Kuang *et al. EURASIP Journal on Wireless Communications and Networking* (2020) 2020:1
https://doi.org/10.1186/s13638-019-1618-7

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Using location semantics to realize personalized road network location privacy protection

Li Kuang, Yin Wang, Xiaosen Zheng, Lan Huang and Yu Sheng*

Abstract

With the rapid development of location-based services in the field of mobile network applications, users enjoy the convenience of location-based services on one side, and they are exposed to the risk of privacy disclosure on the other side. Attackers may attack based on the semantic of the user's location and user's query location. A few of existing works on location privacy protection consider to protect the user's location and his query location simultaneously, while the query location may reflect his requirement. In this paper, based on the existing location privacy protection framework, we first generate sensitive weight documents based on the user's sensitivity to different location semantics automatically, then obtain the best collaborative segment for *k*-anonymity of the user's location by using the reinforcement learning algorithm, and finally, the bidirectional *k*-disturbance of the user's location and query location is performed based on the location semantics in real road network environment. The experiment verifies the effectiveness of the proposed method.

Keywords: Location-based service, Location semantics, Road network, Location *k*-disturbance, Reinforcement learning

1 Introduction

In recent years, with the increase of computing power and storage capacity of mobile computing equipment, mobile phones, vehicle terminals, and wearable intelligent devices become more and more popular. At the same time, location-based services (LBS) are blooming, such as Google Maps, Bing Maps, Facebook, and Twitter. Users can obtain related services based on their locations, which can be acquired by the inner GPS (Global Positioning System) in the mobile equipment.

When enjoying the convenience of location-based service, people are exposed to the risk of privacy disclosure, which may result in economic and honorary loss and even live threat. When using LBS, the user sends his own location information and service request to the location provider. Although the location provider claims that he is trustworthy, if the server data are stolen by the attacker, even the location provider himself is the attacker, and users' location information will be leaked. Based on the location information, the attacker can infer the user's identity, behavior, habits, health status, social relations, and other privacy information.

The existing location privacy framework can be divided into two categories based on whether the third-party trusted server is used. In order to reduce the client computational cost, most works use third-party trusted servers. However, the third-party trusted server is not absolutely credible, since it can also become the objective of the attack, thus resulting in information disclosure. Although we use the existing location privacy service framework, there is a difference when the service request is sent to ensure that the user's real location cannot be uniquely identified.

Existing location privacy protection methods mainly include *k*-anonymous, space cloaking, dummy position generation, and encryption. However, few works on location privacy protection consider the semantic information behind the location of the real road network or consider protecting the user's location and his query location simultaneously. For example, suppose that a user searches for "nearby skin hospital" after drawing money from the bank. Previous work only provides protection to the user's location, and not to the user's query location which may reflect the user's need. If such sensitive

* Correspondence: shengyu@csu.edu.cn
School of Computer Science and Engineering, Central South University, Changsha 410075, China

© The Author(s). 2019 **Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

need is obtained by malicious attackers, the user's privacy will be leaked, so the user's query location also needs to be protected.

There are three problems with the existing location privacy protection solutions. First, different users have different sensitivity weights to different location semantics, while users' different preferences are not incorporated into existing solutions. For example, if John is a doctor, the hospital is not a sensitive place for him and it is normal that he appears in the hospital. But if Alice is a supermarket employee, she does not go to hospital often, so the hospital is a sensitive place for her. Therefore, each user should obtain personalized protection due to different location semantics sensitivity. Second, when k-anonymizing a user, his location is anonymized with other sections of the road except for his real road section, but the user's sensitive location semantics in other sections are not considered. For example, if Bob is sensitive to location semantics A and B, Bob needs three segments to achieve k-anonymity. If A and B on the three selected road segments are more than that on the other three unselected segments, although the k-anonymity requirement is reached, the algorithm is still not satisfactory since it does not consider the user's sensitive location semantics on the candidate roads in the anonymity set. Third, protecting the user's location is important, but the location that the user has queried is also important, since the query location may reflect the user's living habits, personal health, or status. For example, Bob inquired about "dermatology hospitals" near "banks," and the two locations are sensitive to Bob, so it is necessary to protect the user's location and the query location at the same time.

In order to solve the three problems, in this paper, we propose a personalized location privacy protection framework based on location semantics, which consists of three steps: first, we propose a personalized sensitivity weight assignment algorithm, by which the location semantics sensitivity preference of the user can be obtained quickly without manual settings; second, in order to protect the user's location, we propose a collaborative road segment matching algorithm using reinforcement learning, which gets the safest collaborative road segments for k-anonymity based on the user's sensitivity to location semantics; Third, in order to protect the user's query location, we propose a bidirectional location k-disturbance algorithm based on location semantics, which can k-disturb the user's location and query target at the same time, and greatly reduce the privacy leakage caused by the intersection of anonymity sets when continuous queries are encountered.

The remainder of the paper is organized as follows. Related work is discussed in Section 2. The preliminaries of this paper are given in Section 3. The proposed approach is illustrated in Section 4. The experimental results are presented in Section 5. And finally, the conclusions and future work are given in Section 6.

2 Related work

The existing work on location privacy protection can be divided into the following 4 categories:

(1) k-anonymous [1–9]. k-anonymous is the basis of many privacy protection methods. For example, Gruteser and Grunwald [1] proposed the location k-anonymous, and the main idea is that the location of the user who sends the service request cannot be differentiated with other k-1 user's location so that the probability that the user is identified does not exceed $1/k$. Yiu et al. [2] in the Space Twist program introduced a trusted third-party server so that users can avoid sending their own specific location to the location provider, which reduces the possibility of location information disclosure. In addition, the introduction of a third-party trusted server can greatly reduce the client's computing overhead, and the user can access to services more quickly. Mokbel et al. [3] proposed a k-anonymity protection method, which also needs to introduce a trusted third party. But the k-anonymity protection method will produce a k-1 dummy location of the generalized area to interfere with location providers so that the possibility of identifying the user for the location provider cannot be greater than $1/k$. In addition, trusted third-party server also has a filtering function, which can remove the dummy location's request results and return request service results to the client.

(2) Space cloaking [10–17]. Space cloaking is a popular location privacy protection method. Chow et al. [10] proposed the *Casper cloak* algorithm, which uses the quadtree data structure to divide the spatial structure into H layers, each with information about the entire spatial structure. The algorithm allows the user to customize the minimum anonymous area $A_{\min}$ and generalize the user's location into the area. Sun et al. [11] designed a geolocation tag to distinguish sensitive locations and common locations in order to minimize the response time of the requesting service and to protect the user's location privacy through spatial stealth technology.

(3) Dummy position generation [18–22]. Dummy position generation method is commonly used. Kido et al. [18, 19] first proposed a dummy location method in the study of location privacy protection. Guo et al. [20] realized trajectory privacy protection by using the dynamic

pseudonym change mechanism and user-controllable dummy location generation mechanism combining with a geometric transformation algorithm.

(4) Encryption [23–29]. Instead of using a third-party trusted server, the method directly sends encrypted requests to the server and then decrypts in the client. Although the security of this method is very high, the computational cost is huge, the deployment is complex, and the algorithm needs to be optimized. Khoshgozaran et al. [23] proposed a Hilbert curve based on the encryption method. The user's location and points of interest are transferred from the two-dimensional coordinates to one-dimensional encryption space, and one-dimensional encryption space, transferred through two different parameters of the Hilbert curve, still maintains the proximity in two-dimensional space, so that *k*-nearest neighbor query and range query can also be performed in one-dimensional encrypted space. PIR (private information retrieval) [24] method replaces the homomorphic comparison step with an unconscious transfer to achieve a more secure solution. The PIR method has the advantage of high privacy protection security, but the client computing is very large. Lu et al. [25] proposed PLAM privacy protection framework, which uses homomorphic encryption technology to protect the user's privacy, and cannot only achieve satisfactory privacy requirements, but also resist most of the external attacks.

Most of the existing work on location privacy protection do not take the real road network situation, the semantic meaning of locations and user's sensitive preference to location semantics into account. More and more researchers begin to devote themselves to privacy protection based on location semantics [30–42]. Damiani et al. [30] considered the semantic information of the user's location and introduce a framework that includes semantic perception ambiguities to generate chaotic space. Xiao et al. [31] proposed *p*-sensitivity mechanism. The mechanism uses PE-Tree to divide the query into sensitive and non-sensitive categories, but the query is not protected. If the query location is sensitive to the location semantics, it is possible to disclose the user's privacy. The paper [32, 33] proposed the PROBE framework, in which the user can edit the privacy document according to his own sensitivity to the location semantics, but the operation is too complicated. Xue et al. [34] protected the user's location privacy by using the generated semantic position. In order to improve the *k*-anonymous protection efforts, it is necessary to ensure that each query is at least related to *k*-different semantic positions.

In summary, the main contributions of our paper are as follows:

(1) We protect the security of the user's location and the query location simultaneously and conduct the bidirectional *k*-disturbance based on the semantics of the user's location and query location.

(2) User's sensitivity preference to different location semantics can be obtained more conveniently without manual settings.

(3) According to the users' sensitivity preference to different locations, the collaborative road segments with the highest security for users can be matched to meet the demand for *k*-anonymity.

3 Background and problem definition

3.1 Location service framework and workflow

As shown in Fig. 1, this paper uses a semi-trusted anonymous server location service framework [43]. The framework workflow is shown in Fig. 1 marked with Arabic numerals 1 to 4. The first step, two-way concealment: the client will form the *k*-anonymous user's location and the query request location information and submit to the intermediate server together with sensitive location semantics information selected by the user; the second step, the query request: the intermediate server will submit the *k*-anonymous location information and query information to the location service provider; the third step, the intermediate server receives the return result set of the location service provider, and the intermediate server does comprehensive scoring combining the result set, the user's sensitive location semantics information, and the distance between the user and the query result, and then, the intermediate server can form a query target rank; the fourth step, the client receives the data processed by the intermediate server, filters out the redundant data, and finally sends the request result to choose corresponding to the user's location.

