SOKA：An Optimized K-Anonymity Algorithm in Location-based Services

Zinan Ding and Xuebin Ma

***Abstract*—In recent years, the demand for location-based services (LBS) has grown explosively with the development of wireless networks and mobile services. It is of paramount importance to protect location privacy which may disclose users' private information. Previous studies have focused on location obfuscation-based scheme such as perturbing and location spoofing. However, these methods reduce data availability and increase time overhead. In this paper, we apply the self-organizing map network in LBS for the first time to transform the high-dimensional data into two-dimensional data while preserving the topology of the original data. In addition, we propose a constraint-based k-anonymity region construction algorithm which considering l-diversity and t-closeness. For cloaking regions that do not satisfy the conditions, a cloaking region adjustment strategy is used to add a varying number of optimal dummy locations based on location semantics, making it difficult for attackers to distinguish them. Experimental results show that our approach is able to reduce the information loss while significantly reducing the running time, achieving a balance between privacy-preserving capability and data utility.**

***Index Terms*—K-anonymity, location-based services, location privacy, self-organizing map, cloaking regions.**

I. INTRODUCTION

With the popularity of wireless communication and mobile network technologies, more and more mobile users are requesting location-based services (LBS) by sending queries to LBS service providers (LSPs), LBS refers to various information services provided to mobile users based on the geographic location information provided by GPS [1],[2].

However, while enjoying the convenience of LBS, users must provide LBS servers with the location information involved in LBS queries, so LBS service providers may intentionally or inadvertently leak the location information involved in queries, based on which attackers can easily infer not only the user's trajectories but also the user's sensitive preferences (e.g., sensitive interest points). In addition to leaking these kinds of information to untrustworthy third parties, there is also a risk of interception by attackers when users send location query requests to LSPs in transit, which will undoubtedly trigger a large-scale privacy breach.

To solve the above privacy problems, many approaches have been proposed in the past few years, mainly divided into location perturbation and location obfuscation. Perturbation-based approaches typically use differential privacy, which defines a rigorous adversary model and provides a strict proof of privacy preservation. However, the added noise affects the accuracy of the POI service and does not work well in the dataset for user private value correlation like location data. Obfuscation-based approaches often use well-known privacy metrics such as k-anonymity. In order to achieve k-anonymity, an LBS query needs to be forwarded to the LBS provider via an anonymizer, which expands the query location to a larger anonymized region covering more (k-1) users. As a result, it is difficult for an untrusted LSP to distinguish the real location of a user from the virtual locations. However, although false locations can reach k-anonymity, how to select false locations is a challenge. Most existing approaches assume that the team hand cannot have the background knowledge, such as the user's social status or personal characteristics, and thus use a random walk model or a virtual circle or grid model. However, unrealistic false locations such as those distributed in lakes or mountains can be easily filtered by the attacker. Therefore, it is difficult to guarantee desired k-anonymity. Besides, generating a large number of virtual locations and a large range of regions requires huge time and space overhead.

In this paper, we propose an optimized k-anonymity algorithm called SOKA, which consists of a self-organizing map network model and an optimized k-anonymity model as a way to improve the utility and privacy level. First, our proposed approach uses greedy and approximate strategies to generate efficient steganographic regions by setting some constraints such as l-diversity, t-closeness, and the number of fake locations in the region varies according to the number of real locations in the region, which can reduce the overhead and time complexity of the system. Second, to generate dummy locations that match the user's behavioral characteristics, we build a semantic generalization tree of locations for interest points and preferences around the user to quantify the semantic distance between real and fake locations, and filter the best set of fake locations to fill the hidden regions. The generated false location set is more confusing because it combines realistic geography, spatio-temporal correlation and other semantic factors. Experimental results on the real trajectory dataset confirm that our algorithm outperforms other existing algorithms.

The main contributions of this paper are summarized as follows.

1) We deploy the self-organizing map into the LBS to preprocess the input data to reduce the complexity of high-

ZINAN DING and XUEBIN MA are with the Department of Inner Mongolia Laboratory of Wireless Networking and Mobile Computing, Inner Mongolia University, Hohhot, 010000, China (e-mail: 32009139@mail.imu.edu.cn; csmaxuebin@imu.edu.cn)

dimensional data while maintaining the topological features of the original data.

2) In order to better balance the privacy and data utility, we design an optimized k-anonymity region formation algorithm and combine it with self-organizing map for the first time, which can effectively reduce time consumption.

3) We propose a location-semantic-aware dummy locations generation algorithm that generates different numbers of dummy locations while considering location context information.

The rest of this paper is organized as follows: Section 2 discusses related works; some related background knowledge and k-anonymity related definitions are introduced in Section 3; In Section 4, we propose a new optimized k-anonymity region construction algorithm SOKA; The experimental results are reported in Section 5; Section 6 concludes the paper.

II. Related Work

A. K-anonymity

The k-anonymity model was proposed by Sweeny [3] in 2002, which is a common model used to solve the privacy problem of big data. And this technique ensures that individual records of each sensitive attribute stored in the published dataset cannot be distinguished from other k-1 individuals, so that the probability of each user being identified is less than or equal to 1/k. At this point, an observer cannot link records by quasi-identifiers to link records.

K-anonymity is widely used in social networking, medical, Internet of Things, smart transportation and other fields. To achieve anonymization of social network graphs, Navid Yazdanjue et al. [4] optimize the clustering process in the k-anonymity method by particle swarm optimization (PSO) algorithm and combine it with GA algorithm to minimize the normalized structural information loss, achieving a balance between normalized structural information loss (NSIL) and convergence rate. Wang et al. [5] presents a privacy notion of client-based personalized k-anonymity (CPkA) for autonomous vehicles querying services in Cyber-physical systems, which allows users to specify different anonymity level of each query content. Rohan Iyer et al. [6] propose applications of spatial k-anonymity in the data sharing and managing of COVID-19 contact tracing technologies as well as heat maps showing a user's travel history.

Location k-anonymity, first proposed by Gruteser et al. [7], is a common approach to privacy preservation in LBS which is applied in the scenario that the location data of each user contained in an anonymous set cannot be distinguished from the location data of at least k-1 other users in that set. The combination of steganography and k-anonymity is usually achieved by forming a spatial region covering the anonymized set of the target user. Since then, most steganography-based techniques are generally k-anonymous and many schemes for forming anonymity regions have been proposed [8], [9]-[10].

