Using Location-Labeling for Privacy Protection in Location-Based Services

Dan Liao[1,2], Hui Li[1], Vishal Anand[3], Victor Chang[4], Gang Sun[1] and Hongfang Yu[1]
[1]*Key Lab of Optical Fiber Sensing and Communications (Ministry of Education), University of Electronic Science and Technology of China, Chengdu, China*
[2]*Institute of Electronic and Information Engineering in Dongguan, UESTC, Dongguan, China*
[3]*Department of Computer Science, The College at Brockport, State University of New York, Brockport, U.S.A.*
[4]*Leeds Beckett University, Leeds, U.K.*

Keywords: Location-Based Service (LBS), K-anonymity, Location Privacy, Location-Label, Sensitive Location.

Abstract: The developments in positioning and mobile communication technology have made applications that use location-based services (LBS) increasingly popular. For privacy reasons and due to lack of trust in the LBS provider, *k*-anonymity and *l*-diversity techniques have been widely used to preserve user privacy in distributed LBS architectures. However, in reality, there exist scenarios where the user locations are identical or similar/near each other. In such a scenario the *k* locations selected by *k*-anonymity technique are the same and location privacy can be easily compromised or leaked. To address the issue of privacy protection, in this paper, we propose the concept of location-labels to distinguish mobile user locations to sensitive locations and ordinary locations. We design a location-label based (LLB) algorithm for protecting location privacy while minimizing the query response time of LBS. We also evaluate the performance and validate the correctness of the proposed algorithm through extensive simulations.

1 INTRODUCTION

Due to the development in positioning technology and mobile communication technology, applications of location -based services (LBS) (Li and Yiu, 2015) have rapidly risen as more and more people make use of these services. After receiving a LBS request from a user, the LBS provider (LP) responds to the request, according to the user location information and the requested content. Although users enjoy the conveniences of the services provided by the LBS provider, there is a potential security risk of losing their privacy. For example loss of privacy of location or trajectory which may be leaked to other parties).

Existing *k*-anonymity (Yang et al., 2013) and the pseudo-ID technique (Bettini et al., 2007) are effective techniques to protect user location privacy in LBS. The authors in (Liu et al., 2014, Zhu et al., 2013, Shao et al., 2014) provided solutions to solve the problem of privacy preservation by using *k*-anonymity. In this, when a user sends a query to the LP, the user merges other *k*-1 user queries along with their original request and submits the mixed query to the LP. However, the LP can easily get user requested contents when the requested contents of the *k* users are similar to each other. Using data analysis and mining the LP can infer more information about users, such as common interests and hobbies. To combat this deficiency, researchers introduced the *l*-diversity concept (Niu et al., 2015, Lu et al., 2014) to protect the requested contents (or preference privacy (Lu et al., 2014)). In this method, all LBS queries can be classified into different categories (e.g., medical, traffic, entertainment, etc.) according to their requested contents. The basic idea of the *l*-diversity technique is to make LBS queries of users different. Therefore, this property can ensure that there exists at least *l* services in the *k* LBS queries, where $k \geq l$. The Privacy-preserving framework for Local-Area Mobile social networks (PLAM) (Lu et al., 2014) adopts *k*-anonymity and *l*-diversity to protect location and preference privacy of users. As shown in Figure 1(a), there exist 6 (i.e., $k = 6$) users who are distributed in different locations requesting 3 (i.e., $l = 3$) services. Then the LP cannot link a specific service/location to a specific user. Thus, the PLAM method can protect the location and preference privacy when the users' locations are different. However, consider the scenario in Figure

Liao, D., Li, H., Anand, V., Chang, V., Sun, G. and Yu, H.
Using Location-Labeling for Privacy Protection in Location-Based Services.
DOI: 10.5220/0005843402990306
In *Proceedings of the International Conference on Internet of Things and Big Data (IoTBD 2016)*, pages 299-306
ISBN: 978-989-758-183-0

1(b) where the *k* users have the same location and send requests together to the LP. Although the PLAM can protect the preference privacy of users with *l*-diversity technique, the LP can know that the *k* users are in the same location and the location privacy is leaked. Thus, PLAM cannot protect location privacy when the users have the same location, especially in some locations such as supermarket, school and hospital where the probability of selecting the same location with *k*-anonymity technology is very high.