3.2 Location semantic information

Definition 1: location semantics. It represents the true meaning behind the location. All the location semantics is divided into n classes, TP = {tp_1, tp_2, tp_3,...,tp_n}, TP is all types of position semantics.

Definition 2: road information. The research object of this paper is the location semantic information in the real road network, and $road_i$ (rid, Set_{adj},Set_{sem}) is used to denote the road information, where rid denotes the road number, Set_{adj} denotes the number set of adjacent road, and Set_{sem} denotes the collection of all location semantic on the road.

Definition 3: user location. We use $Upos_i(rid,id,x,y,tp)$ to denote the user's location, where rid is the road number, id is the user identifier, x is the longitude of the user's location, y is the latitude of the user, and tp is the type of the user's location semantics.

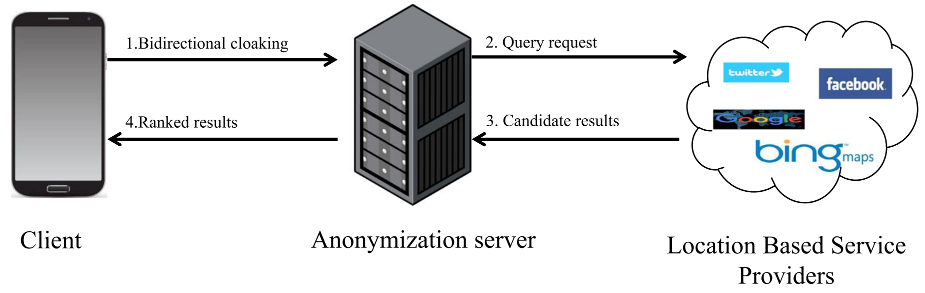


Fig. 1 Shows a semi-trusted anonymous server location service framework, which includes four steps: (1) bidirectional cloaking, (2) query request, (3) candidate results, and (4) ranked results

Definition 4: location semantics sensitive set. Users will set up sensitive collections based on their sensitive location semantics before using the service for the first time. We use SEN_u={($\mathrm{sen}_{\mathrm{tp}_1}$,level), ($\mathrm{sen}_{\mathrm{tp}_2}$,level), ($\mathrm{sen}_{\mathrm{tp}_3}$,level) ,…, ($\mathrm{sen}_{\mathrm{tp}_n}$,level)}to represent the user's sensitive location semantics set, $\mathrm{sen}_{\mathrm{tp}_i}$ represents different types of user's sensitive location semantics, and level represents the sensitivity level.

Definition 5: collaboration sections. Use Cr_i to indicate the *i*th collaborative segment required by the user's demand for *k*-anonymity. $\mathrm{Cr}_{\mathrm{set}}$ represents a collection of all the collaborative segments required by the user.

Definition 6: privacy requirements. The user's privacy requirements can be represented as *PPD* (privacy protection demand) = (SEN_u,*k*). SEN_u denotes the user's own sensitive location semantics collection, and *k* denotes that the process of *k*-disturbance of the user's location and the query location needs to be done respectively.

3.3 Problem definition

The purpose of this paper is to solve the user's location security and user's query security. On this basis, in order to make users get the service of high privacy security and service quality, in the background of real road network, taking the user's location semantics sensitivity and the distance between the user and the query location into comprehensive consideration, give a comprehensive rank, which provides the user with reference basis to choose.

In this paper, we make the following assumptions on the attack: (1) the attacker has the map information, the location semantics information on each road section; (2) the attacker has the user's location information and the user's query information; (3) the attacker has the id identifier information of each inquirer.

4 Method

4.1 Personalized sensitivity weight assignment algorithm

4.1.1 Definition 7 Location semantic sensitive weights

Although each user has a location semantic privacy collection SEN_u that can know what type of location semantic the user is sensitive to, but cannot reflect the user's sensitivity to different location semantics. So we design a personalized sensitivity weight partitioning algorithm for different users. We appoint that the value of the sum of the user's location semantic sensitive weight is 1, expressed as

$$w_{tp_1} + w_{tp_2} + w_{tp_3} + \ldots + w_{tp_n} = 1 \qquad (1)$$

As shown in Fig. 2, we design a semi-automated method of obtaining user-sensitive weights. When the user first uses the service, he needs to drag the user-sensitive location semantics from all location semantics columns to the sensitive location semantics column. Sensitivity columns 1 to 3 indicate the degree of sensitivity. We will divide location semantic sensitivity into three levels, recorded as level 1, level 2, and level 3, and each level has the same location semantics sensitivity and the assigned sensitive weight, each sensitivity of the position semantics in level 1 > level 2 > level 3. The weight of the location semantics sensitivity that has not been dragged into the sensitive location semantics column is 0. With the information, the middle of the server can automatically figure out the user's location semantic sensitive weight so that users do not need to fill in the privacy documents, which makes it faster to access to the user's location semantic sensitivity and increases the user experience of service.

If the amount of sensitive location semantics in level 1 is n_1, in level 2 is n_2, and in level 3 is n_3. Because the location semantics sensitivity in each level is the same, marking it as w_i, for level 1 is w_1, for level 2 is w_2, and for level 3 is w_3. Based on the above information and formula (1), we can deduce the following formula:

$$n_1 \cdot w_1 + n_2 \cdot w_2 + n_3 \cdot w_3 = 1 \qquad (2)$$

We already know each sensitivity of the position semantics in level 1 > level 2 > level 3, so we can deduce the following formula:

$$w_1 > w_2 > w_3 \qquad (3)$$

Setting up (1) and (2) simultaneously, we can get the region of solution to w_1, w_2,and w_3, calling the location semantic sensitive weight FR (feasible domain). Finally,

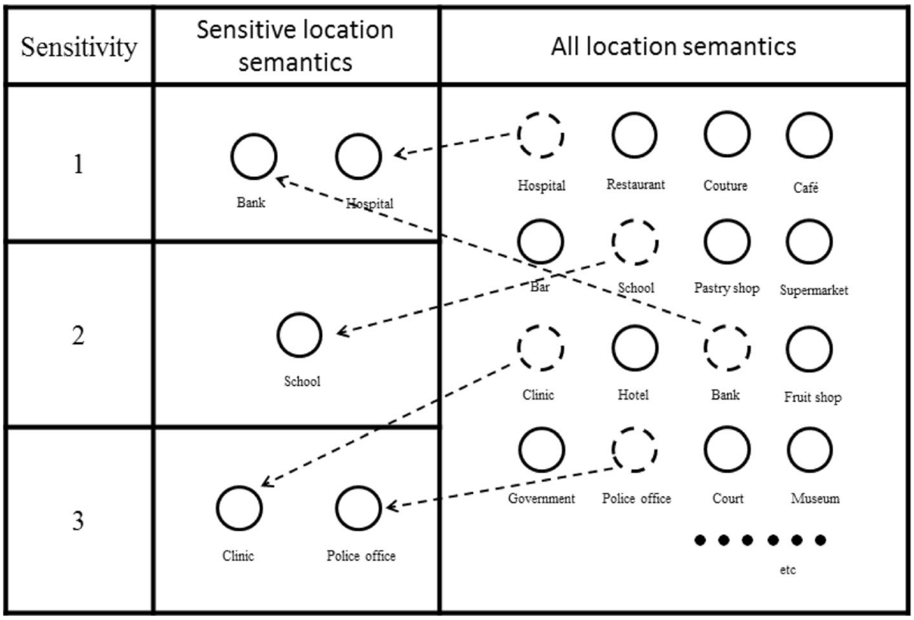


Fig. 2 Shows an example of sensitive location semantic selection. The sensitivity columns 1 to 3 indicate the degree of sensitivity, and the user needs to drag the user-sensitive location semantics from all location semantics columns to the sensitive location semantics column

the intermediate server randomly selects the value of w_1, w_2, and w_3 from the location semantic sensitive weight feasible domain, as the user's personalized position semantically sensitive weight. Transfer SEN$_u$={(sen$_{\text{tp}_1}$,level), (sen$_{\text{tp}_2}$,level), (sen$_{\text{tp}_3}$,level) ,..., (sen$_{\text{tp}_n}$,level)} into SEN$'_u$ ={(sen$_{\text{tp}_1}$,w), (sen$_{\text{tp}_2}$,w), (sen$_{\text{tp}_3}$,w),..., (sen$_{\text{tp}_n}$,w)}. The personalized sensitivity weight assignment algorithm is shown as follows:

Algorithm 1. PSWA (*Personalized Sensitivity Weight Assignment*) Algorithm

```
Input: Privacy protection demand PPD( $SEN_u$,k)
Output: The weight of positional semantics in each sensitive level $w_1$, $w_2$, $w_3$
For level in $SEN_u$
    count level1, level2, level3=$n_1, n_2, n_3$
End For
Draw feasible region based on conditions 0<$w_3$<$w_2$<$w_1$<1, $w_2$>$\frac{1-n_1}{n_2+n_3}$
Randomly select a point from the FR notes on P
P[0] = $w_1$,P[1]= $w_2$, $w_3$=1-$w_1$-$w_2$
For level in $SEN_u$
    if level== level1 then
        level=$w_1$
    end if
    if level== level2 then
        level=$w_2$
    end if
    if level== level3 then
        level=$w_3$
    end if
End For
```

4.2 Cooperative segment matching algorithm

As shown in Fig. 3, the location semantics marked as red in the figure are user-sensitive semantic information, and the semantic information marked as black is non-sensitive semantic information. In the picture, the user's location is "Supermarket," and R_1 is the user's location road. In order to protect the user's query location, we use the classic location disturbance protection mechanism. The core idea is to select an area on the user's location, which includes the location of the user and the other k-1 users, to confound the user's true location. In this paper, we discuss semantic-based location privacy protection, so the difference from the existing method is that we choose the k-1 class location with the user's location to confound according to the different type of semantics of the user's location. For example, if the user selects k = 3, the other two types of locations where the user is located are preferred to meet the location k-disturbance requirement.