B. Location Privacy Protection Mechanism

In location privacy protection mechanisms, architecture and technology are two important aspects. Two architectures have been used to design location privacy mechanisms, namely, P2P (peer-to-peer) architecture and TTP (trusted third party) based architecture. P2P architecture consists of only mobile users and LSPs, and this system architecture requires mobile devices with specific computational and storage capabilities. For example, Yang et al. [11] address the need to protect location privacy under different network conditions and node requirements by incorporating differential privacy in mobile opportunity networks, but this affects service accuracy. Nisha et al. [12] use a novel virtual location generation technique and virtual identity mechanism to create one-time spatial groups with nearby neighbors to reduce interaction with untrusted LSPs, but can significantly increase the system overhead. Gursoy et al. [13] propose DP-star, which constructs a density-aware grid that guarantees increased noise while still satisfying differential privacy. Niu et al. [14] use geographic indistinguishability and K-anonymity to obfuscate locations, which effectively perturbs the location distribution and suppresses privacy leakage under long-term observation attacks, but is vulnerable to multi-query attacks. Ren et al [15] propose a privacy-defining DistPreserv that enables LBS servers to obtain an efficient location distribution while providing users with strict location protection, but faces the risk of message interception. Alotaibi et al. [16] proposed an attacker location exclusion (ALE) algorithm and a new metric location privacy level (LPL) to qualify the ability of malicious LBS servers to reduce the privacy level of requesters. Qiu et al. [17] describe the user's role in location by constructing a hierarchical semantic tree based on the user's role in the mobile semantic location set, this paper proposes a more efficient semantic tree and makes a security proof.

In trusted third-party based architectures, one or more entities are usually incorporated to deploy location privacy algorithms. Exploring k-anonymity based anonymity techniques in further depth, Djordje Slijepčević et al. [18] conducted a systematic comparison of the impact of different k-anonymity algorithms on the results of machine learning models, but do not provide suggestions for improvement. To balance privacy security and query quality of location services, Zheng et al. [19] proposed a k-anonymous clustering algorithm with anonymous group centers instead of user location queries, but it causes higher time complexity compared to other clustering algorithms. Farough Ashkouti et al. [20] proposed a DI-Mondrian multidimensional anonymization method by better selection of cut points and creation of balanced partitions, providing less information loss. LODS [21] first selects candidate sets containing virtual locations with a low number of occurrences to achieve the preferred distribution, and it also considers historical anonymity sets, but does not take into account the semantic similarity between locations. Tu et al. [22] were the first to identify semantic attacks and proposed an algorithm that provides strong privacy protection against

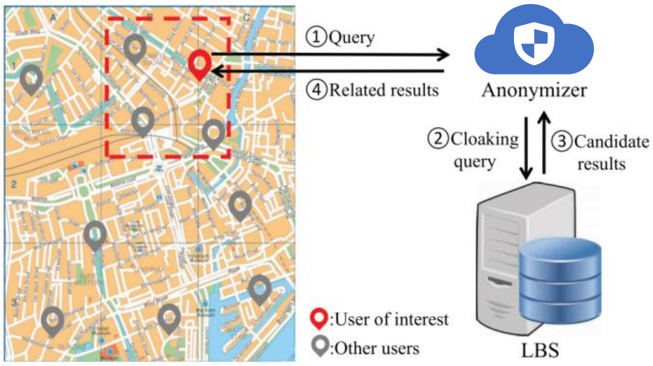


Fig.1. TTP-based architecture

semantic and re-identification attacks while retaining high data utility. Niu et al. [23] propose a hierarchical LTS model that takes into account prior knowledge and can counter global adversaries while maintaining K-anonymity. Zhang et al. [24] used Markov models and caching strategies to predict the next query location based on the user and form a spatial K-anonymity region, but required high storage and computational overhead. Siddiqie, S et al. [25] propose an improved virtual generation approach to promote location privacy for mobile users and provide personalized services to users. Li et al. [26] proposed a reversible perturbation of user's location information in spatial and temporal dimensions using a spatio-temporal steganography model with higher success rate and spatial resolution.

However, this TTP-based architecture suffers from high communication costs due to anonymous processing and transmission of location service information. Moreover, it causes a waste of resources on the LBS server due to the server answering several fake queries.

TABLE I. COMPARISON OF EXISTING STUDIES

Approach	Complexity	Architecture	Metric
SOKA	$O(N\log N)$	TTP	Semantic-similarity
Nisha, N et al. [12]	$O(N\log N)$	TTP	Euclidean Distance
Zheng et al. [19]	$O(N^2)$	P2P	Outlier point
B. Niu et al. [23]	$O(K\ log\ N)$	P2P	Entropy
Nisha, N et al. [30]	$O((R+\beta)k^2)$	CA	Scattered location
Zhao et al. [31]	$O(\sum_{i=1}^{N(u)} k_i)$	TTP	Mobility similarity
Tu et al. [32]	$O(N^2\overline{m}M)$	P2P	KL Distance

III. PRELIMINARIES

C. System Architecture

As shown in Fig.1, the trusted-third-party (TTP) based architecture consists of the following three entities:

Mobile users: The user sends the location query Q = {*ID*, *t*, *loc*, *C*} to the anonymizer, where *ID* denotes the user's identity, *t* denotes the time when the query is sent, *C* denotes the query content, and *loc* denotes the real-time location of the service requested by the user.

Anonymizer: Upon receiving location queries from mobile users, the anonymizer generates a cloaking region containing k user locations and sends it to the LSP. After receiving the location service responses from the LSP, the anonymizer forwards the query results to the mobile users.

Location Service Provider: It receives the cloaking region forwarded by the anonymizer, searches it in its internal database, gets the information that meets the user's requirements, and sends it back to the anonymizer.

D. Definitions related k-anonymity

Definition 1: (Quasi-identifiers [3]). Assume T is a table with attributes $\{A_1, A_2, \ldots, A_n\}$ and records $\{t_1, t, \ldots, t_m\}$, where each record corresponds to an individual data. A quasi-identifier of table T is a minimum set of attributes $\{A_i, \ldots, A_j\} \in \{A_1, A_2, \ldots, A_n\}$ such that $U_i^j A_x = \exists_t(T)$.

Definition 2: (K-anonymity [5]). An anonymized table T^* satisfied K-anonymity if for every tuple $t \in g_i$ there exist at least $k_i - 1$ other equivalent tuples $t' \in T^*$, where g_i is a group consisting of tuples that have the privacy constraint $k_i \in K$.

Definition 3: (L-diversity [15]). Let table T contains d quasi-identifiers attributes $\{A_1, A_2, \ldots, A_d\}$ and sensitive SA, the table T satisfies l-diversity when there are at least l distinguishable values.

Definition 4: (T-closeness [17]). An equivalence class E is said to satisfy t-closeness if the distribution of values of sensitive attributes in E does not exceed a threshold t. The table is said to satisfy t-closeness if all equivalence classes in the table satisfy t-closeness.

