position according to previously determined position using built-in sensors (e.g., gyroscope, accelerometer, and compass) of mobile devices.

However, fingerprinting-based indoor tracking is time-consuming and vulnerable to many factors, such as heterogeneous devices or environmental changes. Triangulation positioning will result in heavy infrastructure cost since it requires anchors to coverage the indoor space. Dead reckoning approaches rely on the initial location and thus suffer from cumulative error.

2.2. Indoor POI Recommendation. To the best of our knowledge, only a few literatures [13–15] address the problem of indoor POI recommendation based on user's trajectories. Specifically, Lin [13] proposed an indoor location system by regarding the stay time in certain shops as item rating. Fang et al. [15] mined customer's preference from WiFi RSSI patterns, that is, time spent in a store and check-in frequency of store, and then proposed a recommendation system for indoor shopping. Jin et al. [14] proposed an indoor hotspots detecting method by considering user's interests in locations and the relationship between users and locations.

We also surveyed related works about indoor recommendation by considering user's context. For instance, [16] proposed an store recommendation systems by mining the context of decision-making behaviour using eye-tracking data, [17] proposed a POI discovery approach by matching the user profile and the semantic-enhanced POIs, [18] proposed a recommended system to help users in shopping for technical products by considering user preference and technical product attributes, and [19] proposed an automatic mobile assistant for museum visiting based on WiFi-based indoor positioning. Additionally, Shin et al. [20] constructed an indoor database platform for indoor location-based services.

Our proposed indoor POI recommendation method differs from the above-mentioned works in the following three aspects: (1) We generate user's indoor trajectories using opt-in WLAN without needing time-consuming and labor intensive cost for constructing fingerprinting map for each small grid of indoor space, such as in [2, 11]. (2) We only use the indoor trajectories to learn user's preferences for making recommendations, unlike literatures [17, 18] that need additional user profiles for recommendation. (3) Existing indoor POI recommendation algorithms, such as [7–10], merely use user-based or item-based collaborative filtering for making recommendation and will suffer data sparsity problem since numerous users only have few check-in information. To address this challenge, we first utilize a two-layer relation graph to capture multirelation among users and POIs from user's historical trajectories. Then, we infer

User-User relation with a random walk-based propagation algorithm and perform indoor POI recommendation using user-based collaborative filtering.

3. Generate Indoor Spatiotemporal Trajectory

Unlike outdoor trajectories that can be easily obtained from a large number of user-generated GPS trajectories, user's indoor moving trajectories are significantly difficult to obtain. For generating user's indoor trajectory, our approach utilizes WiFi RSSI by considering the widespread deployment of WLAN infrastructure in indoor space nowadays. We formally describe our method for generating indoor trajectory as follows.

3.1. Problem Definition. For ease of the following presentation, we first define the key data structures and notations used in the problem of generating indoor spatiotemporal trajectory.

Definition 1 (indoor POI). $P = \{p_1, p_2, \dots, p_N\}$ denotes the set of POIs. N is the number of POIs and POI p_i refers to an indoor geographical region that may be useful or interesting for a user.

Definition 2 (indoor spatiotemporal trajectory). An indoor spatiotemporal trajectory is a sequence of indoor POIs consecutively visited by a user, which is defined as $\text{Traj}_u = \langle q_u^1 \rightarrow q_u^2 \rightarrow \dots \rightarrow q_u^K \rangle$ and K is the POIs numbers that are visited by user u in one visiting and $q_u^i = \langle u, p_i, ts \rangle$ is a triple that means user u checks in POI p_i at a particular timestamp ts .

In short, user's visiting history can be regarded as a set of indoor trajectories. Clearly, user's trajectories imply spatial and temporal information.

Definition 3 (POI feature). POI feature is defined as a tuple $f_{p_i} = (p_i, R_{p_i})$. R_{p_i} is M -dimensional vector and is denoted by (r^1, r^2, \dots, r^M) , which means the scanned WiFi RSSI recorded from surrounding WiFi APs in p_i , and M is the number of WiFi access points (APs) in indoor space.

According to the principle of indoor fingerprint-based localization [21], the WiFi RSSI collected in a physical place can be regarded as the location landmark for positioning.

Definition 4 (POI feature set). POI feature set is the feature collection of all POIs, denoted by $F = \{f_{p_1}, f_{p_2}, \dots, f_{p_N}\}$.

Definition 5 (WiFi logs). A user-generated WiFi log is defined as $S = \langle s_1, s_2, \dots, s_n \rangle$, and $s_i = (t_i, R_{t_i})$ denotes the scanned WiFi RSSI record by user's mobile device at time t_i , $1 \leq i \leq n$.

As mentioned above, our approach utilizes existing WiFi infrastructure to generate user's interaction behaviours from WiFi logs, which is infrastructure-free and no user involvement. To achieve both goals, we utilize WiFi probe requests to collect the data with a nonintrusive way. WiFi probe requests are frames that are broadcast by mobile phones to discover

TABLE 1: The RSSI records when people walk through a POI entrance.

Records	AP1	AP2	AP3	AP4	AP5	AP6	AP7	AP8
R1	-69	-56	-87	-95	-62	-71	-48	-94
R2	-64	-61	-80	-92	-51	-65	-52	-91
R3	-71	-52	-75	-86	-55	-61	-63	-89
R4	-68	-54	-63	-80	-60	-59	-69	-82
R5	-74	-84	-67	-84	-68	-77	-87	-71
R6	-85	-92	-71	-65	-86	-81	-89	-62
R7	-92	-90	-65	-60	-92	-85	-92	-59

nearby WiFi APs and can be sniffed by WiFi compatible antennas on 802.11b/g/n channels. According to [22], mobile phones will broadcast WiFi probe requests every few seconds. That means collecting data by WiFi probe requests allowed us to track every mobile device that connects to the WiFi infrastructure. In our experiment, every device that connects to WLAN infrastructure at each POI has agreed to this data collection as part of the sign-on agreement. For privacy issue, we collect user-generated WiFi logs as hashed entities with no additional knowledge about them and finish collecting data when the user leaves the shopping mall. We believe that this is a privacy-safe application.

Based on the above definitions, we formulate the problem of generating user's indoor spatiotemporal trajectory as follows: given a user-generated WiFi logs S in one visiting and POI feature set $F = \{f_{p_1}, f_{p_2}, \dots, f_{p_N}\}$, obtain the corresponding indoor spatiotemporal trajectory $\text{Traj} = \langle q^1, q^2, \dots \rangle$.

3.2. Solving Approach. The idea behind our solving approach is to utilize the WiFi RSSI fluctuation of a small time window when passing a physical boundary point. The principle is that WiFi RSSI is relatively stable in a small area according to the indoor propagation model of wireless signal [23]. However, some indoor physical constraints, that is, POI's entrances and stairs, will cause WiFi RSSI change dramatically even in a small area. Table 1 shows that the RSSI sequences extracted from a small time window include 7 WiFi RSSI records when a user walks through a POI entrance; the location for collecting $\{R_1, R_2, R_3\}$ is outside the POI, nearby the POI entrance when collecting $\{R_4, R_5\}$, and inside POI when collecting $\{R_6, R_7\}$. We further calculate the Euclidean distance between two adjacent records as follows: $\text{Dist} = \{26, 4, 17, 77, 19, 4\}$. Obviously, the WiFi RSSI will have a dramatic "jump" when people walk through the POI entrance.

Based on the above analysis, our approach generates user's indoor spatiotemporal trajectory by the following two steps.

Step 1 (recognize POI entrance). Given a user-generated WiFi logs $S = (\langle t_1, R_{t_1} \rangle, \langle t_2, R_{t_2} \rangle, \dots, \langle t_n, R_{t_n} \rangle)$, we define $F(t_i, h)$ as the WiFi RSSI variation in time window $(t_i - h, t_i + h)$, as shown in

$$F(t_i, h) = \frac{1}{M} \sum_{i=1}^M D(R^i), \quad (1)$$

where M is the number of WiFi APs and $D(R^i)$ is the variation of WiFi RSSI from WiFi AP i during this time window, as calculated in

$$D(R^i) = \frac{1}{2h+1} \sum_{j=t_i-h}^{t_i+h} (R_j^i - \bar{R}^i)^2, \quad (2)$$

where \bar{R}^i is the average WiFi RSSI from AP i and R_j^i is the WiFi RSSI from AP i at timestamp j .

Based on the WiFi RSSI variation, we can obtain the time t_i when user may pass a POI entrance if the WiFi RSSI variation in time segment $(t_i - h, t_i + h)$ is larger than a threshold θ .