If the chosen k is greater than the category of the location semantics contained in the road segment, the user's location road and the adjacent section are used to complete the user's k-anonymous request together. For example, the user selects k = 8, while the user's location road has five types of location semantics which cannot meet the user's needs. We can obtain adjacent sections collection Set$_{\text{adj}}$ of the user's location road from road$_1$(rid,Set$_{\text{adj}}$,TP). If a random section is selected from Set$_{\text{adj}}$ as a collaborative section, a path with higher semantic sensitivity may be chosen, which exposes the user's sensitive location with a high probability and increases the risk of user privacy disclosure. Therefore, the ability to accurately and intelligently select low-sensitivity roads as a collaborative link is critical to the privacy of users.

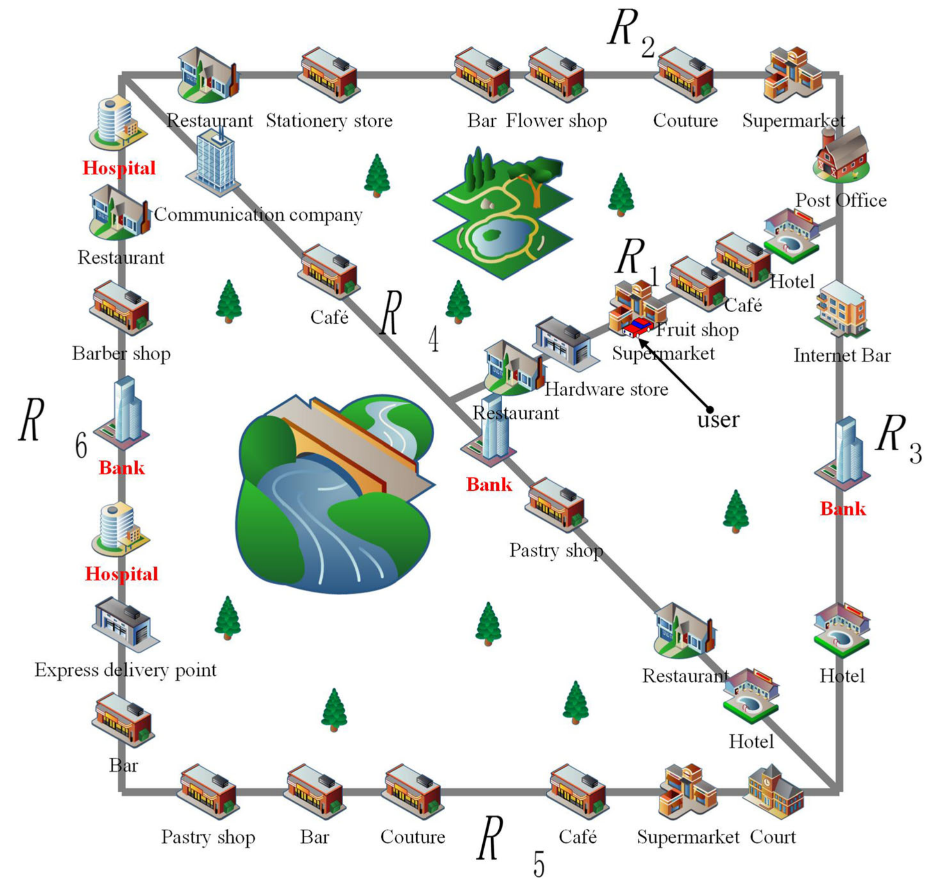


Fig. 3 Shows an example of a simplified road network semantics graph. The location semantics marked as red in the figure are user-sensitive semantic information, and the semantic information marked as black is non-sensitive semantic information

The process of finding a collaborative segment can be seen as an intensive learning process that continuously adds collaborative segments to meet the user's *k*-anonymity requirements. The advantage of intensive learning is that there is no "supervised" signal and only has the "reward" signal. The whole process of finding a collaborative road segment is independently performed by a computer, the process has a certain degree of randomness, and the higher the randomness of anonymity, the higher the security. In this learning process, the two main bodies of "Agent" and "Environment" are included; "Agent" is a learner and a decision-maker and achieves goals by interacting with the environment [44], and the interaction process is shown in Fig. 4: "Agent" makes action A_t based on the current state S_t, and "Environment" responds to get the state S_{t+1} and the reward R_{t+1} at the next moment, in which the state S and the reward R at the same moment appear in pairs, and the ultimate goal is to maximize the sum of the rewards $\sum_{t=0}^{t=T} R_t$. In the decision-making process of obtaining the sum of the maximum rewards, the "brain" that leads the dominant process to be better is the Q table [45, 46], and the value in the Q table represents the return value of performing an action in a specific state, and "Agent" through continuous updating and looking up the table to find the execution action with the highest current return. The R matrix is similar to the Q table, it represents the return value of the "Agent" to each state, and the main purpose of updating the Q matrix is to provide data so that the "Agent" can find the optimal execution action in the R matrix. The update of the Q value can be defined as follow:

$$Q(s,a) = \gamma_1 \cdot R_{ca}(s,a) + \gamma_2 \cdot R_{na}\left(s^{'}, a^{'}\right) + \lambda \cdot \ \max\{Q(\tilde{s}, \tilde{a})\} \tag{4}$$

where s represents the current state, a represents the current action, $\tilde{s}$ represents the next state that is not determined, and $\tilde{a}$ represents the next action that is not determined. $Q(s,a)$ represents the expected maximum

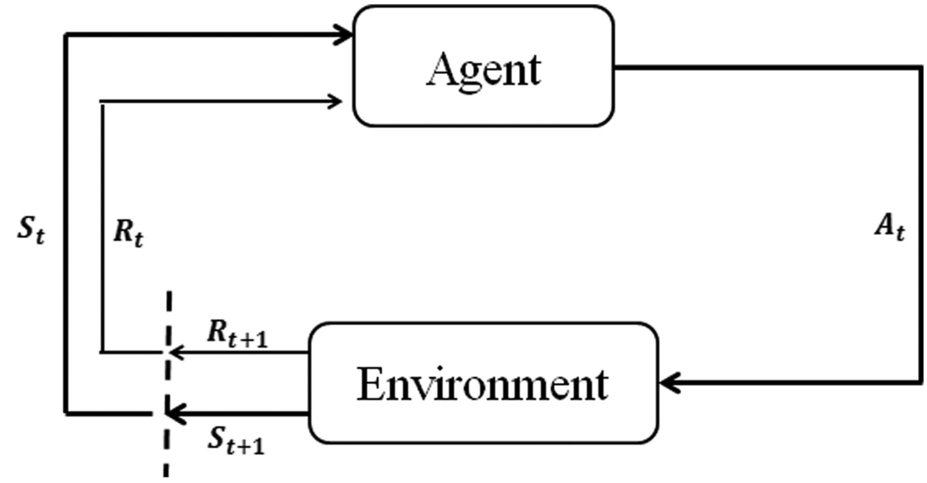
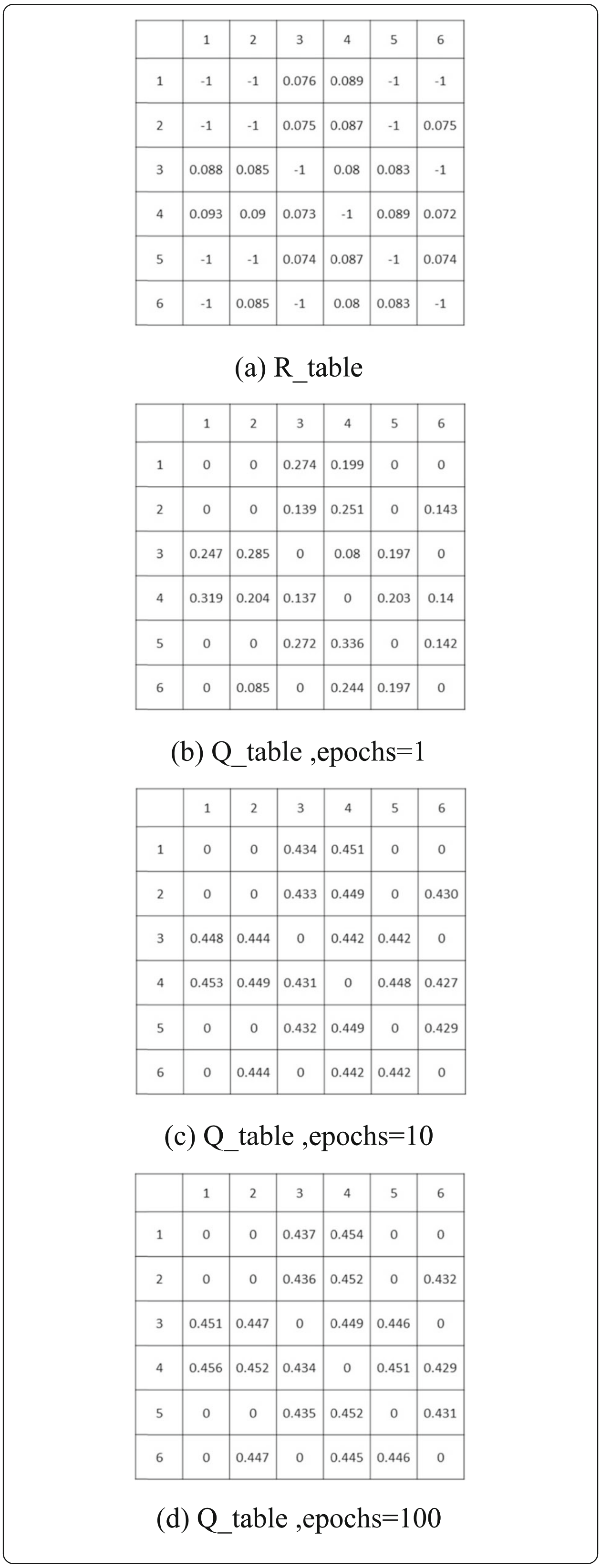


Fig. 4 Shows the interaction process between "Agent" and "Environment," "Agent" is a learner, "Environment" is the learner's environment, A_t, S_t, and R_t represent current action, current state, and current reward, respectively, S_{t+1} and R_{t+1} represent the action of the next moment and the state of the next moment, the dotted line in the graph represents the time boundary

benefit that can be obtained by taking action a under state s. γ_1, γ_2, respectively, represent the greedy factors of the return values $R_{pa}(s,a)$ and $R_{na}(s^{'},a^{'})$ represents the return value immediately obtained by the current action, $R_{na}(s^{'},a^{'})$represents the next action return value, and $s^{'}$, $a^{'}$, respectively, represent the next determined state and action. λ represents the greedy factor of the maximum value of $Q(\tilde{s},\tilde{a})$, and $\max\{Q(\tilde{s},\tilde{a})\}$ represents the expected maximum benefit that can be obtained by taking action $\tilde{a}$ in state $\tilde{s}$.