Definition 5: (Semantic-similarity). For any two locations l_i and l_j in the set of dummy locations, if the semantic distance between them satisfying

$$\left[d_{sem}(l_i, l_j)\right]_{min} \geq u \quad (1)$$

then the semantic discrepancy between the generated positions in the set of dummy locations satisfies the semantic discreteness.

E. Adversary model

In our trust model, anonymizers and users are considered as fully trusted entities. Moreover, the communication between anonymizers and users are considered to be secure. In our adversary model, we focus on the privacy risk of the anonymizer and the LSP, and the privacy risk of the user data

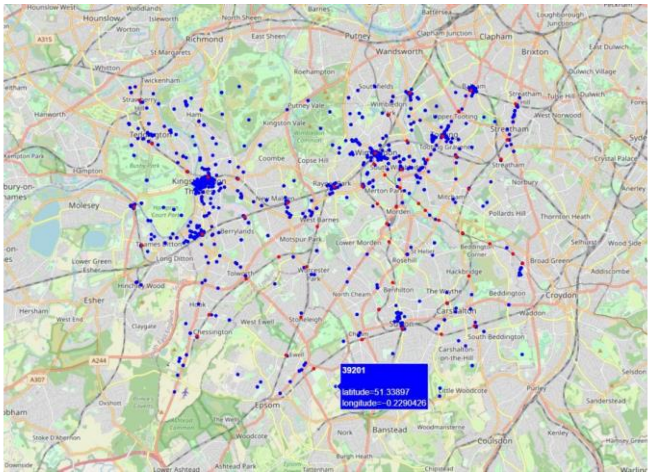


Fig.2. The locations and POIs in the real world

transmitted between the anonymizer and the LSP. Therefore, based on the attacker's knowledge, we mainly consider two attack models in our scenario.

(1) The LSP is honest and curious and manages all query data of all users. In other words, it honestly executes our proposed protocol, but may try to identify a specific user by analyzing the user's query data. At the same time, the LSP may be compromised by an adversary to obtain sensitive user information from this data.

(2) The communication channel between the anonymizer and the LSP allows the transmission of query requests and results. However, an adversary can eavesdrop on the communication channel and learn additional information related to the user. Therefore, it is possible for the adversary to obtain sensitive information from this data, such as the user's identity, sensitive location and trajectory.

F. Privacy Metric

Information loss is an important evaluation metric for measuring big data privacy protection algorithms, which can be classified as numerical and categorical based on data types.

1) **Numerical attributes**

Assume a numerical data set S = $\{A_1^n, A_2^n, \ldots, A_m^n\}$, where A_i^n denotes the set of numerical attributes in column i of the table data. For any A_i^n, assume that any attribute t_i in the record l = $(t_1, t_2, \ldots, t_m)$ in S is generalized on A_i^n to $[p_i, q_i]$, where $p_i \leq t_i \leq q_i$. Then the loss of information of record l on the numeric attribute A_i^n is defined as

$$IL_{A_i^n}(\,\mathrm{l}\,)=\frac{q_i-p_i}{A_i^n} \qquad (2)$$

Where $|A_i^n|$ is the value field of the numeric attribute A_i^n, For example, the value field of the attribute Day is $[0,30]$, assume that a record has an attribute Day of 40 and its generalized value is $[5,10]$, then the information loss after generalization of the attribute Day for that record is $(10-5)/30 = 1/6$.

2) **Categorical attributes**

Assuming that the attribute value v of record l on the categorical attribute A_j^c is generalized to w_j along with the other attribute values $v_1, v_2, \ldots, v_m$, $(1 \leq i \leq m, m+1 \leq j \leq n)$, then the information loss of record l after generalization on the categorical attribute A_j^c is defined as

$$IL_{A_j^c}(\,l\,)=\frac{Num(w_j)}{|A_j^c|} \qquad (3)$$

Where $|A_j^c|$ is the total number of categorical attributes A_j^c and $Num(w_j)$ is the number of leaf nodes of w_j.

IV. Algorithm Design

Fig.2 shows a visual map of the real trajectory dataset, where the blue points are the real locations of users when they send location queries. It is obvious that the location points are denser in urban areas and sparser near suburban areas. Therefore, the distribution of location points and location semantics are highly related. In this paper, we use a partitioning strategy of greedy sums and construct a semantic generalization tree based on location semantics and service similarity. Through the above algorithms, information loss is reduced and time and space consumption are reduced. The specific process and implementation details of SOKA algorithm are discussed in this section.

A. Algorithm Description

In this paper, we propose an optimal SOM-based k-anonymity algorithm in location-based services called SOKA. the SOKA algorithm consists of three stages: preprocessing using SOM, cloaking region construction, and cloaking region adjustment.

```
Algorithm 1: SOKA Algorithm
Input: Dataset S, Quasi-identifiers QI, Anonymity degree
       threshold k, Diversity threshold l, Distance
       threshold t, Information loss θ ,Self-organizing
       map som_size;
Output: Dataset S' consist of Cloaking Regions
$S' = \emptyset$
Coordinates (x, y) ← SOM (S, QI, som_size)
QI ← Coordinates (x, y)
Cloaking Regions ← Constructing Cloaking Region
    (S, QI, k, l, t)
$S' = S' \cup$ Cloaking Regions
return $S'$
```

The flow of the location query service is shown in Fig. 3. First, SOM effectively reduces the complexity of high-dimensional data by mapping it to a lower dimensional output space (usually 2 dimensions), and additionally SOM can extract features. We add SOM to the anonymizer, which not only reduces the time complexity, but also can have high generalization ability. In our proposed model, the SOM can not only reduce the anonymous processing time but also retain its topology to clearly distinguish different classes in the dataset.

Next, we propose the optimized k-anonymous region construction algorithm (Algorithm 2). First, while generating

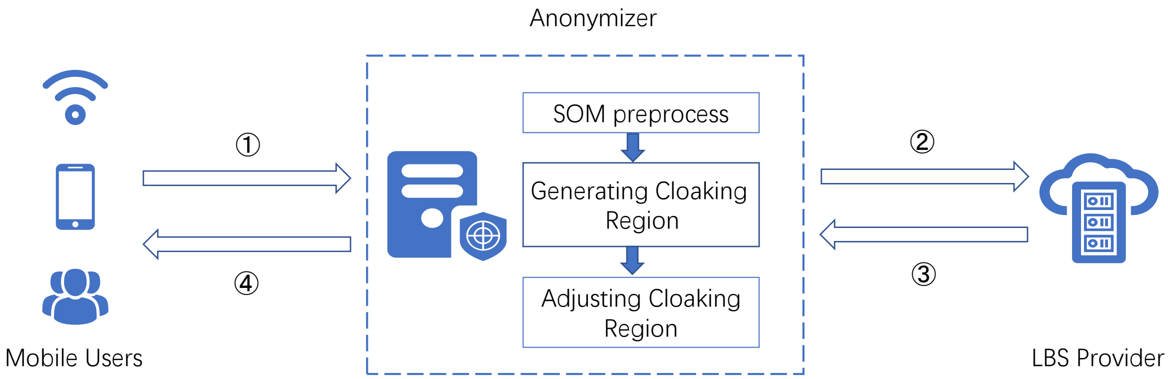


Fig.3. An overflow of location query service

the partitions, we borrow the Mondrian partitioning idea [17] to greedily partition the dataset. Moreover, we add a series of constraints, including l-diversity, t-closeness, etc., so that we can obtain higher data availability and ensure that each region has low information loss. Therefore, when constructing partitions, we use different cloaking region construction strategies according to the location semantics (Algorithm 3). To compute the semantic distance between locations, we construct the Location Semantic Tree (LST), which is a generalized hierarchical tree.