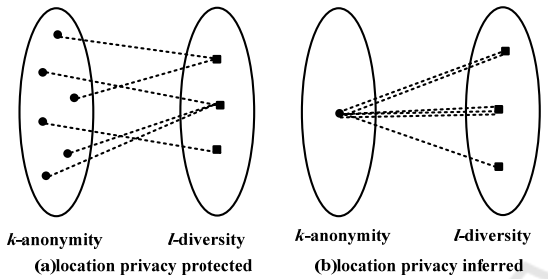

Figure 1: Same vs. different k locations.

Most existing solutions (e.g. *k*-anonymity (Yang et al., 2013), *l*-diversity (Niu et al., 2015), (Lu et al., 2014)) address the issue of privacy preservation in LBS assuming that the user locations are different. In this paper, we study the problem of privacy protection in LBS for users within the same location. We devise an algorithm based on *location-label* to protect the location privacy, preference privacy and trajectory privacy of users. The main contributions of this paper are as follows:

- We propose the concept of location-labels that classifies all locations into *sensitive* and *ordinary locations*. Due to the large population density in a sensitive location, the locations selected by *k*-anonymity are much more identical than in ordinary locations;
- We propose a *location-label* based algorithm (called LLB) for privacy preservation for the scenario where the locations of *k* users are nearby, similar or identical;
- We propose three protocols including the *request aggregation protocol*, the *pseudo-ID exchange protocol* and the *improved PLAM protocol* in our proposed algorithm, which help in reducing the response time of the LBS system.

2 RELATED WORK

There are several studies on location-privacy, which focus on the possible loss of location privacy during the localization process. These localization techniques in a LBS system are able to derive user locations using anchor points (Uchiyama et al., 2013). Since the localization algorithm takes anchor points as input and outputs users' location, location of anchors and user's location may be leaked to others. Thus in order to efficiently protect user location information during localization process, the authors in (Li et al., 2014) proposed the PriWFL algorithm, and the authors in (Shu et al., 2014) studied the problem of multi-lateral privacy preserving localization.

There are other studies that focus on protecting user location privacy in LBS applications. In these studies, the user locations are calculated by local facilities, and two kinds of requests are considered: single requests and continuous requests. For single requests, the location privacy and preference privacy need to be protected. Several strategies such as *k*-anonymity (Niu et al., 2015), Mix Zones (Beresford and Stajano, 2014), *l*-diversity (Niu et al., 2015), *m*-unobservability (Chen et al., 2013) etc. have been proposed to prevent the LBS provider from inferring the users' location or preference privacies. The authors in (Niu et al., 2014) proposed the DLS algorithm with the *k*-anonymity and *l*-diversity properties for protecting location privacy and preference privacy of users.

When a user sends continuous requests (i.e. sending requests continuously for a period of time) to the LP, the trajectory information of the user needs to be protected. Feng et al. (Feng et al., 2012) proposed an algorithm called VAvatar to protect users' locations and trajectories. Mohammed et al. (Mohammed et al., 2009) proposed a Track False Data method for the problem of protecting the privacy of continuous requests, in which the users send their fake location and track information to the LBS provider, rather than their real trajectory data. Wang et al. (Wang et al., 2012) proposed a distributed query privacy preserving solution to protect the trajectory privacy of user.

The existing studies mentioned above address the problem of privacy preservation under the assumption that users are distributed in different locations. However, in reality there exists a situation that multiple users may have the same location. In this work, we investigate the problem of location privacy preservation for users in the same location.

3 PRELIMINARIES AND PROBLEM SRATEMENT

In this section, we first introduce the basic concepts and definitions. Then, we give the problem description and LBS system model for the problem.

3.1 Preliminaries

1) *Sensitive Locations and Ordinary Locations*: All locations can be classified into two categories: *sensitive locations* and *ordinary locations*. Sensitive locations typically have a dense population, which includes hospitals, supermarkets, schools, etc. While the ordinary locations are sparsely populated, such as the locations on general roads.