As mentioned above, we recognize indoor POI entrance using the WiFi RSSI jump characteristic, which may bring some false recognitions, since other factors (e.g., crowd passing and room layout change) may result in similar RSSI jump characteristic. However, the WiFi RSSI jump characteristic caused by these factors (e.g., crowd passing and room layout change) is temporary, while that caused by POI entrances is stable (there is an obvious RSSI jump characteristic when users pass through a POI entrance). Therefore, we utilize a user-specific threshold method to remove false recognitions. Formally, let \bar{R}^i as the scanning WiFi RSSI record that user may walk through a POI entrance, and θ and τ are the user-specific threshold. We consider \bar{R}^i as a false recognition if

$$I(\|\bar{R}^i, R_j\| < \theta) < \tau, \quad (3)$$

where $\|\bar{R}^i, R_j\|$ is the Euclidean distance between \bar{R}^i and R_j and $I(\|\bar{R}^i, R_j\| < \theta)$ means the number of RSSI records that the Euclidean distance between \bar{R}^i and R_j is smaller than θ .

Step 2 (map WiFi RSSI to POI). After obtaining time set $T = \{t_p, t_{p+1}, \dots, t_q\}$ when user walks through POIs entrance, we first split the WiFi logs S into a subsequence set $S = \{S_{t_1:t_p}, S_{t_p:t_{p+1}}, \dots, S_{t_{q-1}:t_q}, S_{t_q:t_n}\}$ and then map each subsequence to the corresponding POI using POI feature set. Finally, we construct indoor spatiotemporal trajectory according to Definition 2.

The framework of generating indoor spatiotemporal trajectory is described in Algorithm 1.

TABLE 2: Notations used in indoor POI recommendation.

Symbol	Description
P, U, TS	The set of POIs, users, and timestamps
N, M	The number of POIs and WiFi APs
φ_{up}	The visiting interaction between user u and POI p
Tra_j	A trajectory of user u
I_u	The interest of user to POIs
w_{ui}	The relation strength of user u to POI p_i
$\text{sim}(u_i, u_j)$	The similarity of two user's interest

4. Indoor POI Recommendation Algorithm

In this section, we first introduce the key data structures and notations used in our POI recommendation algorithm and then present the offline modelling part and online recommendation part of the proposed algorithm.

4.1. Preliminary. For ease of the following presentation, we define the key data structures and notations used in the proposed approach. Table 2 lists the relevant notations used in this paper.

Definition 6 (User-POI visiting interaction). A visiting interaction is a triple $\varphi_{up} = \langle u, p, ts \rangle$, which means user u visits POI p at a particular timestamp ts . Information about the user and POI history interaction is given by $\{\langle u, p, ts \rangle \mid u \in U \wedge p \in P \wedge ts \in TS\}$.

Note that the interaction between user and POI carries three-dimensional information: user, POI, and timestamp, which can provide rich information to learn the latent relationship between user and POI. Typically, different users have different interesting POIs in indoor environment. For instance, customers choose to visit shops according to their income level or visit clothing shops based on personal dress style. The timestamp of interactions carries certain behaviour patterns; that is, people usually eat lunch at 12:00 p.m.–14:00 p.m., which means that a customer has a higher probability to check in a restaurant rather than a clothing store at this timestamp. In addition, the stay time when a customer visits different shops implies her preference since most users have finite resources; the more time spent in a shop indicates the more interesting. Obviously, the time dimension information can help to better recommend POIs to users.

Definition 7 (User-POI relation). The relation strength indicates the interest of a user u for POIs, which is defined as a vector $I_u = \{w_{u1}, \dots, w_{ui}, \dots, w_{uN}\}$, where $0 \leq w_{ui} \leq 1$.

Definition 8 (User-User relation). The relation strength $\text{sim}(u_i, u_j)$ indicates the similarity of two user's POI interest.

Definition 9 (relation graph). We denote the graph by $G = \langle V, E \rangle$, which is an undirected bipartite graph. Here $V = U \cup P$, where U and P are the sets of users and POIs, respectively. Edges $E = E_U \cup E_P$, where $E_U = \{(u, u') : (u, u') \in U \times U\}$ represents the relation between users, and

$E_P = \{(u, p) : (u, p) \in U \times P\}$ represents the relation strength between users and POIs.

4.2. Offline Modelling. In this subsection, we first describe the offline modelling part of the proposed method, a graph-based model to capture the relation between *User-User* and *User-POI*, and then present the inference process. Our goal of the inference process is to derive the pairwise relevance scores for each pair of *User-User*. To achieve this, a random walk with restart (RWR) [24] is performed on the constructed relation graph, and the value of relevance scores reflects the relation strength of users.

4.2.1. Relation Graph Construction. Our approach for constructing relation graph is comprised of two stages.

In the first stage, we calculate the initial relation strength of *User-POI* and *User-User* from user's visiting trajectories. More exactly, the two kinds of relation strength calculation are described as follows.

User-POI Relation Calculation. We calculate the initial relation strength of a user to POI with considering two factors: one is the POI check-in frequency and another is the average stay time of this POI. Obviously, the more times user visits a POI and the longer stay time demonstrate the more interest for this POI. Therefore, we calculate the *User-POI* relation strength as shown in

$$I(u, p_i) = \sum_{k=1}^{C_u} \frac{T_k(u, p_i)}{\max(T_k(u))}, \quad (4)$$

where $T_k(u, p_i)$ is the stay time of user u in POI p_i at trajectory T_k , $\max(T_k(u))$ is the longest POI stay time of user u at trajectory T_k , and C_u is the history trajectories of u .

User-User Relation Calculation. Following the principle of collaborative filtering, we use Pearson's correlation [25] to measure *User-User* relation strength based on *User-POI* relation, and the relation strength $\text{sim}(u_i, u_j)$ is defined as

$$\begin{aligned} \text{sim}(u_i, u_j) &= \frac{\sum_{p_k \in p_{ij}} (I(u_i, p_k) - \overline{I(u_i)}) (I(u_j, p_k) - \overline{I(u_j)})}{\sqrt{\text{var}(u_i)} \cdot \sqrt{\text{var}(u_j)}}, \end{aligned} \quad (5)$$

where $\overline{I(u_i)}$ and $\overline{I(u_j)}$, respectively, represent the average relation strength that users u_i and u_j have given to all POIs, p_{ij} represent the POIs that are visited by both users u_i and u_j , and $\text{var}(u_i)$ and $\text{var}(u_j)$ are calculated according to

$$\begin{aligned} \text{var}(u_i) &= \sum_{p_k \in p_{ij}} (I(u_i, p_k) - \overline{I(u_i)})^2, \\ \text{var}(u_j) &= \sum_{p_k \in p_{ij}} (I(u_j, p_k) - \overline{I(u_j)})^2. \end{aligned} \quad (6)$$

At the second stage, we model the POIs, users, and *User-POI* interaction in a relation graph based on the two kinds

Require: (1) User-generated WiFi logs $S = \langle s_1, s_2, \dots, s_n \rangle$; (2) POI fingerprint set $F = (f_1, f_2, \dots, f_N)$,
Ensure: Indoor spatio-temporal trajectory Traj.
(1) Initial user trajectory Traj = \emptyset .
(2) Obtain the time set $T = \{t_p, t_{p+1}, \dots, t_q\}$ that user walks through POIs entrance according to RSSI variation.
(3) Split S into a subsequence set $S = \{S_{t_0:t_p}, S_{t_p:t_{p+1}}, \dots, S_{t_{q-1}:t_q}\}$.
(4) **for** $\forall S_{t_k:t_{k+1}} \in S$ **do**
(5) Initial POI set $P = \emptyset$.
(6) **for** $\forall t_i \in (t_k, t_{k+1})$ **do**
(7) Obtain the corresponding POI p at time t_i using nearest neighbour.
(8) Add p to P .
(9) **end for**
(10) Find the element p_k with the most times in P .
(11) Obtain the timestamp $\Delta t = t_{k+1} - t_k$.
(12) Construct triple $q^k = \langle u, p_k, \Delta t \rangle$ and add q^k to trajectory Traj.
(13) **end for**
(14) **return** Indoor Spatio-temporal trajectory Traj.

ALGORITHM 1: Generating indoor spatiotemporal trajectory.

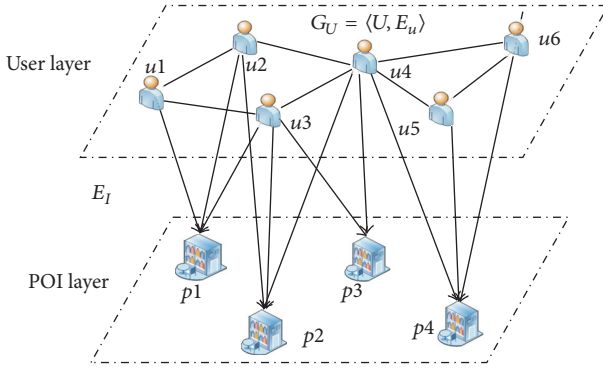


FIGURE 2: Two-layer relation graph using historical spatiotemporal trajectories.

of relation strengths obtained from the first stage, as shown in Figure 2. The relation graph contains a user layer and a POI layer; each node in the user layer represents a user of which the edge between any two users represents their relation strength. Each node in the POI layer represents a POI while the edge between a user and a POI represents the *User-POI* relation strength. We formally described the relation graph construction as follows.