Algorithm 1 describes the general process of SOKA. In the second line, the dataset and the quasi-identifier attributes are fed into a self-organizing map network for training, and the neurons compete with each other to gradually optimize the network by a competitive learning strategy. And a nearest neighbor relationship function is used to maintain the topology of the input space. After the process, each high-dimensional vector of the input obtains one of its coordinates mapped to the two-dimensional coordinate system. In line 3, we consider the obtained two-dimensional coordinates as a "super" quasi-identifier and use it as a new set of quasi-identifiers (*QI*). Since the SOM has a high generalization capability, this two-dimensional site-standard identifier can improve the accuracy of the model and reduce the time complexity of the anonymization algorithm. In line 4, the cloaking region construction algorithm partitions the locations scattered on the map into a block of non-overlapping rectangular regions. In line 5, the regions that are partitioned but do not satisfy the constraints are further adjusted, and finally the set containing all optimal cloaking regions is returned.

B. Self-organizing map

For SOM, the traditional usage is clustering analysis. We have also done work related to k-anonymity using the SOM clustering algorithm. However, this approach does not significantly reduce the extra processing time due to high-dimensional attributes. Based on this situation, we use SOM for data preprocessing, and after inputting appropriate quasi-identifiers, we downscale the multidimensional quasi-identifier data with the help of SOM while preserving the topology oh-dimensional space.

Step 1: **Weight initialization**. The weights of the competing layers of the SOM are initialized by the initialization function as

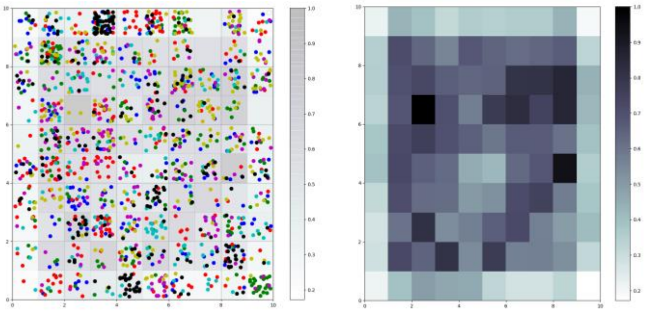
$$W \in R^{X \times Y \times N} \tag{5}$$

where $X \times Y$ denotes the number of nodes in the competitive layer and N denotes the n feature dimensions of the input samples.

Step 2: **Sample selection**. The training process is similar to stochastic gradient descent, where one data sample is selected at a time *x* for learning.

Step 3: **Distance calculation**. The distance between sample *x* and the other node *y* of the competitive layer is calculated by a distance function (we use Euclidean distance):

$$\mathrm{d} = \sqrt{\sum_{i}^{n} (x_i - y_i)^2} \tag{6}$$



(a) scatter map (b) U-Matrix

Fig.4. SOM distribution of trajectory dataset

Step 4: **Competition process**. The node (i, j) with the closest distance to sample x is selected as the winner node, also called the best matching unit (BMU).

Step 5: **Self-organization process**. Determine the winning neighborhood according to the neighborhood radius sigma σ, and calculate the magnitude g of each node update by the neighborhood function (usually the closer to the superior node, the larger the update magnitude).

Step 6: **Weight update**. Updating the weights of the nodes in the superior neighborhood by learning rate η.

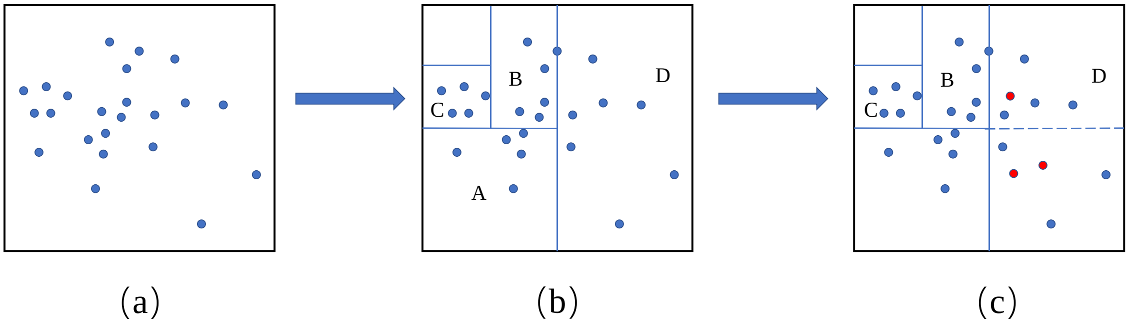


Fig.5. The process of adjusting cloaking region

$$W = W + \eta g \cdot (x - W) \tag{7}$$

We can also visualize high-dimensional data through SOM. The Fig.4(a) is an overview of distributed samples, it shows the mapping of high-dimensional location data into two-dimensional vectors on a two-dimensional coordinate system after SOM preprocessing. Fig.4(b) shows a U-Matrix plot, the U-Matrix is obtained by calculating the weight distance between each node and its neighboring nodes, and the darker colored points in the figure indicate that the distance is larger. Comparing Fig.4(a) and Fig.5(b), we can find that the distribution of the dark positions in Fig.4(b) and the dense points in Fig.4(a) are the same.

C. Constructing cloaking region

Algorithm 2 describes the construction process of cloaking regions. In the latest k-anonymity algorithm, researchers have developed algorithms that prioritize the adjustment to determine the weights of attributes. In our algorithm, there is no need to use this prioritized adjustment algorithm since the high-dimensional data is mapped into two dimensions. In addition to this, as described in Section 3 of this paper, the weights in SOM differ from convolutional neural networks in that the output nodes in SOM contain weights as coordinates rather than as a result of adding weights. Therefore, in line 4 to 12, we use a reordering based on the magnitude of the impact of each sensitive attribute on the information loss metric for the whole dataset.