Generally, there are some commonalities between sensitive locations, for example, *i*) the sensitive location is usually in a region with developed traffic; *ii*) they are located in an area with dense populations; *iii*) the users who gather in a sensitive location have common characteristic(s). For example, if the users are in a hospital (a sensitive location), the possibility of that they are patients or doctors is very high. Since the users in a sensitive location have common characteristics (e.g., interests, needs), the request contents of these users may be similar while using LBS. Thus we need to protect the identity of users in these locations, and do not need to pay much attention on protecting their requested contents and the preference privacy. For example, users who in hospital do not care whether the attacker knows the requested information (e.g., health information), but they do know that their identities cannot be inferred by the malicious attacker. For an ordinary location, since it is just a location on general roads, users are more concerned about the location privacy and the preference privacy that have not been leaked out. Figure 2 shows an example of partitioning of locations.

2) *User Location:* A user location is denoted by d (x, y, *label*), where x and y represent the latitude and longitude of a location, and *label* represents the category of the location. For example, if a user is in a sensitive location (e.g., hospital), the content of *label* describes the information about the hospital. If the user is in an ordinary location, the content of *label* is null.

3) *Service Category:* We can classify user requests into different service categories, according to the services provided by the LBS system. For example, some users query for entertainment information, while others may query for dining or dating. We store the various service categories in set

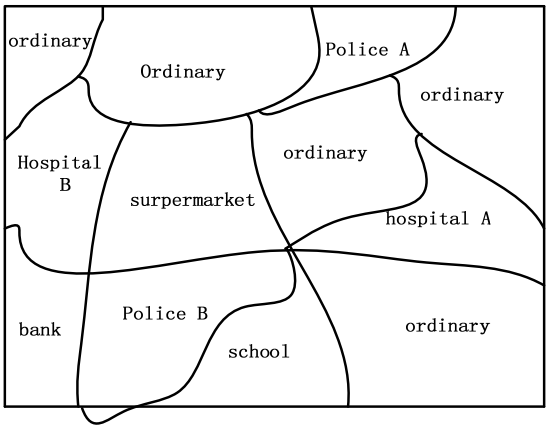

Figure 2: An example of the partition for locations.

$Serve = \{sc_1, sc_2, \ldots, sc_i, \ldots, sc_m\}$, where sc_i denotes a service provided by the LBS system. For sc_i, we use s_i to denote whether a user is using the service. For example, when s_i=1, it means that a user is using the service sc_i.

4) *Single Request Packet:* The single request packet is denoted by $Rq_i = \{Pid_i, d_i, serve, R_i, t\}$, where Pid_i represents the user's identity, d_i denotes the user's location information, *serve* represents the service category, and R_i represents specific content of the corresponding request. The time t is set to indicate the tolerable response time for the LBS request of a user.

5) *Aggregated Packet:* Before a user u_i submitting a request to the LP, he/she aggregates requests from other users. User u_i first broadcasts the aggregating message to other users. If there are other k-1 users agreeing to join with the user u_i, they send their *single request packet* to user u_i, and user u_i will become the *representative user* for them. The representative user merges the k users' request packets and forms the aggregated request packet, denoted by $Ag=\{P_{list}, \{d_1, serve_1, R_1\}, \{d_2, serve_2, R_2\}, \ldots, \{d_k, serve_k, R_k\}\}$. Where the P_{list} is a list of identities of the k users and $P_{list}=\{Pid_1, Pid_2, \ldots, Pid_k\}$. The k triples denote the requests of k users. Each triple contains the location information d_i, the requested content R_i, and the service category $serve_i$ of a user. Due to the randomness and uncertainty of users, the k locations and the requested services corresponding to the k users may be the same. Thus, we have $1 \le |d| \le k$ and $l \le |s| \le k$, where d is the set of locations of k users, $|d|$ is the number of distinct locations, s is the set of the requested service categories of k users and $|s|$ is the number of the different requested service categories.

6) *Bilinear Pairings*: Similar to (Lu et al., 2014, Liao and Hsiao, 2011), let G and G_T be the cyclic

additive and multiplicative groups, both generated based on the same prime order q. Suppose that p is the generator of group G, Z_q is the residual class ring with modulo q and Z_q^* is an invertible element set relative to Z_q. There exists a mapping e: $G \times G \rightarrow G_T$ that meets the following three conditions:

- *Bilinear:* For any two elements $g_1, g_2 \in G$, $a, b \in Z_q^*$, we have $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab} \in G_T$;
- *Non-degenerated:* There exists a $P \in G$ such that $e(P, P) \neq \rho$, where the ρ is the unit-element of group G.
- *Computable:* For any two elements $g_1, g_2 \in G$, we can compute $e(g_1, g_2)$ via an efficient computational technique.