As described in Definition 9, we construct the relation graph including user layer and POI layer. The edge weight in the user layer is calculated according to (5), while the edge weight between user and POI is calculated according to (4). We formulate the weight matrix W of graph G as (7), where W_U and W_I are the weight matrix of edges E_U and E_I , respectively. Since we do not consider the relation between POIs, W_P is a zero matrix.

$$W = \begin{bmatrix} W_U & W_I \\ W_I^T & W_P \end{bmatrix}. \quad (7)$$

4.2.2. User-User Relation Inference. Existing indoor POI recommendation algorithms, that is, user-based [13] method or item-based method [15], will suffer serious data sparsity problem since numerous users only have few POI check-ins. In addition, few historical trajectories cannot reflect user's POI preference and further infer reasonable *User-POI* relation and thus may result in cold-start problem in recommendation system. To address this challenge, we perform the random walk with restart to derive the relation between each pair of users. This method first infers transition probabilities between users based on their similarities and models finite length random walks on the user space to compute predictions, which is especially useful when training data is less than plentiful, that is, when typical similarity measures fail to capture actual relationships between users (such as data sparsity challenge). On the other hand, our method has resemblance to the studies in [26, 27] in the manner by which they deal with sparsity problem. The goal of RWR in our approach is to find neighbour users with *top-K* highest relevance score for a given user based on their historical POI check-ins.

Random walk on relation graph can alleviate the sparsity problem in indoor POI recommendation by fusing multirelation between users and POIs. Users usually have few check-ins for most POIs; thus the directly connected vertices are sparse. However, one vertex can reach another vertex through hidden propagation path, which can better estimate the relation strength between two vertices that are not directly connected with these hidden propagation paths. The intuition on how hidden propagation paths can alleviate sparsity can be explained by the following example.

Example 10. Suppose we need to infer the relation between two vertices: u_1 and u_4 (as shown in Figure 3), which is not directly connected in the relation graph. Since u_1 has an

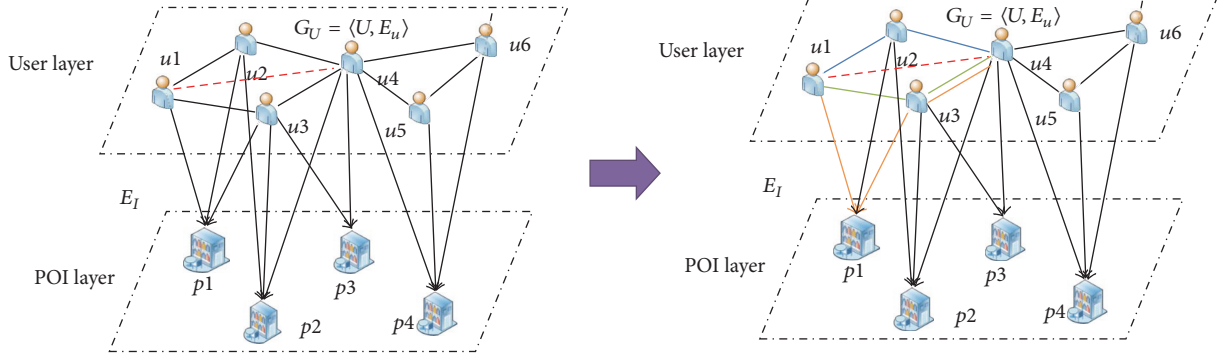


FIGURE 3: Example of inferring the relation strength between two users by hidden relationships.

edge with u_2 and u_3 , respectively, we can find some hidden relations between u_1 and u_4 through intermediate vertices (e.g., p_1). More exactly, we can find one hidden propagation path $\langle u_1 \rightarrow p_1 \rightarrow u_3 \rightarrow u_4 \rangle$ from the *User-POI* relation and two hidden propagation paths $\langle u_1 \rightarrow u_2 \rightarrow u_4 \rangle$ and $\langle u_1 \rightarrow u_3 \rightarrow u_4 \rangle$ from the *User-Store* relation. Such hidden propagation paths are beneficial to infer the relation strength of two vertices that are not connected directly, thus alleviating the data sparsity problem.

Without loss of generality, we assume the random walker starts from a user node u_i on graph G . Then, the random walker iteratively transmits to other nodes which have edges with u_i , with the probability that is proportional to the edge weight between them. At each step, u_i also has a restarting probability λ to return itself. We can obtain the steady-state probability of u_i by visiting other vertexes when the RWR process is converged. The RWR process can be formulated as

$$Y_{t+1} = (1 - \lambda) M_p Y_t + \lambda X, \quad (8)$$

where Y and X are two $R^{N \times 1}$ vectors, X is the initial relation strength of target user u_i to other users and calculated as (5), Y_t represents the probability distribution in step t and $Y_0 = X$, and M_p is the transition matrix, which is obtained based on weight matrix W of G by now normalization, as shown in

$$M_p = WD^{-1}, \quad (9)$$

where D is a diagonal matrix with $D(i, i) = \sum_j W(i, j)$.

4.3. Online Recommendation. After constructing the relation graph, the problem of recommendation POIs to a target user u_i can be converted to calculate the relation strength between vertex u_i and the unvisited POIs and then generate POI recommendation lists according to the top- k relation strength rank.

Following the principle of collaborative filtering, we calculate the relation strength between vertex u_i and unvisited POI p_k as

$$I(u_i, p_k) = \sum_{u_j \in U} \text{sim}(u_i, u_j) \left(I(u_j, p_k) - \overline{I(u_j)} \right), \quad (10)$$

TABLE 3: Statistics of shop categories.

Restaurant	Fashion	Leisure	Fitness
19	23	8	10

where $\text{sim}(u_i, u_j)$ represents the relation strength between u_i and u_j , $I(u_j, p_k)$ is the relation strength of u_j to p_k , and $\overline{I(u_j)}$ is the average relation strength of u_j and all POIs.

Algorithm 2 formally describes the proposed two-stage approach for recommendation indoor POIs to users in urban shopping mall. The framework consists of two stages. First, as shown in Lines 2~5, we calculate the initial *User-POI* relation strength and *User-User* relation strength from user's indoor spatiotemporal trajectories. Second, as depicted in Lines 6~8, we construct relation graph and perform a random walk with restart to infer *User-User* relation strength. Third, we calculate the relation strength of the target user to his/her unvisited POIs as shown in Lines 11~13. Finally, we obtain top- k POIs with the highest relation strength as the POI recommendation list.

5. Experiment Evaluation

In this section, we report on the results of a series of experiments conducted to evaluate the performance of the proposed approach to generate indoor spatiotemporal trajectory and recommend top- k POIs to users, followed by discussion. Our experiment environment is a large indoor shopping mall with four floors and over 60 shops, and we regard each shop as an indoor POI, and these shops belong to 4 categories given by the mall owner, as shown in Table 3.

5.1. Generate Indoor Spatiotemporal Trajectory. In this subsection, we describe the experimental settings for generating user's indoor spatiotemporal trajectory using WiFi logs including datasets, comparative approaches, and the evaluation metric. We then report the performance of our proposed method and compare it with three baseline methods.

5.1.1. Experimental Datasets. To evaluate our proposed approach for generating user's indoor spatiotemporal trajectory using WiFi logs, we develop a mobile application based

Require: (1) User set U , POI set P , and user's indoor spatial-temporal trajectory T ; (2) Target user u_i .
Ensure: POI recommendation list of user u_i .
(1) Stage 1: Relation graph construction and inference.
(2) **for** $\forall u \in U$ **do**
(3) Calculate initial *User-POI* relation strength according to Equation (4);
(4) Calculate initial *User-User* relation strength according to Equation (5);
(5) **end for**
(6) Construct relation graph G ;
(7) Obtain transition probability matrix M_p according to Equation (9);
(8) Perform random walk with start over G to infer *User-User* relation;
(9) Stage 2: POI online recommendation
(10) Obtain the unvisited POIs set \bar{P}_i of target user u_i
(11) **for** $\forall p \in \bar{P}_i$ **do**
(12) Calculate *User-POI* relation strength $I(u_i, p)$ according to Equation (10).
(13) **end for**
(14) Rank $\forall p \in \bar{P}_i$ according to $I(u_i, p)$.
(15) **return** Select *top-K* POIs as the recommendation list of u_i .

ALGORITHM 2: The proposed two-stage approach for indoor POI recommendation.

TABLE 4: One example of WiFi RSSI record.

[001]	...	[57]	[58]
-61	...	-75	2016/11/12 14:48:23

on Android system to collect experimental dataset with a sampling rate of 1 Hz. The format of each record is a triple: $\langle \text{mac}, \text{time}, R \rangle$, where mac is the MAC address of scanning device and time is the time of collecting data. R is the scanned RSSI from surrounding WiFi APs, which is represented as a serial of tuples $\text{rssi} = \{\langle \text{mac}_1, r_1 \rangle, \dots, \langle \text{mac}_i, r_i \rangle, \dots\}$, where mac_i represents the MAC address of ap_i and r_i is the scanned RSSI from ap_i .