The priority adjustment algorithm is as follows: suppose we have a data table S containing attributes $Q_1, Q_2, \ldots, Q_n$ with N tuples, each with attributes $q_1, q_2, \ldots, q_n$. We first perform a greedy partitioning algorithm using the given attributes where the attributes are not assigned priority, which means that the attribute first selected in the algorithm for partitioning is arbitrary. Other attributes are considered after there are no divisible tuples on that attribute. By performing this process, we are able to determine the impact on the availability of the dataset based on the information loss value IL_i computed by the algorithm executed *n* times, which in turn sets the priority order of each attribute. Then, in line 10, we input the list of quasi-identifiers sorted by priority into the SOM for training to obtain two-dimensional nodes, i.e., new quasi-identifiers. The above algorithm needs to be executed only once, and the quasi-identifier priorities do not need to be adjusted after that.

Algorithm 2: Constructing Cloaking Region Algorithm

```
Input: Dataset S, Quasi-identifiers QI, Anonymity degree
       threshold k, Diversity threshold l, Distance threshold
       t, Information loss θ;
Output: Cloaking Regions
1  Final_partitions = Ø
2  Flag = False
3  partition ← the index of dataset S
4  while Flag ≠ True do
5     for each qi ∈ QI do
6        $IL_i$ ← Cloaking_Region (S, qi, partition)
7        Priority ( $IL \cup \{IL_i\}$)
8        sort the QI by Priority (IL)    //update QI series by
   information loss priority
9     end for
10    QI ← SOM (S, QI, som_size)
11    return QI, Flag=True
12 end while
13 while Flag =True do
14    for qi, i, the span of partition do
15       if the number of records in $i^{th}$ partition < k then
16          continue
17       else if the distinct value in $i^{th}$ partition > l or the
   distance in $i^{th}$ partition < t then
18          lp, rp = partite (S, qi, partition)   // split the
   partition by the median value
19       end if
20       if IL(lp) ≥ IL(partition) or IL(rp) ≥ IL(partition)}
   then
21          lp ← Adjust($lp, qi, \theta$)
22          rp ← Adjust($lp, qi, \theta$)
23       end if
24          final_artitions.$apppend(lp, rp)$
25    end for
26  return final_artitions
27 end while
```

Lines 13 to 27 of the algorithm are the partitioning and adjustment process of the cloaking region. In lines 15 to 19, the partition strategies are designed based on the constraints, i.e.,

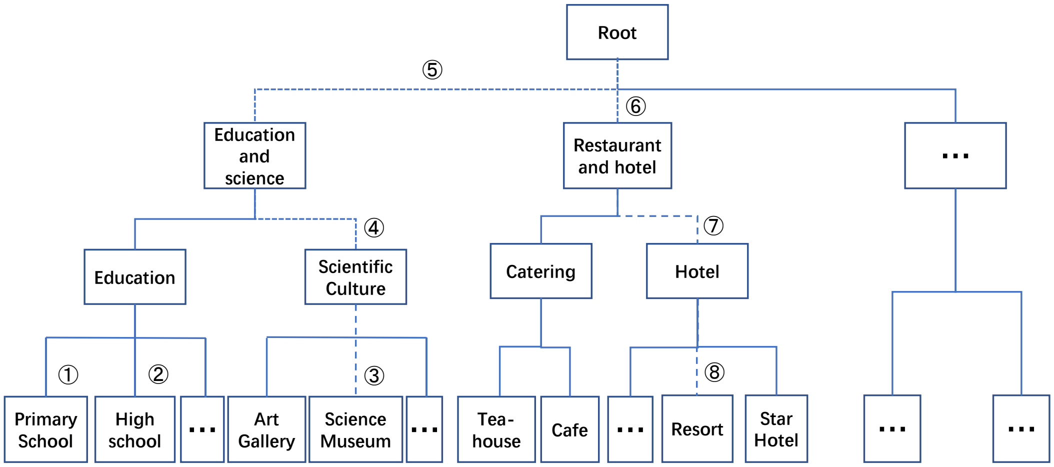


Fig.6. The process of adjusting cloaking region

partitions are required to satisfy k-anonymity, l-diversity and t-closeness [17]. The specific description is as follows:

Definition 6: For any k-anonymous region, the number of locations in the region must be greater than or equal to *k*.

$$Num(l \in CR_i) \geq k \tag{8}$$

where CR_i denotes the i^{th} cloaking region and *l* denotes all locations in CR_i.

Definition 7: For a sequence q of location queries issued from the i^{th} cloaking region, the number of queries with different service types must be greater than or equal to *l*.

$$|CR_i(q)| \geq l \tag{9}$$

where *{q}* denotes all the location queries in CR_i.

Definition 8: The difference between the distribution of sensitive attributes in the i^{th} cloaking region and the distribution of sensitive attributes in the whole dataset cannot exceed the threshold *t*.

$$P - G \leq t \tag{10}$$

where *P* denotes the frequency of a user in CR_i and *G* denotes the global frequency of a user in all data, and $P = \frac{count}{CR_{i\_}count}, G = \frac{count}{total\_count}$.

The optimal anonymous partition can be constructed based on the above three constraints. If (8) is not satisfied, the next cycle continues, and if (8) is met but (9) and (10) are not satisfied, the partition is adjusted, and finally the set of adjusted partitions is returned.

D. Adjusting cloaking region

The location semantic tree (LST) is a hierarchical generalized tree structure constructed base on POIs, in which all locations within the current mobile network coverage are organized according to their semantics, where each leaf node represents a semantic label and its parent node represents the category to which the child nodes belong, i.e., the semantic label in a broader sense. The depth of the LST is *h*, and the value of h is equal to the number of layers classified in the LST plus one. The height of the semantic tree, i.e., the generalization level, implies the granularity of the POI classification. Conversely, if the height of the tree is too high, it is not meaningful and adds unnecessary computational overhead. Note that due to space constraints, only some of the nodes are shown in this positional semantic tree, and the rest of the nodes are presented as ellipses.