By using the personalized sensitivity weight division algorithm in Section 4.1 to derive the position semantic sensitivity weight feasible domain, we can get the number of semantic sensitivity levels of each location of the user, then we mark n_1=3, n_2=5, and n_3=10, and get the feasible solution w_1=0.1, w_2=0.08, and w_3=0.03 through formulas (2) and (3). Because the higher the sensitivity weight, the more sensitive the user is to the semantics of the location, which means that the reward value is inversely proportional to the sensitivity weight, we use R_1, R_2, and R_3 to represent the return values of the three sensitive levels, and find R_1=0.03, R_2=0.08, and R_3=0.1. We use an example to explain how to find the maximum value action by updating Q table, and the initial value of Q table is 0. Taking the location semantic road network of Fig. 3 as an example, the sum of the return values of all position semantics of each road is taken as the return value of the road segment. R table is shown in Fig. 5a, the first row of the R table represents the name of the road segment, and the value in the table represents the return value of one road matching to another road, where the value of − 1 indicates that the two road segments cannot communicate. The Q table is the same as the R table, the first row column indicates the name of the road segment, and the value in the matrix indicates the expected maximum benefit that can be obtained by taking the action a under the state s. Q table is shown in Fig. 5b–d, where different epochs represent different training rounds, Fig. 5b shows the matrix obtained after a round of training in the initial state where Q table is all 0 when epochs = 1, Fig. 5 c and d are the matrices obtained by training with epochs = 10 and epochs = 100, respectively, and the values in the Q table obtained by training epochs = 10 and epochs = 100 are very close, which indicate that Q table has converged when epochs = 10.

If the required k value of k-anonymity is 15, the user's location semantic on the road segment R_1 of Fig. 3. There are only six location semantics on R_1, which cannot meet the requirement of k = 15, and need other road segments to cooperate to complete the k-anonymous task. Sometimes, the k value will be larger, and then more road segments are needed as collaborative road segments. The collaborative road segment can be found through the updated Q table, where the user can be located as R_1 and corresponding to the first row or column in the Q table (randomly take the rows in the table). The largest value in the first row is Q (1, 4), so we choose R_4 as the collaborative segment of R_1, but the requirement of k = 15 cannot be satisfied at this time, we need to continue to match R_4's collaborative segments, and the largest value in the fourth row is Q (4, 1), but R_1 is already a collaborative road segment, we need to select the section with the largest remaining Q value. At this time, the largest value is Q (4, 2) and k = 18 satisfies the anonymity requirement of k = 15, and the collection of collaborative segments is R_1, R_4, and R_2. If the user-required k is larger, more road segments can



	1	2	3	4	5	6
1	-1	-1	0.076	0.089	-1	-1
2	-1	-1	0.075	0.087	-1	0.075
3	0.088	0.085	-1	0.08	0.083	-1
4	0.093	0.09	0.073	-1	0.089	0.072
5	-1	-1	0.074	0.087	-1	0.074
6	-1	0.085	-1	0.08	0.083	-1

(a) R_table

	1	2	3	4	5	6
1	0	0	0.274	0.199	0	0
2	0	0	0.139	0.251	0	0.143
3	0.247	0.285	0	0.08	0.197	0
4	0.319	0.204	0.137	0	0.203	0.14
5	0	0	0.272	0.336	0	0.142
6	0	0.085	0	0.244	0.197	0

(b) Q_table ,epochs=1

	1	2	3	4	5	6
1	0	0	0.434	0.451	0	0
2	0	0	0.433	0.449	0	0.430
3	0.448	0.444	0	0.442	0.442	0
4	0.453	0.449	0.431	0	0.448	0.427
5	0	0	0.432	0.449	0	0.429
6	0	0.444	0	0.442	0.442	0

(c) Q_table ,epochs=10

	1	2	3	4	5	6
1	0	0	0.437	0.454	0	0
2	0	0	0.436	0.452	0	0.432
3	0.451	0.447	0	0.449	0.446	0
4	0.456	0.452	0.434	0	0.451	0.429
5	0	0	0.435	0.452	0	0.431
6	0	0.447	0	0.445	0.446	0

(d) Q_table ,epochs=100

Fig. 5 a is a R_table, the first row of the R table represents the name of the road segment, and the value in the table represents the return value of one road matching to another road, where the value of − 1 indicates that the two road segments cannot communicate. Q table is shown in **b**–**d**, where different epochs represent different training rounds. **b** Q table, epochs = 1. **c** Q table, epochs = 10. **d** Q table, epochs = 100

be matched as the cooperation road segment to meet the user's needs, and the collaborative road segment matching algorithm can intelligently and quickly find the cooperation road segment with the user's road segment. The following is the collaborative segment matching algorithm:

Alogorithm 2.CSM(*Cooperative Segment Matching*) Algorithm

```
Input: Privacy protection demand $PPD(SEN_u^{'},k)$;Road information $road_i(rid,Set_{adj},TP)$;
User information $Upos_i(rid,id,x,y,tp)$
Output: $Cr_{set}$ a set of all collaborative segments required by the user
1. Get rid in $Upos_i$
2. Get $Set_{adj},TP$ in $road_{user}$ by rid
3. Randomly select m segments around $road_{user}$ form a Net
4. Get $w_1, w_2, w_3$ from $SEN_u^{'}$
5.For $road_i$ in Net
6.      Get $TP$ from $road_i$
7.      $n_i$ = len($TP$)
8.      $R_{ca}(s,a) = \frac{\sum_{i=1}^{3} n_i \cdot R_i}{\sum_{i=1}^{3} n_i}, R_{na}\left(s^{'},a^{'}\right) = \frac{\sum_{j=1}^{3} n_j \cdot R_j}{\sum_{j=1}^{3} n_j}$
9.      If s==a or (a not in s'$Set_{adj}$)
10.         $R_{ca}(s,a)$ = -1, , $R_{na}(s',a')$ = -1
11.     End If
9.      Q_table = np.zeros(m,m)
10.     Randomly chose coordinate (s,a) from R_table(R != -1) and remove it
10.     $Q(s,a)$ = 0.3*$R_{ca}(s,a)$ + 0.7*, $R_{na}\left(s^{'},a^{'}\right)$ + 0.8*max($Q(\tilde{s},\tilde{a})$)
11.     Q_table[s][a] = $Q(s,a)$
12.End For
13.IF epochs epoch does not reach the set value or Q table does not converge
14.     repeat 4-11
15.End If
```

4.3 Bidirectional location *k*-disturbance algorithm

4.3.1 k-disturbance for user's location algorithm

Assuming that the user has made three service requests in the "Supermarket" and the user sets k = 3, if the intermediate server randomly selects two different types of location semantics each time the user requesting the service, the situation in Fig. 6 may occur. Because the user's identifier is unique, the location server can determine the user's query content, the first query "Fruit shop, Restaurant, Supermarket," the second query "Restaurant, Hotel, Supermarketuery," the third query "Supermarket, Café, Hardware storey". "Supermarket" exists in all three queries, so the attacker can infer that the user's location is "Supermarket," which causes privacy disclosure. In order to solve the problem of continuous query, this paper proposes a *k-disturbance for user's location algorithm*. The content of the method is that if the user does not change the location semantics of the continuous query in the client, the k class location

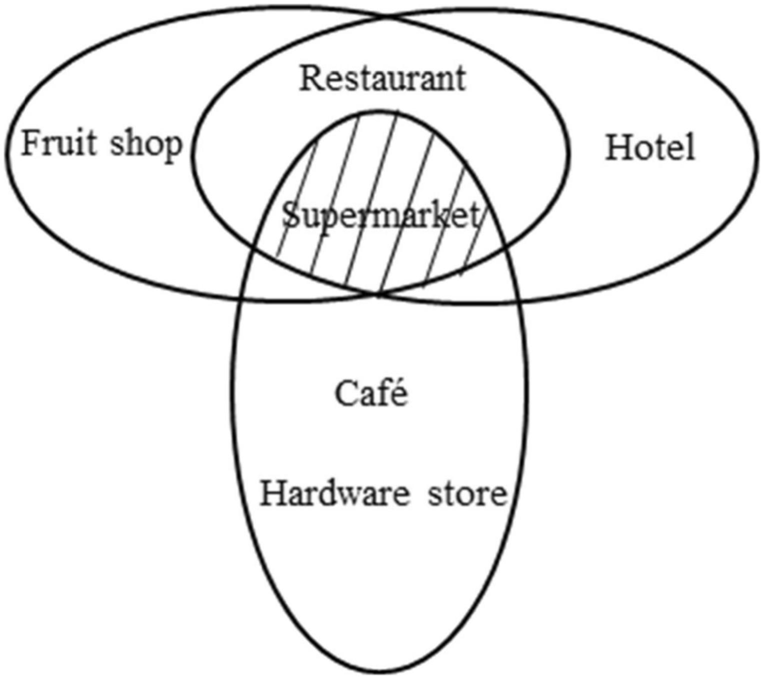


Fig. 6 Shows the possible intersection after the next query, each ellipse represents a service request, and the shaded part represents the intersection of three service requests

semantics of the first location *k*-anonymity is still used, but if the location semantics is changed into new, a new *k* class location semantics is selected to do the location *k*-anonymity. For example, set *k* = 3.The first case: Bob searches nearby "Restaurant" in the "Supermarket," so the client chooses "Supermarket," Bob searches nearby "Restaurant's" location *k*-anonymity; the second case: if Bob does not move to another location semantics range and searches for the nearby "Bar," the *k*-anonymous result is still "Supermarket, Café, Hardware store"; the third case: if Bob moves to "Hotel," *k*-anonymous results will change, and the client will choose "Hotel, Café, Supermarket" to do the user's location *k*-anonymity. Using *k-disturbance for user's location algorithm* can solve the intersection problem in continuous queries.