We invite 25 volunteers carrying mobile phones to collect WiFi logs of 117 predefined trajectories for evaluating the performance of mining indoor trajectory. The predefined information includes the check-in time and check-out time of each shop, which can be regarded as ground-truth data to evaluate the performance of generating indoor spatiotemporal trajectory. After analysis, there are 57 different WiFi APs in all WiFi logs; we extend each RSSI record to a 58-dimensional vector. For WiFi AP without scanning RSSI value, we set -110 dBm as default value in RSSI record. An example of RSSI record is shown in Table 4. In addition, we collect 100 WiFi RSS records in each shop to construct POI feature set according to Definition 4. Note that the phase for constructing POI feature set is not time-consuming and labor intensive, since two hours are enough to collect 6000 RSSI records (the sample rate is 1 Hz) for constructing POI feature set.

5.1.2. Comparative Approaches. We compare our proposed method for generating indoor trajectory with the following three competitor methods:

- (i) *RSSI-NN*. This approach obtains the corresponding POI of each RSSI record using fingerprint-based

positioning method [28], which regards raw WiFi RSSI as POI feature and uses nearest neighbour as matching method.

- (ii) *DIFF-NN*. DIFF-NN [29] uses the difference of RSSI between each pair of WiFi APs as POI feature to solve the RSSI variation problem caused by heterogeneous devices.
- (iii) *Weight-RSS*. Weight-RSS [1] utilizes both the raw WiFi RSSI and their relation to design stable POI fingerprint and uses a weighted k -nearest neighbour as matching method.

After mapping each record of WiFi logs to POI with the above three methods, we construct the corresponding indoor trajectory as Definition 2.

5.1.3. Evaluation Metric. For evaluating the performance of generating indoor spatiotemporal trajectory using WiFi RSSI, we first need to define the distance of two trajectories. Let $C_{ij} = \{p_1, p_2, \dots, p_k\}$ denote the common POI set of trajectory Traj_i and Traj_j , and then the longest common subsequence of Traj_i and Traj_j is defined as

$$\text{LCSS}(C_{ij}) = \begin{cases} 0, & k = 0, \\ \text{LCSS}(\text{Rest}(C_{ij})) + 1, & |t_i(p_k) - t_j(p_k)| < \theta, \end{cases} \quad (11)$$

where $|t_i(p_k) - t_j(p_k)| < \theta$ means that the stay time difference of p_k in Traj_i and Traj_j is less than a threshold and $\text{Rest}(C_{ij}) = \{p_1, p_2, \dots, p_{k-1}\}$.

Following the work of [30], we define the distance of Traj_i and Traj_j as

$$\text{Dist}(\text{Traj}_i, \text{Traj}_j) = 1 - \frac{\text{LCSS}(C_{ij})}{\min\{l_i, l_j\}}, \quad (12)$$

TABLE 5: Recognition accuracy of shop entrance with different time window sizes and user-specific thresholds.

$2h + 1$	θ						
	50	100	150	200	250	300	350
1	0.02	0.05	0.09	0.11	0.17	0.14	0.1
3	0.06	0.11	0.28	0.39	0.47	0.43	0.37
5	0.09	0.26	0.43	0.59	0.71	0.65	0.53
7	0.08	0.21	0.37	0.51	0.62	0.54	0.48
9	0.05	0.13	0.28	0.41	0.47	0.39	0.27

where l_i and l_j are the POIs number of Traj_i and Traj_j , respectively.

5.1.4. Experimental Results

(1) *Recognize Shop Entrance.* In our proposed method, three parameters that directly impact the performance of recognizing shop entrance need to be determined, and they are as follows: time windows size ($2h + 1$) for calculating the RSSI “jump” variation and user-specific thresholds θ and τ for removing false recognition. We empirically set $\tau = 10$ for removing false recognition of shop entrance.

Table 5 shows the recognition accuracy as a function of time window size and user-specific threshold θ for recognizing shop entrance, respectively. From this table, we observe the following: (1) the accuracy drops sharply when the user-specific threshold θ is lower than 150 or greater than 350 and achieves the highest accuracy when $\theta = 250$. (2) Set $\theta = 250$; the accuracy increases with the time window size increasing from 1 to 5 and slightly decreases when the time window size is larger than 5, since the RSSI fluctuation caused by shop entrance and other factors will be smaller with increasing time window size. Finally, the best performance (71%) is achieved when $\theta = 250$ and $2h + 1 = 5$.

Figures 4(a) and 4(b) show the performance of filtering false identification as a function of user-specific threshold and time window size, respectively. From the two figures, we observe the following: (1) Set the time window size $2h + 1 = 5$, and the filtering performance declines sharply when variation coefficient θ is lower than 200 and achieves the best accuracy when $\theta = 250$. (2) Set $\theta = 250$, and the filtering accuracy increases with increasing the number of time window sizes between 3 and 5 and slightly decreases when the time window size is larger than 5. The reason is that the difference of RSS fluctuation between physical boundary points and normal locations will be smaller with increasing the time window size.

Set $\theta = 250$ and $2h + 1 = 5$, and we investigate the recognition accuracy with different sampling rates in Figure 4(c). From this figure, we observe that the recognition performance increases slightly with the sampling rate increasing from 500 ms to 1000 ms and drops significantly when the sampling rate is larger than 1000 ms. The best recognition performance is achieved when setting the sampling rate as 1000 ms. For example, the recognition accuracy is 70.4% after removing false recognition when setting the sampling rate as 1000 ms, while the recognition accuracy is only 24.7% after removing false recognition when setting the sampling rate as

3000 ms. The reason is that when collecting RSS values of a relatively short or long period (e.g., 0.5 s or 3 s), the RSS values of both physical boundary point and normal location will fluctuate wildly when users are moving and thus cannot effectively identify physical boundary points.

(2) *Map WiFi Logs to Indoor Spatiotemporal Trajectory.* Figure 5 shows the performance for generating indoor spatiotemporal trajectory with different methods. We can see that the performance is the same for the three comparative methods, since they are based on a fingerprint-based positioning method using RSSI. Our method outperforms the three methods according to Figure 5. More exactly, the percentage that trajectory distance is less than 0.5 for our method is 81% and 67% for *DIFF-NN*, 60% for *weight-RSS*, and 46% for *RSSI-NN*. The reason our method achieves better performance is that our method can better recognize POI entrance and further more accurately divide WiFi RSSI records into the corresponding POIs, using the RSSI variation characteristics caused by the POI entrance. In contrast, the performance of other three methods will rapidly decline due to the RSSI variation caused by heterogeneous devices or environmental changes.

In addition, we also compare the processing time of generating trajectories for the four methods, as shown in Figure 6. In this figure, we can see that *DIFF-NN* is very time-consuming compared to the other three methods. The reason is that *DIFF-NN* will suffer the curse of dimensionality with numerous WiFi APs, since it constructs POI feature using the difference of each pair of WiFi APs. Our method needs the least time costs. The reason is that our approach firstly splits RSSI sequence into POIs. Therefore, our method does not require dealing with WiFi RSSI records outside POIs.

5.2. *Indoor POI Recommendation.* In this subsection, we describe the experimental settings for indoor POI recommendation, including datasets, comparative approaches, and the evaluation metric. We then report the performance of our proposed method and compare with five baseline methods.

5.2.1. *Experimental Datasets.* We gather an anonymous dataset from 468 registered customers using an opt-in WLAN in an urban shopping mall over 33 days. Firstly, we filter out these WiFi logs that are generated from mall workers and shop employees using the check-in frequency. More exactly, we consider a user as a mall worker or shop employee if his/her check-in frequency is more than 15 during the 33 days.