Algorithm 3: Adjusting Cloaking Region Algorithm

```
Input: The partition AR that needs to be adjusted,
         Quasi-identifiers qi, Anonymity degree
         threshold k, location semantic distance u,
         candidate location set CLS
Output: Adjusted Regions
Build location semantic tree LST
n ← The number of real locations in AR
BLS = Ø
for each cloc in CLS do
  while $d_{sem}$(loc, cloc) ≤ $u$ do  //loc is the real location
      add cloc to BLS
      continue
      if count(BLS) $= k - n$ then:
            break
      end if
  end while
end for
return $AR \cup \{BLS\}$
```

The next section discusses the definition of semantic distance and the relationship between information loss and semantic distance. For any two locations i, j with semantic labels $l_i, l_j(\mathrm{i} \neq \mathrm{j})$, the semantic distance $d_{sem}(l_i, l_j)$ is defined as follows.

$$d_{sem}(l_i, l_j) = \frac{d_{graph}(l_i, l_j)}{d_{graph}(l_i, root) + d_{graph}(l_j, root)} \tag{11}$$

where d_{graph} denotes the graph distance between two semantic labels in the location semantic tree and is measured by the number of edges in the shortest path connecting them. In addition, root denotes the root node of the semantic tree. Take Fig.7 as an example, the semantic distance between primary school and high school can be expressed as 2 / (3+3) = 1/3, and the semantic distance between primary school and science museum is (2+2) / (3+3) = 2/3. Taking the markers in the figure as an example, we can get the semantic distance matrix of the distances between l_1, l_2, l_3, and l_4 as follows.

$$\text{SDM}=\begin{bmatrix} 0 & 1/3 & 2/3 & 1 \\ 1/3 & 0 & 2/3 & 1 \\ 2/3 & 2/3 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \tag{12}$$

As shown in Fig.6(b), when constructing anonymous partitions, the rectangular anonymized regions vary in size because of the uneven distribution of locations. Sparse location scattering in regions spanning too large a range can result in increased information loss and also the risk of exposed locations. Therefore, in line 3 of Algorithm 3, we set an optimal position set BLS to store the dummy locations that satisfy the semantic difference filtered from the candidate pseudo position set CLS. Then, in lines 4 to 12, we iterate through the positions in the candidate position set and calculate the semantic similarity between it and the true position. If $d_{sem} \geq u$, this fake location is put into the BLS until the sum of the true locations and the dummy locations in the region is k. Finally, Algorithm 3 returns the anonymous region to which the optimal dummy locations are added. As shown in Fig.6(c), the red points are the false locations filtered out by computing the location semantics.

V.Privacy Analysis

A. Theoretical Analysis

From (2) and (3), the denominator of the mathematical expression for information loss is the range of value domains of sensitive attributes, and the numerator is the difference between two values in the same generalized equivalence class.

$$0 \leq IL_{A_i^n}(1)=\frac{q_i - p_i}{A_i^n} = \frac{d(l_i, l_j)}{d(l_{min}, l_{max})} \leq 1 \tag{13}$$

where $d(l_i, l_j)$ denotes the difference between the numerical attributes l_i and l_j and $d(l_{min}, l_{max})$ denotes the value domain of the numerical attribute A_i^n.

From (7), it can be seen that because the generalization tree is a multinomial tree with a hierarchical structure and its depth is a fixed value, the denominator of the mathematical expression for the positional semantic distance d_{sem}, i.e., the sum of the distances from each semantic label to the root node of the semantic generalization tree, is a fixed value and the numerator is the sum of the reachable hops between two semantic labels; note that the reachable hops between each semantic label generalized to the same semantic hierarchical level are the same. Thus the positional semantic distance can be equivalently expressed as

$$0 \leq d_{sem}(l_i, l_j) = \frac{d_{LST}(l_i, l_j)}{d_{LST}(l_i, root) + d_{LST}(l_j, root)} = \frac{hop(l_i, l_j)}{depth(LST)} \leq 1 \tag{14}$$

where $hop(l_i, l_j)$ denotes the number of hops between semantic labels l_i and l_j, and $depth(LST)$ denotes the depth of the location semantic generalization tree.

Comparing (13) and (14), we can conclude that positional semantics is the same as information loss in that both are locally divided by global, with the numerator being the distance between two elements and the denominator being the range of the set containing the two elements. Therefore, we can conclude that the positional semantic distance is positively related to the information loss.

B. Security Analysis

MAP-Matching attack: An attacker monitors successive updates or queries, and if a member of the collection changes, the attacker can infer which user sent the initial update or query. In the algorithm proposed in this paper, it is assumed that the query A sent by the user is anonymized to generate K anonymizer sets (A, B, C, D), and since the generated virtual locations consider semantic similarity, the generated virtual locations are duplicated and not completely different. Therefore the anonymity set of the second query is (A, B, C, E), and at this point the client cannot infer that the initial query issued is A.

Background knowledge attack: The attacker can obtain other data about the user by other means, e.g., if the occupation is student, in this case, the attacker can infer the user's school by simply filtering the trajectories of the user's frequent behavior near the school. However, the false location set BLS generated by the anonymous region adjustment algorithm proposed in this paper contains optimal false locations, and the number of false locations is not unique and is in the interval [0, k-1]. So the attacker cannot obtain the precise private information of the user even by using background knowledge.

Location homogeneity attack: Homogeneity attack means that the values of the corresponding sensitive attributes within a certain k-anonymity group are also the same. In the location privacy domain, a homogeneity attack means that the steganographic region covers only one sensitive location (e.g., hospital), then an attacker can easily obtain the user's location semantics and thus grasp the user's privacy information. In the approach of this paper, by constructing a location semantic tree, grading all semantic labels, and proposing the concept of location semantic distance, the generated false location semantic distance is kept in the interval [0.2, 0.8], so it is more diverse and there is no situation that all locations are distributed around the same point of interest, which can counter the location homogeneity attack.

From the above analysis, it can be seen that our SOKA algorithm can effectively resist the inference attack of LSP.

C. Complexity Analysis

The time complexity of the attribute-first adjustment algorithm in the proposed algorithm is $O(N)$, and N is the size of the QI sequence. Since the time complexity of the efficient SOM algorithm used is $O(N)$, the time complexity of the first part of the algorithm is $O(N)$. The time complexity of the anonymous region division algorithm is $O(m * log(N))$, where m is the number of final divisions. The time complexity of the anonymous region adjustment algorithm is always $O(C)$, and C is the number of elements in the list, so. Therefore, the time complexity of the algorithm proposed in this paper is $O(N\,log(N))$.

VI. Experiments

In this paper, the SOKA algorithm is compared with two k-anonymity algorithms proposed in recent years. The first algorithm is an improved k-anonymity algorithm based on clustering proposed by Zheng et al. [21]; this algorithm uses locally optimal clustering. The other algorithm is the greedy search algorithm proposed by Liang et al. [9], which defines k-anonymity as a linear programming problem. We refer to these two algorithms as Improved Clustering and Greedy Search, respectively. We implement the above algorithms in Python 3.7 with Intel Core i5 CPU 2.3GHz and 8GB RAM, running with Windows 10.