We call the mapping e that meets the above three conditions as bilinear pairings. By applying a bilinear mapping on the supersingular elliptic curve, we can obtain a Diffie-Hellman group. Assume that the Diffie-Hellman group is G. The Computational Diffie Hellman (CDH) problem is hard, the Decisional Diffie Hellman problem (DDH) can be easily solved. Based on the characteristic of the bilinear pairings (Lu et al., 2014), we can calculate a user's PID and verify whether the PID is valid.

3.2 Problem Statement

Given the *location label*, a single requested packet and Bilinear pairings, the problem is to protect user's location privacy and reduce the aggregating time for k users in a distributed structure of LBS system.

For preserving user privacy, we design a LBS system, whose framework is shown in Figure 3, which consists of three key components: User Requests (USER), Pseudonym Identity Server (PIDS), and LBS Provider (LP). In this paper, we use a distributed LBS structure without involving a trusted central anonymizer. The LP can operate in accordance with relevant regulations and agreements of the LBS system. But it does not rule out the possibility that the LP is curious and desires to deduce users' location privacy, preference privacy and trajectory privacy. Users communicate with one another while following the system rules and agreements. They cannot collude with each other to infer other users' privacy information, and they also cannot collude with the LP.

1) *PIDS Server Initialization:* For a given secure parameter k, the PIDS server generates a 5-tuple (q, g, G, G_T, e) about bilinear pairings, where q is a k-bit prime number. Then the PIDS server initializes the LBS system with a suitable symmetric encryption

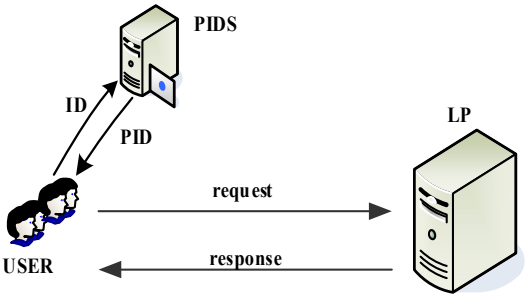

Figure 3: System model for privacy preserving.

algorithm $enc()$, pseudo-random function $f: \{0,1\}^* \rightarrow Z_q^*$ and two hash functions H_1: $\{0,1\}^* \rightarrow \{0,1\}^k$, H_2: $\{0,1\}^* \rightarrow G$. In this work, we assume that the PIDS server has held a public key and a private key (Pk_{pids}, Sk_{pids}). Finally, the PIDS server generates and publishes the system parameters (q, g, G, G_T, e, Pk_{pids}, f, H_1, H_2, $enc()$).

2) *User Registration:* A user u_i registers with the PIDS server by sending registration message. After receiving the registration message, the PIDS server computes $s=f(PK_{pids})$, and then computes user's pseudo-ID by using the value s and the symmetric encryption algorithm $enc()$. The pseudo-ID is represented by Pid_i: $Pid_i=enc_s(u_i \| r_i)$, where $r_i \in Z_q$. Then the PIDS server calculates the corresponding private key for user u_i: $Sk_i=H_1(Pid_i)$. Finally, the PIDS server returns the Pid_i and Sk_i to the user u_i. After receiving Pid_i and Sk_i , the user u_i can verify whether they are correct by checking $e(H_1(Pid_i), Pk_{pids}) =? e(Sk_i, g)$. If they are equal, the Pid_i and Sk_i are valid. Otherwise, they are invalid and the user will register into the PIDS server again.

3) *Request Submission:* If a user u_i initiates a request to the LBS provider, the LBS system would employ the LLB algorithm (will be described in Section IV) for protecting the user's privacy. After aggregating requests of users with k-anonymity and l-diversity properties, the user u_i becomes the *representative user*, and repackages the k users' request packets and gets an aggregated packet Ag. Then the *representative user* sends the aggregated packet to the LP. After receiving the aggregated packet, the LP processes it and returns a list of results to the k users. Users filter the results and find the one that is consistent with their own request from the list.