4.3.2 k-disturbance for user's query objectives algorithm

This paper proposes a *k-disturbance for user's query objectives algorithm.* The content of the algorithm is that the client randomly selects *k*-1 class location semantics and the location semantics of user queries from user's location road or adjacent sections to meet the *k*-anonymous request. *k-disturbance for user's query objectives algorithm* also needs to solve the intersection problem in the continuous query. If the query location semantics of the client does not change in a continuous query, the *k* class location semantics of the first query target location *k*-anonymity is still used; but if the query location semantics is changed into new, a new *k* class query targets' location semantics is selected to do the location *k*-anonymity. When the user's location semantics change, a new *k* class query targets' location semantics also needs to be re-selected to do the location *k*-anonymity. Because if the user location changes, the user may not be in the original section or the original k-1 location semantics used to meet *k*-anonymous requirements may not be in the user's section and adjacent sections. For example, set *k* = 3, the first case: Bob searches nearby "Bank" in the "Supermarket," so "Bank" will be changed into *k*-anonymous "Bank, Bar, Hotel" after *k*-anonymity; the second case: if the user still requests for nearby "Bank" query services in the "Supermarket," *k*-anonymous result is still "Bank, Bar, Hotel"; the third case: if Bob in the "Supermarket" requests for the nearby "Bar" query services, two new location semantics will be re-selected to meet *k*-anonymous requirements, and the possible results can be "Bar, Hotel, Pastry shop".

The set of road segments used by *k-disturbance for user's location algorithm* or *k- disturbance for user's query objectives algorithm* when anonymizing the user's location is provided by *cooperative segment matching algorithm*. We combines *k-disturbance for user's location algorithm* and *k-disturbance for user's query objectives algorithm* comprehensively as *Bidirectional location k-disturbance algorithm*, and *Bidirectional location k-disturbance algorithm* is as follows:

Algorithm 3. BL*k*-anonymity(*Bidirectional location k- disturbance*) Algorithm

```
Input: Privacy protection demand $PPD(SEN_u,k)$, user's location $\mathbf{Upos}_i(rid, id, x, y, tp)$, user queries the location semantics of the target qtp
Output: The user's k query launch location semantics and k query target location semantics
1. Based on rid obtain $\mathbf{road}_i(rid, Set_{adj}, TP_i)$
2. If k<len($TP_i$) then
3.     Randomly select k-1 class tp from $TP_i$
4. end If
5. If k> len($TP_i$) then
6.     Randomly select a $Cr_i$ from $Cr_{set}$
7.     k= len($TP_i$+$Cr_i$)
8.     $TP$=($TP_i$+$Cr_i$)
9. End If
10. Repeat 5-9 Until len($TP_i$)>=k
11. Randomly select k-1 class tp from $TP$(Remove duplicate tp)
12. Return k query launch location semantics
13. If the user makes multiple queries
14.     Decide whether to regenerate k query launch location semantics depending on whether the user moves to another location semantics
15.     if need to regenerate k query launch location semantics
16.         regenerate k query launch location semantics
17.     else Perform step 12
18.     end if
19. Else Perform step 12
20. End If
21. based on user queries the location semantics of the target qtp
22. Traverse the user's section and the tp in the adjacent section notes on $TP_{target}$
23. Randomly select k-1 class tp from $TP_{target}$
24. Return k query target location semantics
25. If the user makes multiple queries
26.     Decide whether to regenerate k query launch location semantics depending on whether the user moves to another location semantics or change query target
27.     if need to regenerate k query target location semantics
28.         regenerate k query target location semantics
29.     else Perform step 24
30.     end if
31. Else Perform step 24
32. End If     33. Return   k query launch location semantics, k query target location semantics
```

5 Experiment and results

5.1 Experimental setting

In this paper, the road network data in Oldenburg, Germany [47] includes a total of 6105 roads, which are composed of 7035 vertices. The vertical and horizontal coordinates of these vertices are between 0 and 30,000 and between 0 and 25,000, respectively. The road network generated from the roads and vertices in the original data is shown in Fig. 7. The research content of this paper is mainly based on the location semantic information in the road network, which needs to generate the location semantic information according to the original data set. The experimental setting randomly generates 3 to 10 kinds of location semantic information on each road in the original road network and generates a road network with semantic information as shown in Fig. 8. As shown in Fig. 9, The generator generates 10 different location semantics that randomly are distributed in the original road network.

The experiments in this paper include the required time to anonymize of the proposed BL*k*-disturbance algorithm, the needed roads to complete anonymity according to the size of user-selected *k* of BL*k*-disturbance algorithm, anonymous success rate of anonymous method based on location semantics compared with random dummy location generation, and the quality of service comparison between BL*k*-disturbance algorithm and random dummy position generation algorithm. The experimental code for this article is written in Python and runs on Windows10 operating system configured with Intel (R) Core i5-4590 CPU, 8GB, 64-bit.

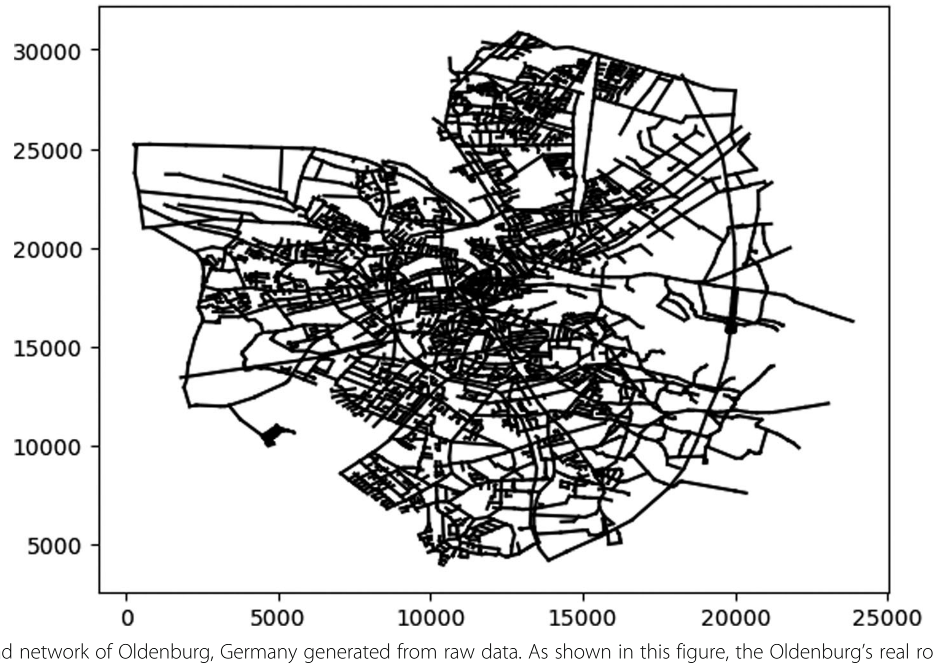


Fig. 7 The city road network of Oldenburg, Germany generated from raw data. As shown in this figure, the Oldenburg's real road network generated from the roads and vertices in the original data

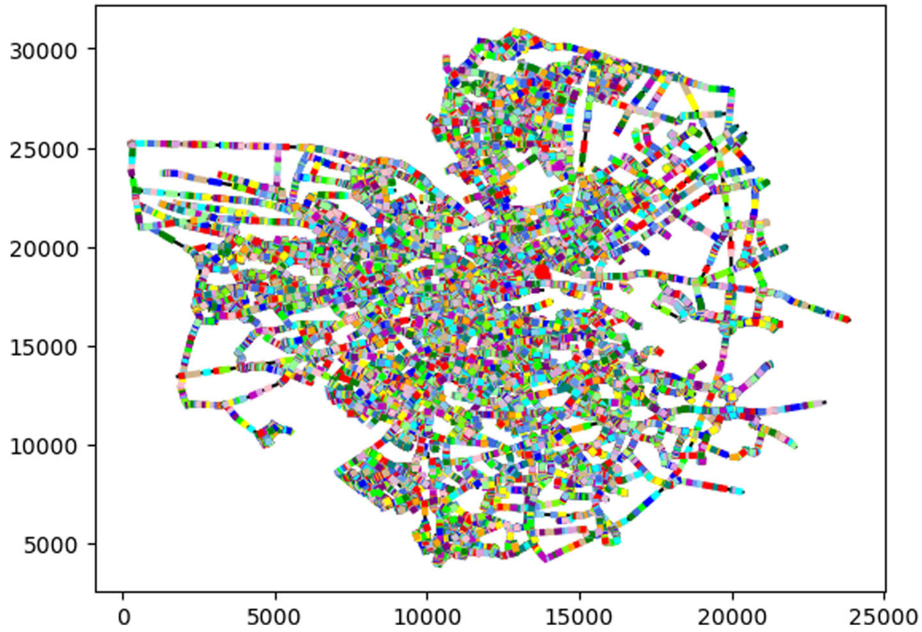


Fig. 8 Road network with semantics information. As shown in this figure, the experimental setting randomly generates 3 to 10 kinds of location semantic information on each road in the original road network and generates a road network with semantic information