A. Datasets

The Gowalla dataset, with 6,442,892 original records, includes 107092 users and 1280969 POIs. Before starting the experiment, we cleaned and filtered the dataset to filter out the records with missing values and duplicates. In addition, we visualized the real dataset using open-street map as shown in Fig.3 above. We use 100%, 10%, and 1% of the original dataset to obtain the impact of different dataset sizes on the algorithms. Besides, the records with more than 10 user visits and more than 15 POI visits are selected for observation and behavioral analysis.

B. Utility Measures

In our experiments, we use the three evaluation indicators mentioned in Section 3, namely information loss, anonymous success rate, and running time, and investigate their effects on the experiments by varying the size of *k*, the number of user check-in locations and quasi-identifier attributes. In addition, the values of *l* and *t* are out of the scope of this paper, and we take values of *l* and *t* to look at 2 and 0.1, respectively.

For the cloaking regions C_1, C_2, …, C_n, location queries q_1, q_2, …, q_m, then the anonymous success rate is defined as follows.

$$\text{ASR} = \frac{Num(q_i)}{\sum_{j=1}^{n} C_j} \tag{15}$$

Where $Num(q_i)$ denotes the queries that are not recognized by the attacker, and C_j denotes the number of location queries in region j.

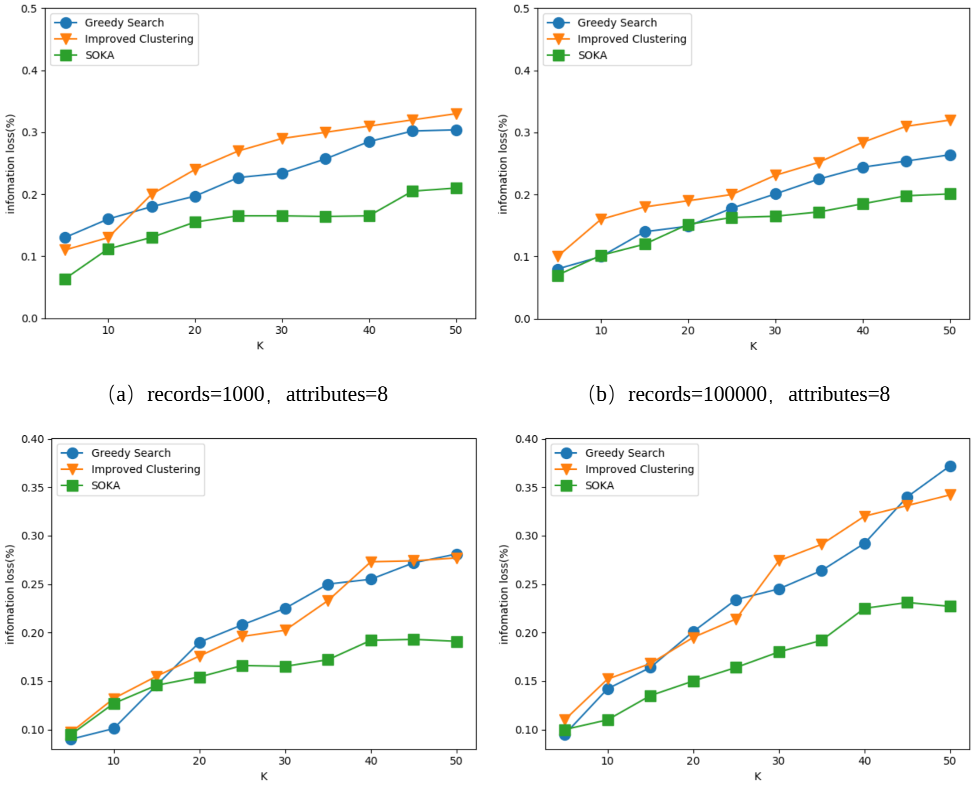
C. Information loss

Fig.7 shows the effects of the number of records and the number of attributes in the dataset on the information loss performance of the algorithms, respectively. With the increases of *k*, the information loss of all three algorithms tends to increase. As seen in Fig.7(a) and Fig.7(b), a smaller value of *k* means that fewer locations within the anonymous group need to be constructed, and it is more difficult to construct a group with reasonable position distribution. Because the optimal dummy positions added in this paper expand the number of locations in the region, the location distribution within each rectangular cloaking region tends to be optimal. In contrast, the Improved Clustering is difficult to construct a small range of anonymous groups for sparse data. In addition, the Greedy Search algorithm is susceptible to the formation of bad clusters by outlier points, resulting in excessive information loss. On the contrary, when *k* < 10, the advantage of SOKA algorithm is not obvious, however, when *k* > 10, our algorithm has an increasing advantage of up to 20%. This is because when k increases, the span of the constructed cloaking regions becomes larger and the information loss is greater, when the cloaking region adjustment strategy splits the cloaking regions according to the span of sensitive attribute values, which makes our algorithm have better performance when faced with different records and attributes which shown in Fig.7(c) and Fig.7(d). In addition, we believe that the best balance of privacy preserving ability and location service quality can be achieved when *k* = 10, so this algorithm has good reliability in terms of information loss.

Fig.7(e) and Fig.7(f) visualize the impact of the above thresholds on data availability by setting more detailed parameters. We can see from the figure that the overall improvement is about 10%. In general, our algorithm can maintain low information loss and ensure data availability for data of different sizes and dimensions.

D. Running time

Fig.8 shows the variation of anonymization processing time with *k* for different thresholds for Greedy Search, Improved Clustering and SOKA algorithms. As shown in Fig.8(a) and Fig.8(b), when the number of records increases, the user locations are more densely distributed, so it is easier to construct cloaking regions and the running time decreases as *k* increases. When faced with large data sets, our algorithm significantly improves the problem that traditional multi-dimensional partitioning algorithms such as Mondrian consume too much time and memory to handle high-dimensional data. As shown in Fig.8(c) and Fig.8(d), when sensitive attributes are increased, our algorithm converts them into two-dimensional vectors by SOM, so our algorithm saves a lot of time overhead and always maintains a stable level. Since Greedy Search is essentially k-member algorithms, it needs to calculate the average distance between clusters and clusters and between