4 ALGORITHM DESIGN

In this section, we first propose three protocols including *request aggregation protocol*, *pseudo-ID*

exchange protocol and the *improved Privacy-preserving framework for Local-Area Mobile social networks* (PLAM) *protocol*. We then design the *location-label* based (LLB) algorithm for efficiently preserving the privacy of users.

4.1 The Request Aggregation Protocol

Without loss of generality, we assume that there is a user u_a who has not received any other queries from other users and wants to launch a request to the LP. Then the user u_a will initiate request aggregating message to aggregate with other k-1 user requests. The detailed aggregating process is as follows.

User u_a first broadcasts the request aggregating message. We assume that user u_b has received the broadcast message. There are three scenarios where user u_b will ignore the broadcast message sent by user u_a: *i*) user u_b has agreed to aggregate with other user; *ii*) the time t is zero in the request packet of user u_b; *iii*) the user u_b has sent aggregate request to other users and there are more than k/2 users who agree to join with user u_b.

If user u_b has neither sent aggregate request to other users, nor has agreed to aggregate with other users, and the time t is not zero, the user u_b is an ideal candidate for user u_a. If less than k/2 users agree to join with user u_b (assume m ($m \leq k/2$) users have joined with user u_b, and then the user u_b is *agent user* for the m users), the agent user u_b will respond to user u_a that "m+1 users (including user u_b and other m users who have joined with user u_b) agree to join".

When user u_a has received k-1 responses from other users, u_a informs the corresponding k-1 users and collect their request packets. Then user u_a repackages the packets from k users and get an aggregated package Ag. If the aggregated package Ag meets the l-diversity requirement, user u_a becomes the *representative user*, who sends the aggregated packet Ag to the LP. Otherwise, the aggregated package Ag will be discarded and user u_a informs the other k-1 users that their aggregation is failed. Then all the k users reset the time t and resubmit their requests.

4.2 The Pseudo-ID Exchange Protocol

We propose the pseudo-ID exchange protocol which can efficiently protect user location privacy in single query when the labels of users are the same and they belong to the sensitive locations. The pseudo-ID exchange protocol can also be used to protect user's trajectory privacy in continuous queries. Figure 4 shows three users u_a, u_b and u_c who come from different roads and gather in a sensitive location. If the users exchange their identities with each other in the sensitive location, the attacker cannot link a specific identity to a specific user. Although the locations are the same, the attacker is unable to distinguish the users. Therefore, it can indirectly protect users' location privacy. As shown in Figure 4, we can see that it can confuse the attacker who cannot infer which trajectory belongs to which specific user. So it can protect users' trajectory privacy. We introduce the process of exchanging identity information between two users in the following.

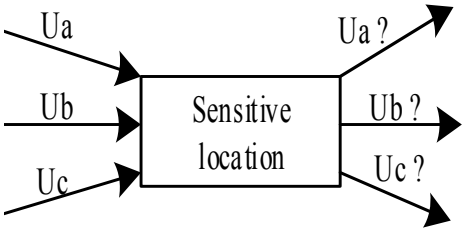

Figure 4: An example for exchanging PIDs.

Assume that there are two users u_a and u_b that have the same sensitive location. Assume that the two users exchange PIDs with the probability ρ. Due to the symmetry, we only need to analyze how user u_a changes his identity information. After receiving the Pid_b and private key Sk_b from the user u_b, user u_a re-verifies their identity by checking $e(H_2(Pid_b), PK_{pids}) =? e(Sk_b, g)$. If they are equal, user u_a will modify their identity. User u_a abandons his/her own identity information about Pid_a and Sk_a. Then user u_a regards the corresponding information about Pid_b as his/her own identity. The user u_b does the same as user u_a. Finally, user u_a and user u_b successfully finish the process of exchanging identities.

4.3 The Improved-PLAM Protocol

Lu et al., (Lu et al., 2014) proposed the PLAM protocol for preserving location and preference privacy in a distributed LBS system. With k-anonymity and l-diversity techniques, the PLAM protocol employed Privacy-preserving Request Aggregation to unite with k users only considering the case where users are in different locations. However, their locations may be the same, especially when they are in a sensitive location causing a loss in privacy of their location. In this work, we propose the improved PLAM protocol, for the scenario when the users have same location.