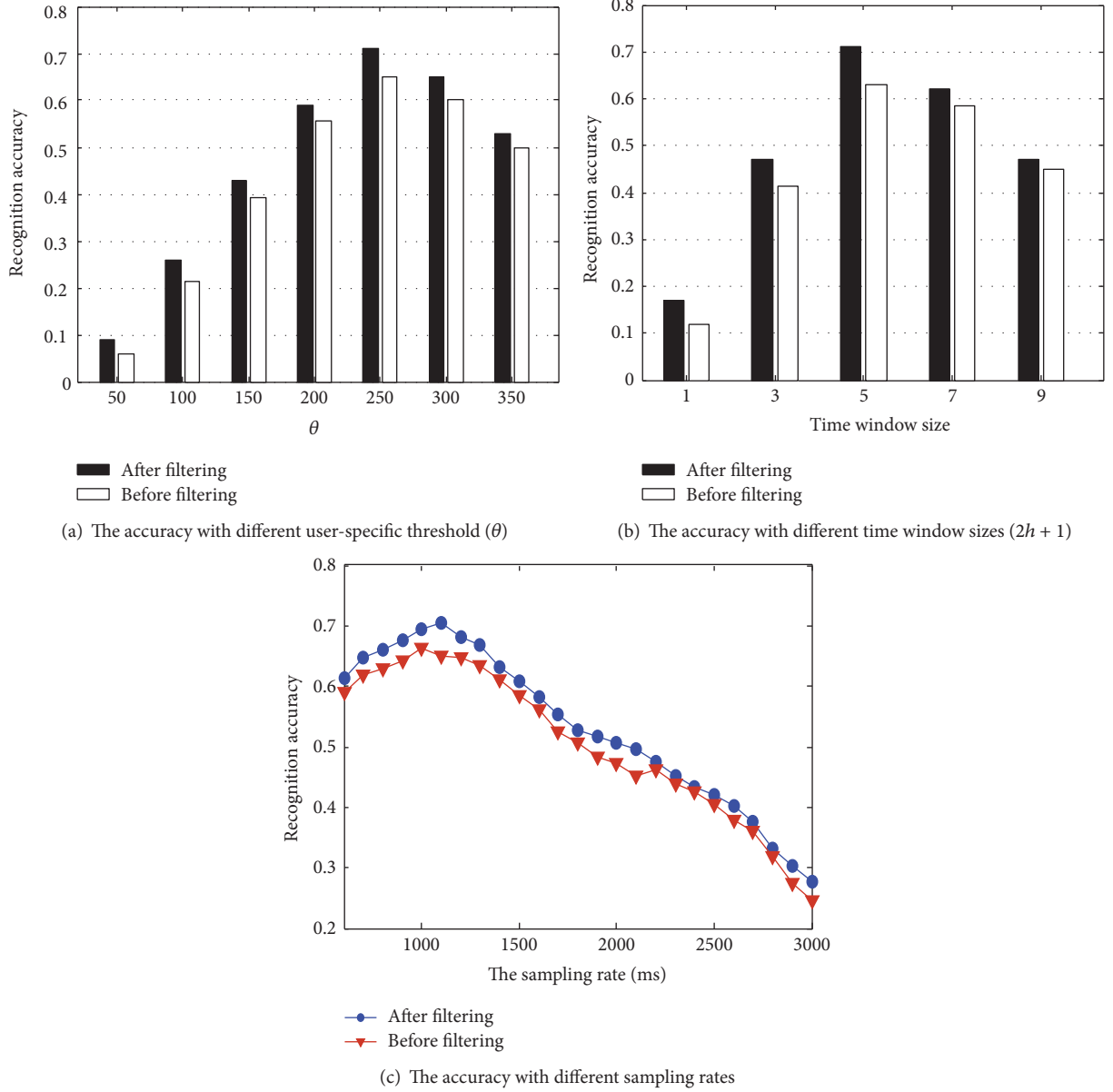


FIGURE 4: The accuracy with different parameters.

TABLE 6: Statistics of data set for indoor POI recommendation.

Number of shops	60
Number of customers	468
Number of WiFi logs	1021
Number of WiFi records	9,556,560
Average RSSI records of each WiFi logs	9360
Average number of visited shops per customer	13
Average number of check-ins per shop	221

After preprocessing, the dataset consists of 1021 WiFi logs and over 9,556,560 WiFi RSSI records. More details of the dataset are shown in Table 6.

5.2.2. Comparative Approaches. We compare our method with the following five well-known recommendation algorithms that have been widely used in POI recommendation systems.

- (i) *Content-Based k -Nearest Neighbours Algorithm (CBNN)* [31]. CBNN utilizes all user's trajectories to create a POI-POI matrix M , and $M(i, j)$ represents the similarity of p_i and p_j based on the visiting correlation of POIs. Formally, $M(i, j)$ is defined as

$$M(i, j) = \frac{\tau_{p_i} \cap \tau_{p_j}}{\tau_{p_i} \cup \tau_{p_j}}, \quad (13)$$

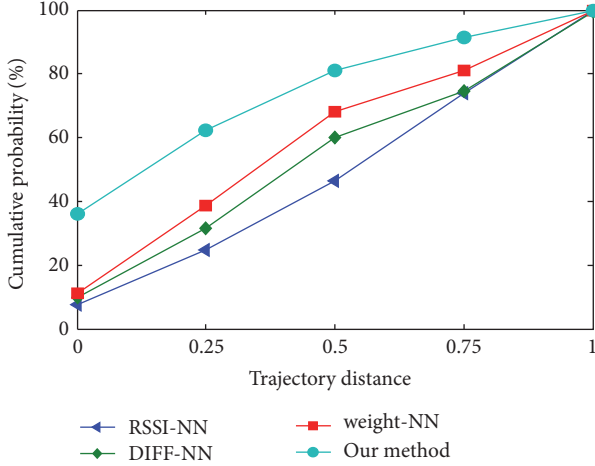


FIGURE 5: The performance for generating indoor trajectories.

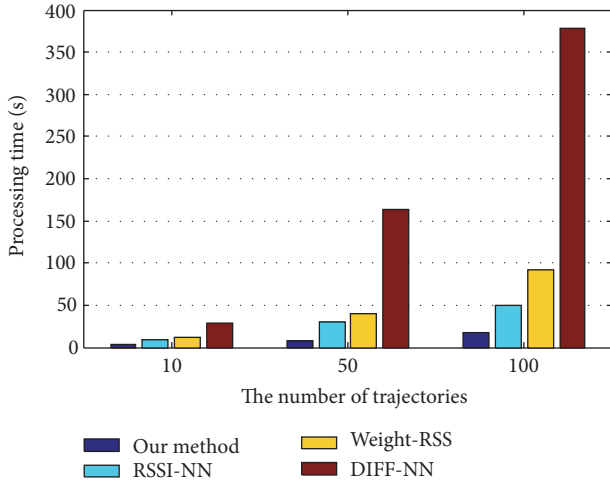


FIGURE 6: The time cost for generating indoor trajectories.

where τ_{p_i} and τ_{p_j} denote the set of trajectories include p_i and p_j , respectively.

For a unvisited POI p_x , CBNN first retrieves K nearest neighbour POIs $P = \{p_1, \dots, p_k\}$ which have been visited by the target user u using *POI-POI* matrix. Then, calculate the recommendation score of p_x using *User-POI* relation, as shown in

$$\text{Score}(p_x) = \sum_{i=1}^K I(u, p_i). \quad (14)$$

- (ii) *Item-Based Collaborative Filtering Algorithm (Item-Based CF)* [32]. This method formulates *User-POI* matrix and *POI-POI* matrix according to (4) and (13) and then applies item-based collaborative filtering to calculate the recommendation score of unvisited POIs for the target user.
- (iii) *User-Based Collaborative Filtering Algorithm (User-Based CF)* [33]. Similar to *Item-Based CF*, this method first obtains *User-POI* matrix and *User-User* matrix

according to (4) and (5) and then applies user-based collaborative filtering to calculate the recommendation score of unvisited POIs.

- (iv) *Collaborative Filtering Based Location Cooccurrence (LCCF)* [34]. LCCF calculates the recommendation score of unvisited POI by combining other user's visiting history of the POI. Let $c\langle u, p \rangle = 1$ if u has visited POI p , and $c\langle u, p \rangle = 0$ otherwise; $C_u = (c\langle u, p_1 \rangle, \dots, c\langle u, p_N \rangle)$ is the check-in vector of customer u . Then, the recommendation score between u and an unvisited POI \hat{p} is calculated by

$$\text{score}(u, \hat{p}) = \frac{\sum_{v \in U} \text{sim}(u, v) * c\langle v, \hat{p} \rangle}{\sum_{v \in U} \text{sim}(u, v)}, \quad (15)$$

where $\text{sim}(u, v)$ is the similarity between user u and user v and is calculated using the cosine similarity between C_u and C_v .

- (v) *Rule-Based Recommendation Algorithm (RBCA)* [15]. RBCA estimates user's POI preference by linearly fusing three factors: time spent in a POI, check-in frequency of a POI, and matching between promotional activities in the POI and user preference towards promotional activities. Recommendation rules are extracted according to two assumptions: one is that the higher the user's preference for a POI is, the more likely he/she is to visit the store; another is that a user will visit the POI in which promotions are in line with his preference.
- (vi) *Graph Recommendation Based Random Walk with Restart (RWR)* [24]. This method first constructs a graph, where a node in the graph denotes a POI and the weights between nodes are assigned using the POI's relation. Then, RWR considers the POI recommendation as an entity ranking problem on the graph and utilizes personalized PageRank algorithm [35] to generate top- k recommended POIs.

5.2.3. Evaluation Metric. For each user u in T_{test} , using $P_{\text{te}}(u)$ represents POIs where u has been visited in the test set. Let $\text{hit}(u)$ represent the recommendation POIs that appear in $P_{\text{te}}(u)$, which means that u is interesting in POIs of $\text{hit}(u)$. Then, we define the recommendation hit rate as

$$\text{Hit}(k) = \frac{\text{hit}(u)}{k}, \quad (16)$$

where k is the length of recommendation list.