5.2 The required time to anonymize of BL*k*-disturbance algorithm

Bidirectional location k-disturbance includes two parts: *k*-disturbance algorithm for the user's location and *k*-disturbance for query targets, where the time spent on the user's location using the user's location *k*-disturbance algorithm is shown in Fig. 10a. The first, second, and third cases correspond to the user location status when three kinds of users of the user's location *k*-disturbance algorithm inquiry in chapter 4.2. In order to compare the time consumption corresponding to different *k* values selected by the user in three cases, 10,000

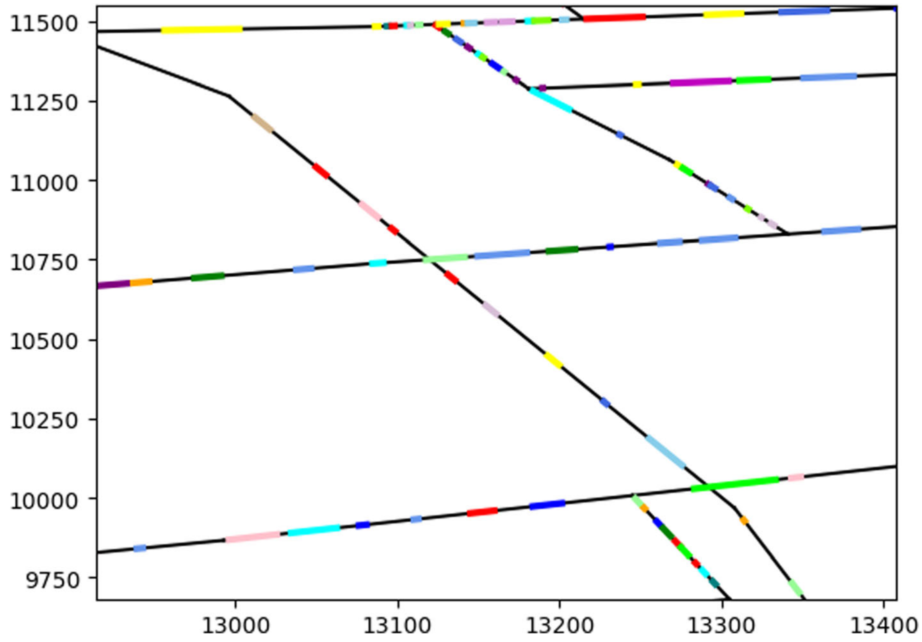


Fig. 9 Location semantics information distribution on the part of roads with semantics information. As shown in this figure, the generator generates 10 different location semantics, which randomly are distributed in the original road network

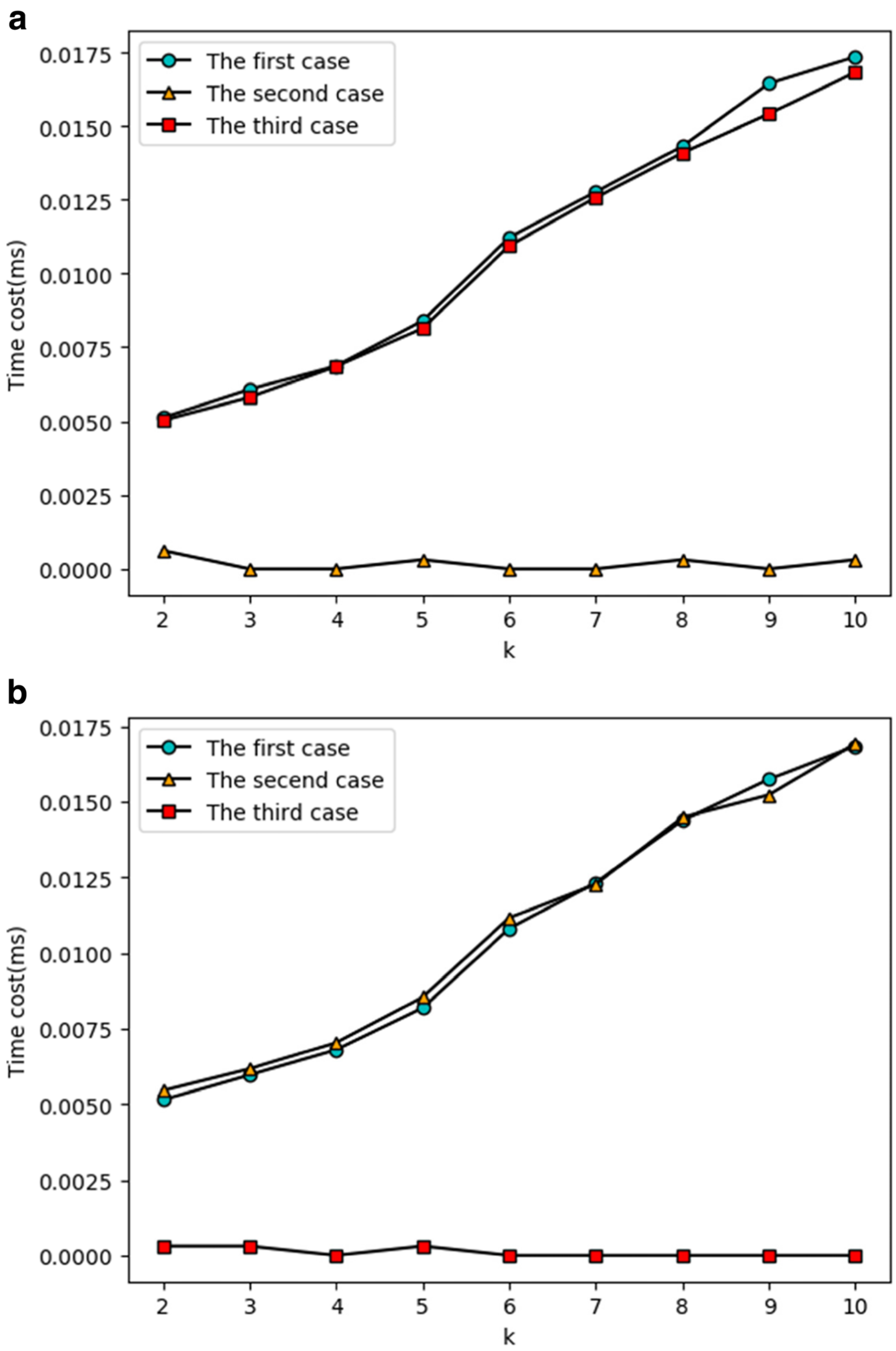


Fig 10 a The time spent on user's location using the user's location *k*-disturbance algorithm. As shown in **a**, line segments with blue solid circles, yellow triangle, and red square represent the time consumed of the first, second, and third query for different *k*-anonymous requirements, respectively. The first, second, and third cases correspond to the user location status when three kinds of users of the user's location k disturbance algorithm inquiry in chapter 4.2. **b** The time spent on user's query target anonymity using the user's query target *k*-disturbance algorithm. As shown in **b**, line segments with blue solid circles, yellow triangle, and red square represent the time consumed of the first, second, and third queries for different *k*-anonymous requirements, respectively. The first, second, and third cases correspond to the three cases of the user's query target *k*-disturbance algorithm in chapter 4.2

experiments were carried out in each of the three cases, and the time consumed in each case was the average of 10,000 experimental results. Anonymous time of the first and third cases is more similar, because the first and third cases need to generate *k* dummy positions and use the same mechanism, and the time spent was only linked with the size of *k*. However, in the second case, only the dummy position generated in the first case is reused, so the time consumption is almost zero.

The time spent on user's query target anonymity using the user's query target *k*-disturbance algorithm is shown in Fig. 10b. Similarly, with the experiment above, the first, second, and third cases correspond to the three cases of the user's query target *k*-disturbance algorithm in chapter 4.2. The time consumption of the first and second cases is similar because the first and second cases need to generate *k* dummy positions and use the same mechanism, and the time consumption was only linked with the size of *k*. However, in the third case, only the dummy position generated in the first case is reused, so the time consumption is almost zero.

5.3 The needed roads to complete anonymity of BL*k*-disturbance algorithm

In the experiment, 2 to 10 kinds of location semantic information are randomly distributed on each road. The number of roads needed to complete anonymity varies with the *k* value. As shown in Fig. 11, the matchbox bar represents the statistical value of the number of roads that the user needs to complete anonymity under the different *k*-value requirements. As the value of *k* increases, the number of roads required to complete anonymity increases. Because the larger the value of *k*, the more semantic information on the road is needed. However, the semantic distribution of the location of each road does not necessarily meet the user's demand for *k*, so cooperation with a number of roads is needed to complete the anonymity of user's location and query target. The solid black line in Fig. 11 represents the number of roads to anonymize for different values of *k*, which is obtained by rounding up the statistics number of required road and more intuitively represents the relationship between *k* and anonymity required the number of required roads to anonymize in real meaning.