First, the *representative user* compares the

information *serve* in the *single packet requests* of k users. If there are at least l services between the k users, it means the aggregated packet Ag meets the requirements of l-diversity. Otherwise the aggregated packet Ag is discarded and the *representative user* informs the other k-1 users that the aggregation is failed.

After ensuring that there are at least l services using the LBS system, the *representative user* compares the location information (x, y) of all k users. If more than k/2 location information (x, y) are the same, it is necessary to exchange the identities of users who have the same locations by using the pseudo-ID exchange protocol. Finally, the *representative user* rearranges the aggregated packet Ag and sends it to the LBS provider.

4.4 Location-Label Based Algorithm

The pseudo code of *location-label* based algorithm (LLB) is shown as follows.

Algorithm 1: Location-Label Based (LLB) algorithm.

```
1: Broadcast the aggregation message;
2: Aggregate users' requests by using
request aggregation protocol;
3: if (k users aggregated together)
4: Compare the k users' location labels;
5:   if (the locations are identical and
sensitive locations)
6:    Exchange the users' PIDs by using
pseudo-ID exchange protocol;
7:   else if (the locations are identical
and ordinary locations)
8:    Call the improved-PLAM protocol;
9:   else if (the locations are different)
10:   Call the improved-PLAM protocol;
11:  end if
12: end if
```

User u_i first aggregates with other k-1 users through the *request aggregation protocol*. If there are k-1 users who agree to send request to the LP together with the user u_i, the user u_i is the *representative user* for the k users. Then the *representative user* will compare the k users' location labels. There are three kinds of situations based on the results of the comparison: *i*) The k users location labels are the same and their locations are sensitive locations, then we use the *pseudo-ID exchange protocol* for the subsequent processing; *ii*) The k users' location labels are the same and their locations are ordinary location, then we use the *improved-PLAM protocol* for the subsequent processing; *iii*) If location labels are different, the *representative user* will use the *improved PLAM protocol* for the subsequent processing.

5 SIMULATIONS AND ANALYSIS

In this section, we first introduce the simulation environment, and then give the simulation results of the compared algorithms. Please produce your figures electronically, and integrate them into your document.

5.1 Simulation Environment

We use OPNET (https://www.opnet.com/) to conduct our simulations, as OPNET can be used to construct complex network topologies and in sending or receiving messages. Assume that there is a region A of size {1.5km×1.5km} with 10×100 locations. For simulating the locations we construct a network consisting of 10 × 100 nodes and randomly assign these nodes as sensitive and ordinary location. For simulating the roads between locations we then construct the network is fully connected. In our experiments, we consider two scenarios as follows.

***Scenario*-1:** There are 100 users uniformly distributed in region A. We assume that user u_a in an ordinary location sends a request to the LP. The rest of the users randomly send their aggregation messages. The PID exchange probability p is fixed at 0.5. To ensure the l-diversity, we set $l = k/2$.

***Scenario*-2:** There are 100 users in region A, and most of them are densely distributed in sensitive locations, and only a few of them are distributed in ordinary locations. We assume that user u_a in a sensitive location sends their request to the LP. The rest of the conditions are the same as *Scenario*-1.

5.2 Simulation Results and Analysis

Figure 5 shows the simulation results for two compared algorithms (PLAM algorithm (Lu et al., 2014) and LLB algorithm) under *Scenario*-1. From Figure 5(a) we can see that the request delay of our proposed LLB algorithm is shorter than that of the PLAM algorithm. Furthermore, for our LLB algorithm, the request delay increases with the parameter k when $k < 20$ and reduces a little when $20 \le k \le 30$. However, the request delay goes up again when $k > 35$. This is because that aggregating more users needs more time. As there must be an agent user who agrees to join the user u_a when $k \ge 20$, the request delay slowly increases with the growth of user number k in LLB algorithm whereas quickly increases with the growth of user numbers k in the PLAM algorithm.

Figure 5(b) depicts the relationship between the number of users and the probability of guessing or identifying a user. From Figure 5(b), we can see that both LLB algorithm and PLAM algorithm have the same probability when $k \leq 15$; and the LLB algorithm has lower probability of guessing a user compared to the PLAM algorithm when $k \geq 20$. Since some of the k users have the same locations (larger k means more users have same locations), and thus our LLB algorithm can better protect location privacy of user compared to the PLAM algorithm.