For evaluating the effectiveness of the proposed POI recommendation algorithm, we randomly select 30% of indoor trajectories as the test set and use the rest as training set, denoted by T_{test} and T_{training} . We use T_{training} to construct the relation graph for inferring *User-User* relation and *User-POI* relation. Specifically, T_{test} includes 133 users as the target users. For evaluation of the performance of the proposed recommendation algorithm, we perform a *top-k* recommendation task on the 133 target users.

In addition to hit rate, we also measure the recommendation diversity of the proposed method, which indicates the

surprise of recommended results. Following the work in [36], we exploit the category information of POIs to evaluate the recommendation diversity. For each user, the diversity $\text{div}(u)$ is measured as follows:

$$\text{div}(u) = \frac{|C(\text{Tr}_u) \cap C(\text{Te}_u)|}{|C(\text{Tr}_u)|}, \quad (17)$$

where $C(\text{Tr}_u)$ denotes the categories that u likes in the training data while $C(\text{Tr}_u) \cap C(\text{Te}_u)$ are the categories that appear in the recommendation list of user u . Then, we measure the total diversity of a recommendation algorithm by averaging the diversity of all users:

$$\text{div} = \frac{\sum_{u=1}^{|\text{Te}|} \text{div}(u)}{|\text{Te}|}, \quad (18)$$

where $|\text{Te}|$ denotes the number of users in test data.

5.2.4. Experimental Results. In this subsection, we first report the impact of model parameters for the proposed algorithm and then present the results of our experiments for all users and cold-start users, respectively.

(1) *Impact of Model Parameters.* Tuning model parameters, such as the parameter λ for inferring the *User-User* relation using random walk with restart, is critical to the performance of the proposed algorithm. We study the impact of model parameter (λ) on the dataset. Set the recommendation list k as 5, and we tested the performance of the proposed recommendation model by varying parameter λ and present the results in Figure 7. From the figure, we observe the following: (1) the $\text{Hit}(5)$ slightly increases with the increasing of λ from 0.1 to 0.7 and then decreases when λ is greater than 0.7; (2) the best performance $\text{Hit}(5) = 0.281$ is achieved when $\lambda = 0.7$, which means a user will check in 28.1% POIs of the recommendation list if we recommend 5 POIs to the user. The reason is that the convergence of random walk with restart is determined by the parameter λ ; that is, a greater λ leads to faster convergence and thus can make better recommendation. But an overlarge λ will cause a high probability to return to the target user when selecting recommended neighbours, thus reducing the number of high quality recommended neighbours and further decreasing the recommendation performance.

(2) *Effectiveness of Recommendation for All Users.* Figure 8 shows the recommendation hit rate of the six recommendation algorithms. Note that we only perform experiments where the recommendation list $k \in [1, 8]$, due to a greater value of k , is usually ignored for *top-k* recommendation task since there is 60 POIs in total. It is apparent that all the six algorithms have significant performance disparity in terms of *top-k* hit rate. As shown in Figure 8, the $\text{Hit}(k)$ of our method is about 71% when $K = 1$ and 35% when $K = 4$, which means that a user has 71% probability to check in the recommended POI if we only recommend one POI. Similarly, if we recommend 4 POIs to the user each time, only 35% recommendation POIs can attract the

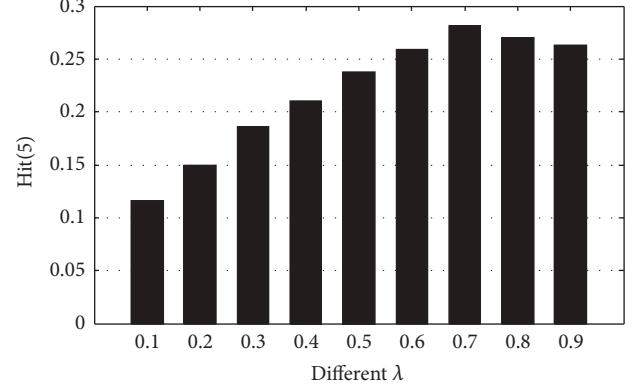


FIGURE 7: The performance of $\text{Hit}(5)$ with different λ .

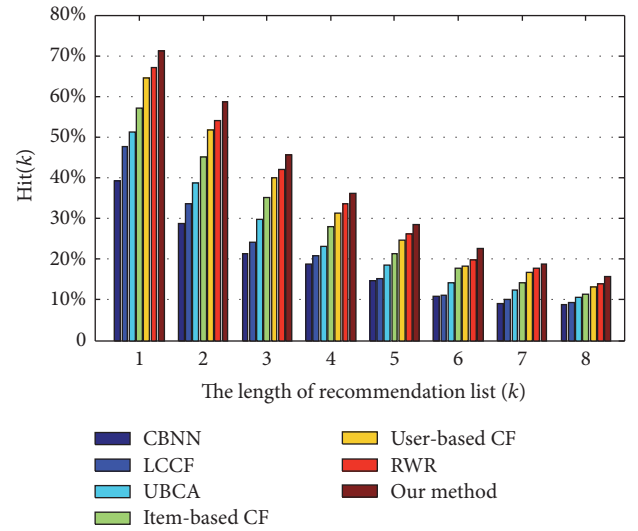


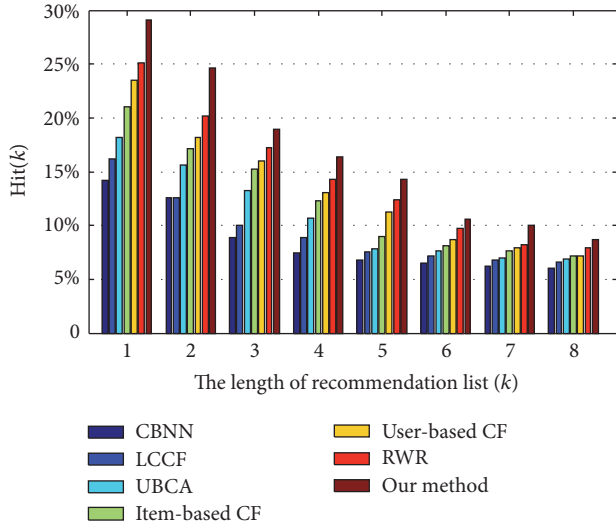
FIGURE 8: The performance of *top-k* POI recommendation for all users.

user's attention. We can see that CBNN and LCCF perform rather worse in the experiment, showing that only using the location cooccurrence is not enough to learn user's preference. *Item-based CF* and *User-based CF* can achieve better performance than CBNN, showing the advantages of using collaborative filtering to model user's preference and POI's characteristics. Obviously, our proposed method outperforms the five recommendation algorithms significantly, showing the advantages of our relation graph to capture multirelation between users and POIs, which can better make recommendation for "cold-start" users that only have little historical visiting information.

Table 7 reports the results of diversity for the six recommendation algorithms. From this table, we can observe the following: (1) Compared to recommendation algorithms using POI correlation (such as CBNN, LCCF, and user-based CF), item-based CF achieve much better diversity. For example, item-based CF improves about 19% diversity compared to CBNN, 24% diversity compared to LCCF, and 13% diversity compared to user-based CF when the recommendation list is 5. The reason is that user-based CF usually recommends POIs with high popularity to users, so it

TABLE 7: Recommendation diversity for all users.

Methods	Length (k)		
	3	5	7
CBNN	0.21	0.39	0.52
UBCA	0.27	0.51	0.63
LCCF	0.18	0.34	0.42
User-based	0.24	0.45	0.58
Item-based	0.31	0.58	0.71
RWR	0.27	0.52	0.64
Our method	0.29	0.56	0.67

FIGURE 9: The performance of $top-k$ POI recommendation for cold-start users.

is insufficient to discover POIs in the long tail and thus result in low recommendation diversity. (2) Our method achieves slightly worse performance when compared to item-based CF (item-based CF improves about average 2.3% compared to our method), which is also much better than the other four algorithms. This improvement is achieved by utilizing random walk with restart to derive pairwise score between each pair of users, which can solve the problem of data sparsity to a certain extent.

(3) *Effectiveness of Recommendation for Cold-Start Users.* Figure 9 reports the recommendation performance of the six recommendation algorithms; similarly, we only perform experiments where the recommendation list $k \in [1, 8]$. From this figure, we observe the following: (1) the performance of all the six algorithms for cold-start users degrades significantly compared to all users, showing data sparsity caused by cold-start users bring serious challenge for indoor POI recommendation. (2) Our method outperforms the other five methods by improving average 6% hit rate, showing the advantage of using random walk with restart to learn user's preference.

Table 8 reports the results of diversity for cold-start users with the six recommendation algorithms. The results show that, compared to traditional user-based models (user-based

TABLE 8: Recommendation diversity for cold-start users.