5.4 Anonymous success rate of anonymous method based on location semantics compared with random dummy location generation

Anonymous success rate can evaluate an anonymous algorithm. In the real road network, each location semantic information has the geographic location range. However, the anonymous algorithm based on random dummy location generation does not consider the

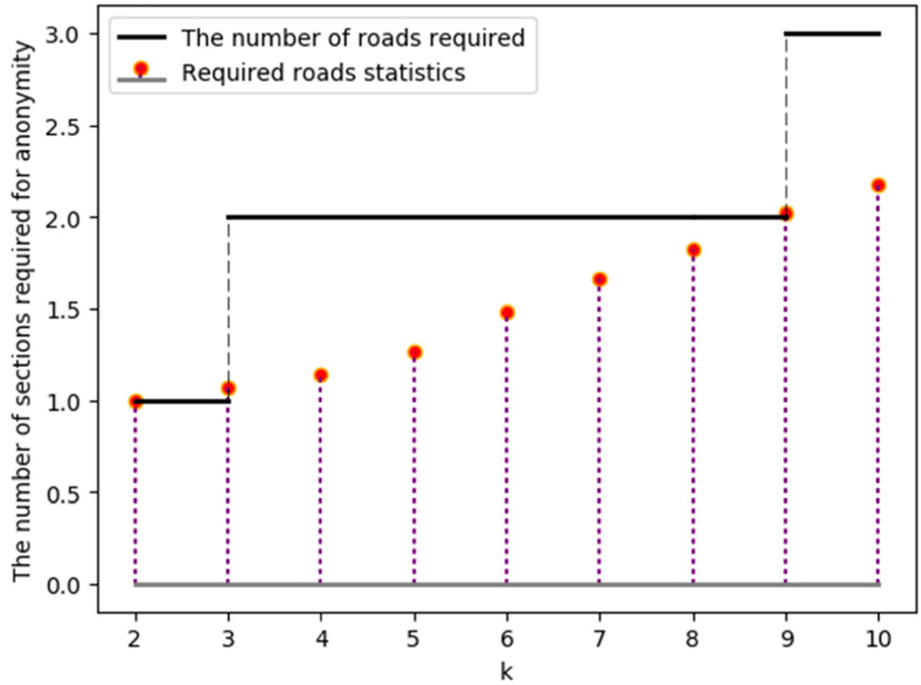


Fig. 11 The needed roads to complete anonymity of BL*k*-disturbance algorithm. As shown in this figure, the matchbox bar represents the statistical value of the number of roads that the user needs to complete anonymity under the different *k*-value requirements

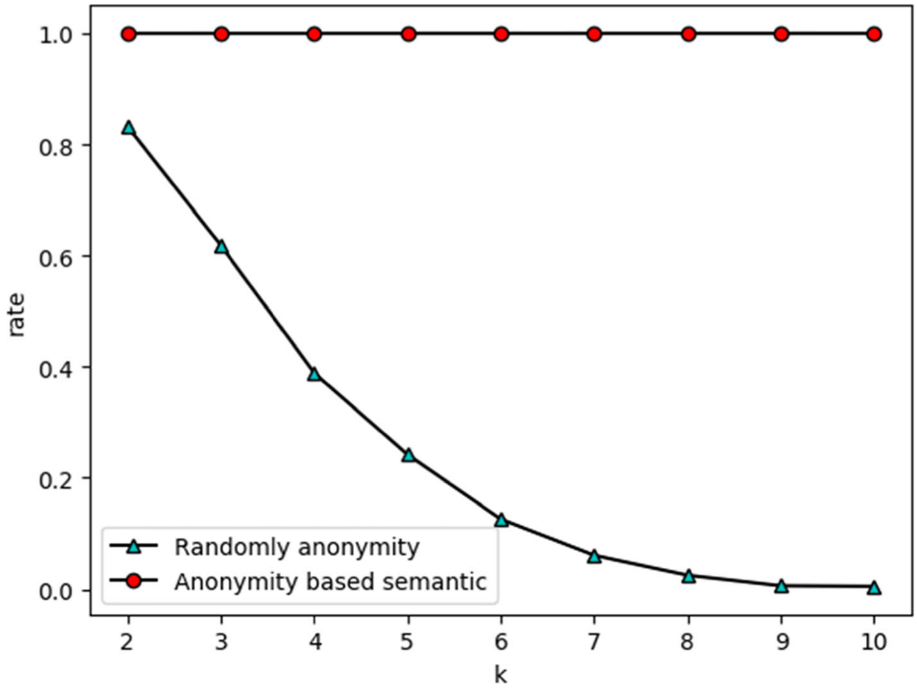


Fig. 12 Anonymous success rate of anonymous method based on location semantics and random dummy location generation. As shown in this figure, red solid circles represent semantic-based anonymity and a blue triangle represents random anonymity. The *x*- and *y*-axes represent the size of *k* and the anonymous success rate, respectively

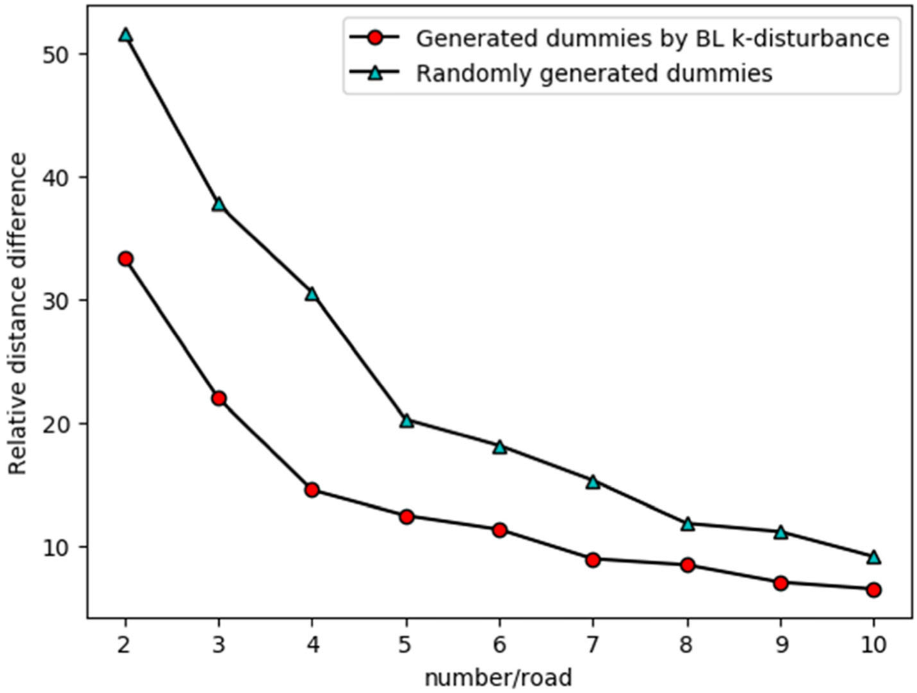


Fig. 13 The service quality of BL*k*-disturbance algorithm and the random false location generation algorithm. As shown in this figure, red solid circles represent generated dummies by BL*k*-disturbance, and blue triangle represent randomly generated dummies. The *x*- and *y*-axes represent the number of location semantic categories on each road and the relative distance difference, respectively

geographic location range, and k dummy positions are generated randomly in a given anonymous area. And if used in road network, two or more dummy locations will distribute within the geographic range of the same location semantics. If an attacker attacks based on location semantics information, user's k-anonymous requirement cannot be met, or anonymity fails. As shown in Fig. 12, the proposed method based on location semantic information anonymity has a 100% success rate, but the anonymity success rate of anonymous method based on random dummy location generation decreases with the increase of k value. When k is set as 9 or 10, the success rate is essentially zero, which greatly increases the probability of an attacker's successful attack.

5.5 The quality of service comparison between BL*k*-disturbance algorithm and random dummy position generation algorithm

In order to compare the quality of service finally obtained by the BLk-disturbance algorithm and the random dummy position generation algorithm, we set up a comparative experiment. Regardless of the final selection of several roads to complete the anonymity, the final calculation of the distance from the user to the target is only related to the location of the user and has nothing to do with the other sections that jointly accomplish the k-disturbance. In order to evaluate the quality of service that the algorithm ultimately enables the user to obtain, 10,000 experiments were conducted in which the quality of service was replaced by the relative distance of the user-selected target. The relative distance is the distance between the dummy location and the user's location. The smaller the relative distance is, the better the service quality is. As shown in Fig. 13, the more semantic information of the location there are on the road where the user is located, the smaller the relative distance difference is and the more accurate the quality of service obtained is. As can be seen in Fig. 13, the quality of service obtained by the random dummy position generation algorithm is inversely proportional to k, but the quality of service using dummy position generated by BLk-disturbance is superior to the random dummy position generation algorithm when the user selects the same k value.

6 Discussion

The above analysis is based on location privacy protection in areas with rich location semantic information. In fact, if the location semantic information is sparse, it is not easy to reach the requirement of user k-anonymous, and the purpose of location privacy protection is also not achieved. The research on location privacy protection in areas with sparse location semantic information is one of the directions that are worthy of further research.

7 Conclusion

In this paper, we propose a personalized location privacy protection framework based on location semantics. We first obtain the user's sensitivity preference of various location semantics without trivial manual settings. In order to provide better privacy protection and service quality for users, we then propose cooperative road sections matching the algorithm which uses the idea of reinforcement learning and short-time learning, which can get the cooperative road sections for k-anonymity based on user's sensitivity preference. Then we propose a bidirectional location k-disturbance algorithm, which disturb the user's location and the query location by using the candidate set of road segments. In the future work, we will consider the issue of location privacy protection for more complex inbound queries in bidirectional road networks.

Abbreviation

FR: Feasible domain; GPS: Global Positioning System; LBS: Location-based services; PDD: Privacy protection demand; TP: All types of position semantics

Authors' contributions

LK, YW, and YS conceived the main idea and contributed to the writing and the revisions. YW and XZ conducted the experiments. All authors read and approved the final manuscript.