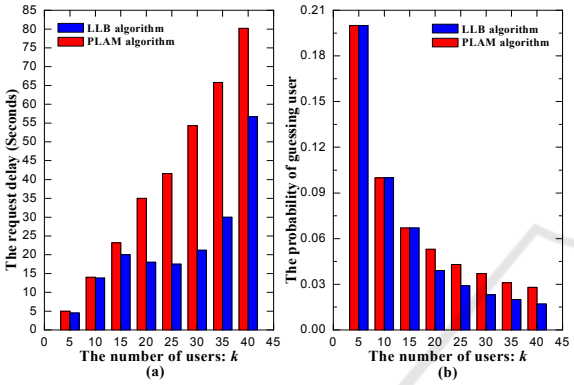

Figure 5: The simulation results for Scenario-1.

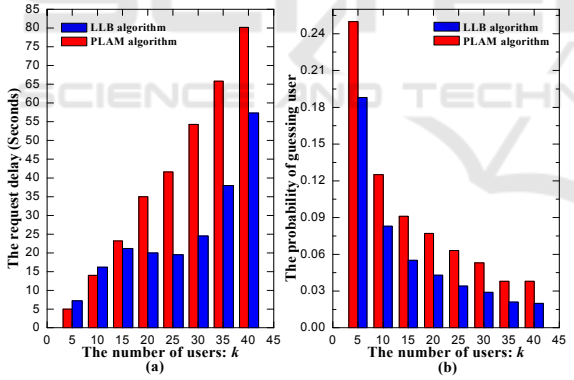

Figure 6: The simulation results for Scenario-2.

Figure 6 compares the LLB algorithm and PLAM algorithm under *Scenario*-2. Figure 6(a) shows the similar results as that in Figure 5(a). The aggregating time of our proposed algorithm is significantly smaller compared to that of the PLAM algorithm when k >15. However, as user u_a is in a sensitive location in *Scenario*-2, the locations of k users merged by user u_a may be identical. Then the proposed LLB algorithm exchanges the users IDs through *pseudo-ID exchange protocol* and the process of exchanging IDs is time consuming. Thus, it is noticed that when the number of users is small, e.g. $k \leq 10$, the PLAM algorithm has lower request delay. However, the overall performance of our proposed algorithm is superior to the PLAM algorithm in *Scenario*-2. Figure 6(b) shows that the LLB algorithm has much lower probability of leaking the privacy of a user compared to the PLAM algorithm. Hence, it can more efficiently protect user privacy. Compared to the Figure 5(b), the probability of guessing a user with LLB algorithm is lower in *Scenario*-2 than that in *Scenario*-1. This is because when users have same locations, the LLB algorithm employs *pseudo-ID exchange protocol* to reduce the probability of guessing user.

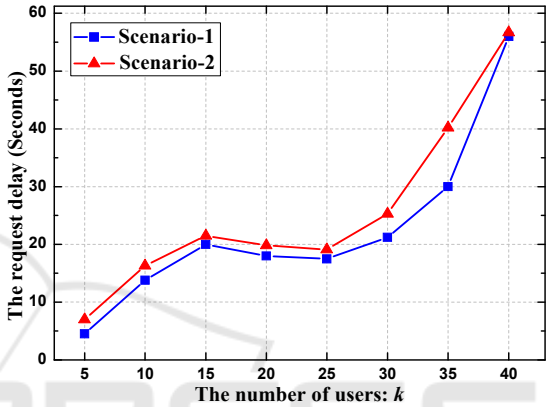

Figure 7: Performance of LLB algorithm under different scenarios.

Figure 7 shows the performance of the LLB algorithm when user u_a is in an ordinary location (*Scenario*-1) and a sensitive location (in *Scenario*-2), respectively. For a user in sensitive location, before sending a request, the user aggregates with k-1 other users. The k locations of users are very likely identical and the LLB algorithm uses the *pseudo-ID exchange protocol* for exchanging user identities. As the process of exchanging identities is time consuming, the request suffers a longer delay in *Scenario*-2 than that in *Scenario*-1.