Methods	Length (k)		
	3	5	7
CBNN	0.11	0.18	0.3
UBCA	0.19	0.31	0.48
LCCF	0.09	0.16	0.21
User-based	0.12	0.22	0.37
Item-based	0.25	0.37	0.55
RWR	0.18	0.29	0.44
Our method	0.21	0.32	0.49

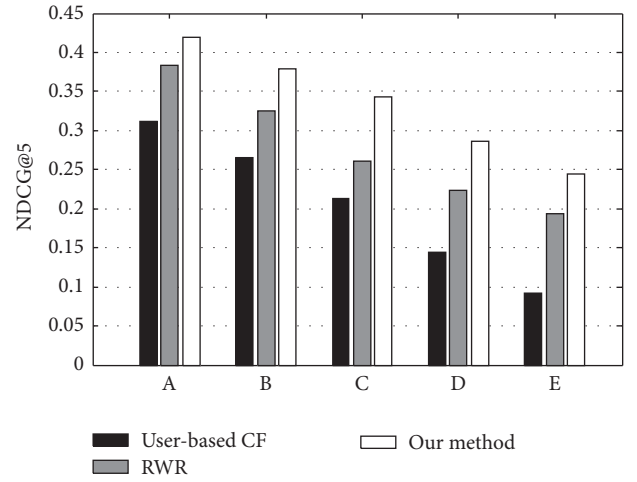


FIGURE 10: The NDCG@5 for different group of users.

CF, LCCF, and CBNN), item-based CF achieves better diversity. For example, item-based CF improves about 19% diversity compared to CBNN, 21% diversity compared to LCCF, and 15% diversity compared to user-based CF when the recommendation list is 5. We can see that the diversity improvement using item-based approaches compared to user-based CF is much more for cold-start users than for all users. We further observe that the proposed algorithm outperforms user-based CF, but slightly worse diversity (about 5%) compared with item-based CF.

In summary, the proposed method substantially improves the coverage of existing user-based POI recommendation approaches while maintaining a slightly better hit rate. This improvement is achieved by exploiting random walk with restart to construct relation graph, which can better model the preference of users with few POI check-ins.

5.2.5. A Case Study. We perform a case study on the usefulness of indoor POI recommendation. A group including 25 volunteers participated in the case study.

Dataset. As mentioned above, we invited 25 volunteers carrying mobile phones to collect 117 WiFi logs for evaluating the performance of generating indoor trajectory. We utilize this dataset for evaluating the performance of this case study,

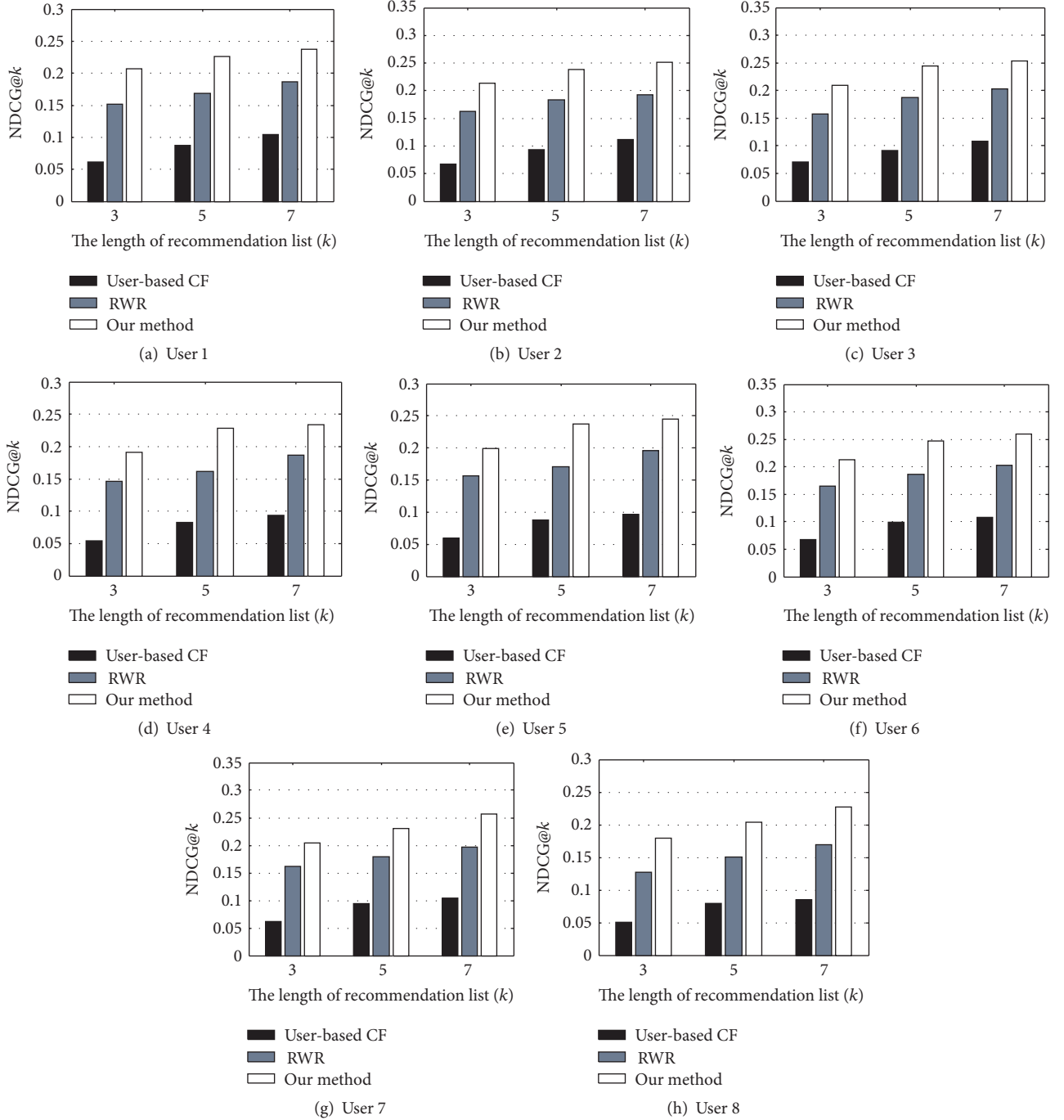


FIGURE 11: The NDCG@k for cold-start users.

as shown in Table 9. The volunteers are divided into 5 groups (i.e., A, B, C, D, and E) according to average visited shops per customer; we regard the volunteers of E group that have only an average of four visiting shops as “cold-start” users.

Evaluation Metric. We utilize NDCG@k to evaluate the performance of indoor POI recommendation. Let corr_i denote a relevance value, and NGCG@k is calculated as

$$\text{NGCG@k} = \frac{\text{DCG}(k)}{\text{IDCG}(k)},$$

$$\text{DCG}(k) = \text{corr}_1 + \sum_{i=2}^k \frac{\text{corr}_i}{\log_2 i}, \quad (19)$$

where IDCG(k) is the DCG(k) value of ideal ranking list.

For instance, if the recommendation list of a user is $[p_1, p_2, p_3, p_4, p_5]$ for five stores while the ideal recommendation list is $[p_2, p_3, p_5, p_4, p_1]$ given by the user, then we can calculate the NDCG@5 = 0.659 according to (19).

TABLE 9: Dataset for case study.

Number of users	Number of WiFi logs	Number of average visited POIs	Category
4	9	38	A
5	8	29	B
3	6	22	C
5	3	10	D
8	1	4	E

Results. Figure 10 shows the NDCG@5 for the five group users with three recommendation models: user-based CF, RWR, and our method. The NDCG@5 is calculated as follows: for a specific user, the recommendation model firstly derives his/her shopping preference from check-in records and then generates top 5 recommendation list using the three recommendation models, respectively. For the recommendation results, each user has an ideal rank list answer in his or her mind. Based on the recommendation list from recommendation model and user's ideal rank list, the NDCG@5 can be calculated as (19). From Figure 10, we can observe the following: (1) for all the five groups of users, the NDCG@5 of our proposed model is better than the compared models (user-based CF and RWR). For example, the performance improvement for group C users is about 13% and 8% compared with user-based CF and RWR; (2) the performance improvement of recommendation model using random walk is significant for "cold-start" users. For example, the performance improvement of our method for group E users is about 15% compared with user-based CF. The results suggest that our recommendation algorithm can learn user's preference even with few POI check-ins by using a graph-based model to capture the latent relation between users and POIs.

To investigate the recommendation effectiveness for "cold-start" users, we further calculate the NDCG@ k (i.e., $k \in [3, 5, 7]$) for each user of E group, as shown in Figure 11. From this figure, we can observe that, for all cold-start users, the NDCG@ k of our proposed model is much better than the compared models (user-based CF and RWR), showing again that the advantage of our method derives user's preference by using a graph-based model to capture the latent relation between users and POIs.