Funding

The authors acknowledge the National Natural Science Foundation of China (No.61772560), National Key R&D program of China (No.2018YFB1003800), and Natural Science Foundation of Hunan Province (No. 2019JJ40388).

Availability of data and materials

The datasets used or analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Received: 18 May 2019 Accepted: 5 December 2019
Published online: 14 January 2020

References

1. M. Gruteser, D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking[C]//Proceedings of the 1st international conference on Mobile systems, applications and services. ACM, 31–42 (2003)
2. Yiu M L, Jensen C S, Huang X, et al. Spacetwist: managing the trade-offs among location privacy, query performance, and query accuracy in mobile services[C]//Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on. IEEE, 2008: 366-375.
3. M.F. Mokbel, C.Y. Chow, W.G. Aref, The new casper: query processing for location services without compromising privacy[C]//Proceedings of the 32nd international conference on Very large data bases. VLDB Endowment, 763–774 (2006)
4. Z. Gong, G.Z. Sun, X. Xie, Protecting privacy in location-based services using k-anonymity without cloaked region[C]// Eleventh International Conference on Mobile Data Management. IEEE Comput Society, 366–371 (2010)
5. A. Khoshgozaran, C. Shahabi, H. Shirani-Mehr, Location privacy: going beyond K-anonymity, cloaking and anonymizers[J]. Knowledge Inform Syst **26**(3), 435–465 (2011)
6. R. Shokri, G. Theodorakopoulos, P. Papadimitratos, et al., Hiding in the mobile crowd: Locationprivacy through collaboration[J]. IEEE Trans Depend Secure Comput **11**(3), 266–279 (2014)
7. Gedik B, Liu L. A customizable k-anonymity model for protecting location privacy[R]. Georgia Institute of Technology, 2004.

8. Zhu X, Chi H, Niu B, et al. Mobicache: When k-anonymity meets cache[C]//Global Communications Conference (GLOBECOM), 2013 IEEE. IEEE, 2013: 820-825.
9. L. Kuang, Y. Zhu, S. Li, et al., A privacy protection model of data publication based on game theory[J]. Secur Commun Netw **2018**, 1–13 (2018)
10. C.Y. Chow, M.F. Mokbel, W.G. Aref, Casper*: Query processing for location services without compromising privacy[J]. ACM Trans Database Syst (TODS) **34**(4), 24 (2009)
11. G. Sun, D. Liao, H. Li, et al., L2P2: A location-label based approach for privacy preserving in LBS[J]. Future Generation Comput Syst (2016)
12. H. Jadallah, A.Z. Al, Aman: spatial cloaking for privacy-aware location-based queries in the cloud[C]//Proceedings of the International Conference on Internet of things and Cloud Computing. ACM **60** (2016)
13. Tai F Y, Song J K, Tsai Y C, et al. Cloaking sensitive patterns Td preserve location privacy for LBS applications[C]//Consumer Electronics-Taiwan (ICCE-TW), 2016 IEEE International Conference on. IEEE, 2016: 1-2.
14. M. Li, Z. Qin, C. Wang, Sensitive semantics-aware personality cloaking on road-network environment[J]. Int J Sec Appl **8**(1), 133–146 (2014)
15. Huang Y, Huo Z, Meng X F. Coprivacy: a collaborative location privacy-preserving method without cloaking region[J]. Jisuanji Xuebao (Chinese Journal of Computers), 2011, 34(10): 1976-1985.
16. C. Li, B. Palanisamy, *De-anonymizable location cloaking for privacy-controlled mobile systems[C]//International Conference on Network and System Security* (Springer, Cham, 2015), pp. 449–458
17. Kuang L, Wang Y, Ma P, et al. An improved privacy-preserving framework for location-based services based on double cloaking regions with supplementary information Constraints[J]. 2017, 2017:1-15.
18. Kido H, Yanagisawa Y, Satoh T. An anonymous communication technique using dummies for location-based services[C]//Pervasive Services, 2005. ICPS'05. Proceedings. International Conference on. IEEE, 2005: 88-97.
19. Kido H, Yanagisawa Y, Satoh T. Protection of location privacy using dummies for location-based services[C]//Data Engineering Workshops, 2005. 21st International Conference on. IEEE, 2005: 1248-1248.
20. Guo M, Pissinou N, Iyengar S S. Pseudonym-based anonymity zone generation for mobile service with strong adversary model[C]//Consumer Communications and Networking Conference (CCNC), 2015 12th Annual IEEE. IEEE, 2015: 335-340.
21. B. Palanisamy, L. Liu, Attack-resilient mix-zones over road networks: architecture and algorithms[J]. IEEE Trans Mobile Comput **14**(3), 495–508 (2015)
22. Niu B, Zhang Z, Li X, et al. Privacy-area aware dummy generation algorithms for location-based services[C]//Communications (ICC), 2014 IEEE International Conference on. IEEE, 2014: 957-962.
23. A. Khoshgozaran, C. Shahabi, H. Shirani-Mehr, Location privacy: going beyond K-anonymity, cloaking and anonymizers[J]. Knowledge and Information Systems **26**(3), 435–465 (2011)
24. S. Papadopoulos, S. Bakiras, D. Papadias, Nearest neighbor search with strong location privacy[J]. Proc VLDB Endow **3**(1-2), 619–629 (2010)
25. Lu R, Lin X, Shi Z, et al. PLAM: A privacy-preserving framework for local-area mobile social networks[C]//INFOCOM, 2014 Proceedings IEEE. IEEE, 2014: 763-771.
26. K. Mouratidis, M.L. Yiu, Shortest path computation with no information leakage[J]. Proc VLDB Endowment **5**(8), 692–703 (2012)
27. A. Khoshgozaran, C. Shahabi, Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy[J]. Adv Spat Temporal Databases, 239–257 (2007)
28. A. Khoshgozaran, C. Shahabi, Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy[J]. Adv Spat Temporal Database, 239–257 (2007)
29. R. Paulet, M.G. Kaosar, X. Yi, et al., Privacy-preserving and content-protecting location based queries[J]. IEEE Trans Knowledge Data Eng **26**(5), 1200–1210 (2014)
30. M.L. Damiani, C. Silvestri, E. Bertino, Fine-grained cloaking of sensitive positions in location-sharing applications[J]. IEEE Pervasive Comput **10**(4), 64–72 (2011)
31. Z. Xiao, J. Xu, X. Meng, p-Sensitivity: A semantic privacy-protection model for location-based services[C]// International Conference on Mobile Data Management Workshops. IEEE Comput Soc, 47–54 (2008)
32. M.L. Damiani, E. Bertino, C. Silvestri, The PROBE framework for the personalized cloaking of private locations[J]. Trans Data Privacy **3**(2), 123–148 (2010)
33. M.L. Damiani, C. Silvestri, E. Bertino, Analyzing semantic locations cloaking techniques in a probabilistic grid-based map[C]// Sigspatial International Conference on Advances in Geographic Information Systems. ACM, 522–523 (2010)
34. Xue M, Kalnis P, Pung H K. Location Diversity: Enhanced Privacy Protection in Location Based Services[C]// International Symposium on Location and Context Awareness. Springer-Verlag, 2009:70-87.
35. Pan X, Wu L, Piao C, et al. P3RN:Personalized Privacy Protection Using Query Semantics over Road Networks[C]// International Conference on Web-Age Information Management. Springer, Cham, 2014:323-335.
36. L. Qi, X. Zhang, W. Dou, Q. Ni, A Distributed Locality-Sensitive Hashing based Approach for Cloud Service Recommendation from Multi-Source Data. IEEE J Selected Areas Commun **35**(11), 2616–2624 (2017)
37. Yanwei Xu, Lianyong Qi, Wanchun Dou, Jiguo Yu. Privacy-preserving and scalable service recommendation based on SimHash in a distributed cloud environment. Complexity, Volume 2017, Article ID 3437854, 9 pages, 2017.
38. Lianyong Qi, Ruili Wang, Shancang Li, Qiang He, Xiaolong Xu, Chunhua Hu. Time-aware distributed service recommendation with privacy-preservation. Inform Sci, 480: 354-364, 2019.
39. Lianyong Qi, Yi Chen, Yuan Yuan, Shucun Fu, Xuyun Zhang, Xiaolong Xu. A QoS-aware virtual machine scheduling method for energy conservation in cloud-based cyber-physical systems. World Wide Web Journal, 2019.
40. Wenwen Gong, Lianyong Qi, Yanwei Xu. Privacy-aware multidimensional mobile service quality prediction and recommendation in distributed fog environment. wireless communications and mobile computing, vol. 2018, Article ID 3075849, 8 pages, 2018.
41. L. Qi, X. Zhang, W. Dou, C. Hu, C. Yang, J. Chen, A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment. Future Gen Comput Syst **88**, 636–643 (2018)
42. Z. Wu, Z. Lu, C. Patrick, K. Hung, S.-C. Huang, Y. Tong, Z. Wang, QaMeC: A QoS-driven IoVs application optimizing deployment scheme in multimedia edge clouds. Future Gen Comput Syst **92**, 17–28 (2019)
43. B. Lee, J. Oh, H. Yu, et al., Protecting location privacy using location semantics[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, Ca, Usa, August. DBLP, 1289–1297 (2011)
44. P.L. Kaelbling, A situated-automata approach to the design of embedded agents[J]. ACM SIGART Bulletin **2**(4), 85–88 (1991)
45. C.J.C.H. Watkins, P. Dayan, Q-learning[J]. Mach Learn **8**(3-4), 279–292 (1992)
46. Tsitsiklis J N . Asynchronous stochastic approximation and Q-learning[J]. Mach Learn, 1994, 16(3):185-202.
47. S. Chen, C.S. Jensen, D. Lin, A benchmark for evaluating moving object indexes.[J]. Proc Vldb Endowment **1**(1), 1574–1585 (2008)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com