6 CONCLUSIONS

In this paper we study the problem of privacy preservation for users have same location in a LBS system. To protect the location privacy, preference privacy and trajectory privacy of users in a distributed structure of LBS system, we propose a *location-label* based algorithm that includes three key protocols: the *user requests aggregation protocol*, the *pseudo-ID exchange protocol* and the *improved PLAM protocol*. We conduct extensive

simulation experiments to evaluate the performance of our algorithm. The simulation results show that the proposed algorithm outperforms the existing approach.

ACKNOWLEDGEMENTS

This research was partially supported by the National Grand Fundamental Research 973 Program of China under Grant (2013CB329103), Natural Science Foundation of China grant (61271171, 61571098), China Postdoctoral Science Foundation (2015M570778), Guangdong Science and Technology Project (2012B090400031, 2012B090500003, 2012B091000163), and National Development and Reform Commission Project.

REFERENCES

Li, Y., Yiu, M., 2015. *Route-Saver: Leveraging Route APIs for Accurate and Efficient Query Processing at Location-Based Services*. IEEE transactions on knowledge and data engineering (TKDE).

Yang, D., Fang, X., Xue, G., 2013. *Truthful incentive mechanisms for k-anonymity location privacy*. IEEE INFOCOM.

Bettini, C., Mascetti, S., Wang, X., Jajodia, S., 2007. *Anonymity in location based services: Towards a general framework*. IEEE International Conference on Mobile Data Management.

Niu, B., Li, Q., Zhu, X., Cao, G., Li, H., 2014. Achieving K-anonymity in Privacy Aware Location Based Services. IEEE INFOCOM.

Zhu, X., Chi, H., Niu. B., 2013. *When k-anonymity meets cache*. IEEE GLOBECOM.

Shao, J., Lu, R., Lin, X., 2014. *FINE: A Fine-Grained Privacy-Preserving Location Based Service Framework for Mobile Devices*. IEEE INFOCOM.

Niu, B., Zhu, X., Li, W., et al, 2015. *A Personalized Two-Tier Cloaking Scheme for Privacy-Aware Location-Based Services*. IEEE International Conference on Computing, Networking and Communications (ICNC).

Lu, R., Lin, X., Shi, Z., et al, 2014. *PLAM: A Privacy-Preserving Framework for Local-Area Mobile Social Networks*. IEEE INFOCOM.

Uchiyama, A., Fujii, S., Maeda, K., et al, 2013. *UPL: opportunistic localization in urban districts*. IEEE Transactions on Mobile Computing, 12(5).

Li, H., Sun, L., Zhu, H., et al, 2014. *Achieving Privacy Preservation in WiFi Fingerprint Based Localization*. IEEE INFOCOM.

Shu, T., Chen, Y., Yang, J., 2014. *Multi-lateral Privacy-Preserving Localization in Pervasive Environments*. IEEE INFOCOM.

Beresford, A., Stajano, F., 2014. *Mix Zones: User Privacy in Location-aware Services*. IEEE Conference on Pervasive Computing and Communications Workshops.

Chen, Z., Hu, X., Ju, X., et al, 2013. *LISA: location information scrambler for privacy protection on smartphones.* IEEE Conference on Communica -tions and Network Security.

Feng, Y., Liu, P., Zhang, J., 2012. *A Mobile Terminal Based Trajectory Preserving Strategy for Continuous Querying LBS Users*. IEEE International Conference on Distributed Computing in Sensor Systems.

Mohammed, N., Fung, B., Debbabi, M., 2009. *Walking in the crowd, Anonymizing trajectory data for pattern analysis*. The 18th ACM Conference on Information and Knowledge.

Wang, Y., Peng, J., He, L., et al. *LBSs Privacy Preserving for Continuous Query based on Semi-honest Third Parties.* IEEE International Perform -ance Computing and Communications Conference.

Liao, Y., Hsiao, C., 2011. T*he improvement of ID-based remote user authentication scheme using bilinear pairings.* IEEE International Conference on Consumer Electronics, Communication and Networks.

Niu, B., Zhu, X., Li, Q., Chen, J., Li. H., 2015. *A novel attack to spatial cloaking schemes in location-based services*. Future Generation Computer Systems.

OPNET. https://www.opnet.com/.