6. Conclusion

This paper proposed a location-aware Point-of-Interest recommendation system for urban shopping mall that recommends a set of POIs to a user by mining customer's preference towards POIs from his/her historical indoor trajectories. For generating indoor spatiotemporal trajectories, we propose a novel method that utilizes the propagation characteristics of WiFi RSSI in indoor space. The proposed recommendation system cannot only facilitate user's shopping experience but also help the shop owner better understand user's shopping preference and intent. By constructing a relation graph model and exploiting random walk with restart, our recommendation algorithm can learn

user's preference even with few POI check-ins. We evaluated the proposed recommendation model on a real dataset collected by 468 users over 33 days. The experimental results show that our approach significantly outperforms existing recommendation algorithms in recommendation hit rate and diversity.

As future work, we plan to consider the POI semantic information (such as POI service or online reviews) for further improving the recommendation performance.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work has been supported by Hangzhou Key Laboratory for IoT Technology & Application.

References

- [1] Z. Zheng, Y. Chen, T. He, F. Li, and D. Chen, "Weight-RSS: a calibration-free and robust method for WLAN-based indoor positioning," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 573582, 7 pages, 2015.
- [2] A. W. S. Au, C. Feng, S. Valaee et al., "Indoor tracking and navigation using received signal strength and compressive sensing on a mobile device," *IEEE Transactions on Mobile Computing*, vol. 12, no. 10, pp. 2050–2062, 2013.
- [3] L. Geng, M. F. Bugallo, A. Athalye, and P. M. Djuric, "Indoor tracking with RFID systems," *IEEE Journal on Selected Topics in Signal Processing*, vol. 8, no. 1, pp. 96–105, 2014.
- [4] R. Nandakumar, S. Rallapalli, and K. Chintalapudi, "Physical analytics: A new frontier for (indoor) location research," Tech. Rep., Microsoft Research, Bangalore, India, 2013.
- [5] J.-B. Griesner, T. Abdesslem, and H. Naacke, "POI recommendation: towards fused matrix factorization with geographical and temporal influences," in *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys '15)*, pp. 301–304, September 2015.
- [6] B. Liu, H. Xiong, S. Papadimitriou, Y. Fu, and Z. Yao, "A general geographical probabilistic factor model for point of interest recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 5, pp. 1167–1179, 2015.
- [7] T. Prentow, A. Thom, H. Blunck, and J. Vahrenhold, "Making Sense of Trajectory Data in Indoor Spaces," in *Proceedings of the 16th IEEE International Conference on Mobile Data Management (MDM '15)*, pp. 116–121, usa, June 2015.
- [8] M. Werner, L. Schauer, and A. Scharf, "Reliable trajectory classification using Wi-Fi signal strength in indoor scenarios," in *Proceedings of the 2014 IEEE/ION Position, Location and Navigation Symposium (PLANS '14)*, pp. 663–670, May 2014.
- [9] L. Radaelli, D. Sabonis, H. Lu, and C. S. Jensen, "Identifying typical movements among indoor objects - Concepts and empirical study," in *Proceedings of the 14th International Conference on Mobile Data Management (MDM '13)*, pp. 197–206, June 2013.
- [10] M. Dakkak, A. Nakib, B. Daachi, P. Siarry, and J. Lemoine, "Mobile indoor location based on fractional differentiation," in *Proceedings of the 2012 IEEE Wireless Communications and Networking Conference (WCNC '12)*, pp. 2003–2008, April 2012.

- [11] S. Lee, B. Cho, B. Koo, S. Ryu, J. Choi, and S. Kim, "Kalman filter-based indoor position tracking with self-calibration for RSS variation mitigation," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 674635, 10 pages, 2015.
- [12] Y. Jin, W.-S. Soh, M. Motani, and W.-C. Wong, "A robust indoor pedestrian tracking system with sparse infrastructure support," *IEEE Transactions on Mobile Computing*, vol. 12, no. 7, pp. 1392–1403, 2013.
- [13] Z. Lin, *Indoor Location-based Recommender System [Ph.D. thesis]*, University of Toronto, 2013.
- [14] P. Jin, J. Du, C. Huang, S. Wan, and L. Yue, "Detecting hotspots from trajectory data in indoor space," in *Database Systems for Advanced Applications*, pp. 209–225, Springer, 2015.
- [15] B. Fang, S. Liao, K. Xu, H. Cheng, C. Zhu, and H. Chen, "A novel mobile recommender system for indoor shopping," *Expert Systems with Applications*, vol. 39, no. 15, pp. 11992–12000, 2012.
- [16] J. Pfeiffer, T. Pfeiffer, and M. Meißner, "towards attentive in-store recommender systems," in *Reshaping Society through Analytics, Collaboration, and Decision Support*, pp. 161–173, Springer, 2015.
- [17] M. Ruta, F. Scioscia, S. Ieva, D. D. Filippis, E. D. Sciascio et al., "Indoor/outdoor mobile navigation via knowledge-based poi discovery in augmented reality," *Archives of Gynecology and Obstetrics*, vol. 291, no. 1, pp. 59–66, 2015.
- [18] M. Das, G. De Francisci Morales, A. Gionis, and I. Weber, "Learning to question," in *Proceedings of the the 19th ACM SIGKDD international conference*, p. 203, Chicago, Illinois, USA, August 2013.
- [19] M. Paschou, E. Sakkopoulos, A. Tsakalidis, G. Tzimas, and E. Viennas, "Intelligent mobile recommendations for exhibitions using indoor location services," *Smart Innovation, Systems and Technologies*, vol. 25, pp. 19–38, 2013.
- [20] H. Shin, Y. Chon, Y. Kim, and H. Cha, "A participatory service platform for indoor location-based services," *IEEE Pervasive Computing*, vol. 14, no. 1, pp. 62–69, 2015.
- [21] Z. Zheng, Y. Chen, T. He, L. Sun, and D. Chen, "Feature learning for fingerprint-based positioning in indoor environment," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 452590, 2015.
- [22] A. B. M. Musa and J. Eriksson, "Tracking unmodified smartphones using wi-fi monitors," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems (SenSys '12)*, pp. 281–294, ACM, Ontario, Canada, November 2012.
- [23] H. Nurminen, J. Talvitie, S. Ali-Loytty et al., "Statistical path loss parameter estimation and positioning using RSS measurements in indoor wireless networks," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN '12)*, pp. 1–9, Sydney, Australia, November 2012.
- [24] S. Lee, S.-I. Song, M. Kahng, D. Lee, and S.-G. Lee, "Random walk based entity ranking on graph for multidimensional recommendation," in *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys '11)*, pp. 93–100, October 2011.
- [25] H. Zhou, Z. Deng, Y. Xia, and M. Fu, "A new sampling method in particle filter based on Pearson correlation coefficient," *Neurocomputing*, vol. 216, pp. 208–215, 2016.
- [26] M. Jamali and M. Ester, "TrustWalker: a random walk model for combining trust-based and item-based recommendation," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*, pp. 397–405, July 2009.
- [27] H. Yildirim and M. S. Krishnamoorthy, "A random walk method for alleviating the sparsity problem in collaborative filtering," in *Proceedings of the ACM Conference on Recommender Systems*, pp. 131–138, New York, NY, USA, October 2008.
- [28] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM '00)*, vol. 2, pp. 775–784, Tel Aviv, Israel, March 2000.
- [29] F. Dong, Y. Chen, J. Liu, Q. Ning, and S. Piao, "A calibrationfree localization solution for handling signal strength variance," in *Mobile Entity Localization and Tracking in GPS-Less Environments: Second International Workshop, MELT 2009, Orlando, FL, USA, September 30, 2009. Proceedings*, pp. 79–90, Springer, Berlin, Germany, 2009.
- [30] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh, "Indexing multi-dimensional time-series with support for multiple distance measures," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*, pp. 216–225, August 2003.
- [31] M. Clements, P. Serdyukov, A. P. de Vries, and M. J. Reinders, *Personalised Travel Recommendation Based on Location Co-Occurrence*, 2011.
- [32] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 143–177, 2004.
- [33] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, "Collaborative Filtering Meets Mobile Recommendation: A User-Centered Approach," *AAAI*, vol. 10, pp. 236–241, 2012.
- [34] M. Ye, P. Yin, and W.-C. Lee, "Location recommendation for location-based social networks," in *Proceedings of the 18th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL GIS '10)*, pp. 458–461, November 2010.
- [35] L. Page, S. Brin, M. Rajeev, and T. Winograd, "The pagerank citation ranking: bringing order to the web. technical report, Stanford InfoLab," Tech. Rep., Stanford InfoLab, 1999.
- [36] L. Wu, Q. Liu, E. Chen, N. J. Yuan, G. Guo, and X. Xie, "Relevance meets coverage: a unified framework to generate diversified recommendations," *ACM Transactions on Intelligent Systems and Technology*, vol. 7, no. 3, article no. 39, 2016.