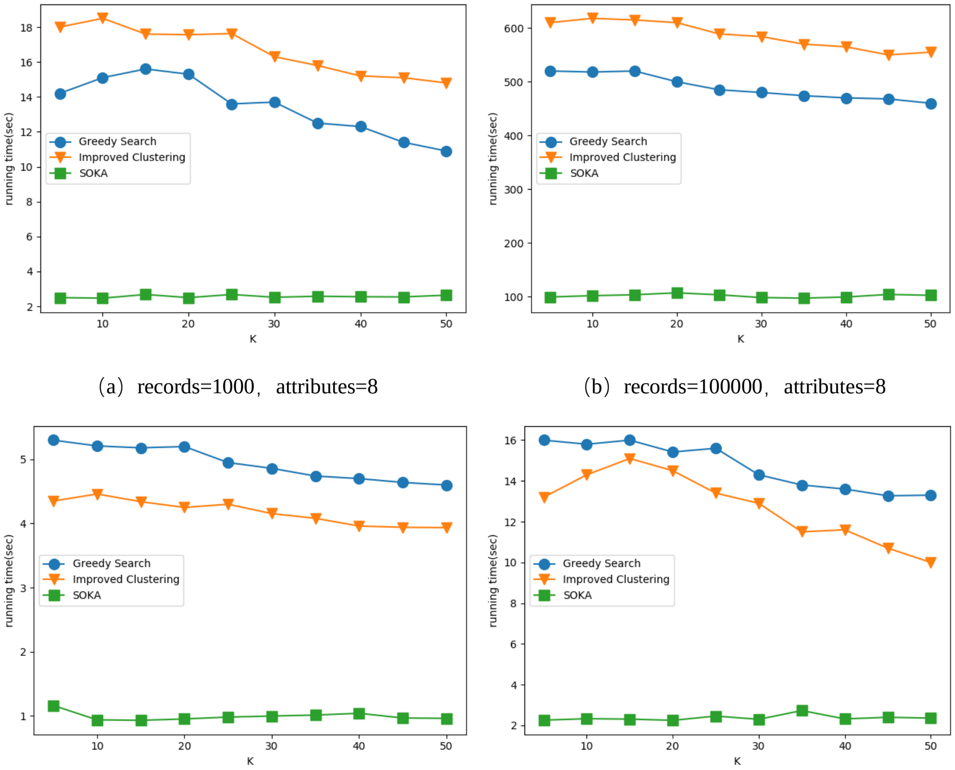
（a） records=1000， attributes=8

（b） records=100000， attributes=8

（c） attributes=2， records=1000

（d） attributes=7， records=1000

Fig.7. The effect of thresholds on information loss



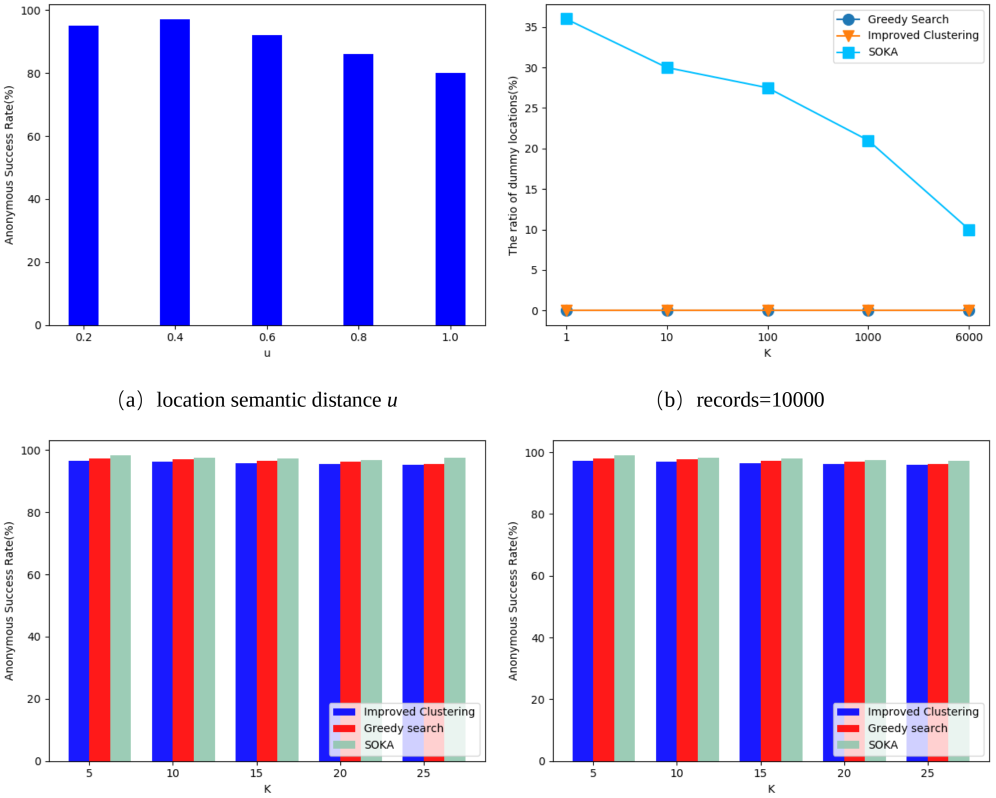
（a） records=1000， attributes=8

（b） records=100000， attributes=8

（c） attributes=2， records=1000

（d） attributes=7， records=1000

Fig.8. The effect of thresholds on running time



（a） location semantic distance *u*

（b） records=10000

（c） attributes=8， records=1000

（d） attributes=8， records=100000

Fig.9. The effect of thresholds on anonymous success rate

clusters and points to construct new clusters. This approach performs unnecessary calculations and brings excessive time consumption. Improved Clustering is density-based clustering algorithms, it requires ranking the k-density of users and discovering the nearest neighbor points, and thus requires greater time cost than the partitioning strategy in this paper. Therefore, the time complexity of both of the above algorithms is higher than the SOKA algorithm proposed in this paper. As can be seen from the figure, our proposed algorithm can save 70% of anonymous processing time especially when facing large data sets.

E. Anonymous success rate

The effect of location semantic difference u on the anonymization success rate is shown in Fig.9(a). When u is greater than 0 and less than 0.4, the anonymity success rate rises. This is because a smaller semantic distance indicates that the semantic labels are more similar, and the POI is even the same when u is 0. While the user can be served precisely in this case, it also gives the attacker information that is detailed enough to infer the user's identity. As the semantic difference $u > 0.4$, the anonymization success rate shows a decreasing trend. This is because the lower the service similarity between the fake locations and the real locations, the higher the risk of being inferred by attackers using background knowledge attacks. Fig.9(b) then shows the ratio of the number of false locations to the total number of locations in the anonymization region when the number of locations in the map is 10000. As can be seen from the figure that the ratio of false locations for the SOKA algorithm increases with *k*. This is because the larger *k* is, the more concentrated the location distribution is, the more users are around the same points of interest, and the less false locations need to be added. the Greedy Search algorithm and the Improved Clustering algorithm do not need to add false locations, but are poorly resistant to background knowledge attacks.

As shown in Fig.9(c) and Fig.9(d), the anonymization success rate decreases when the number of location queries is small and the users are dispersed under a certain k. The higher the number of queries, the better it is to construct the set of false locations that satisfy the constraints. The algorithm proposed in this paper performs better than Improved Clustering and Greedy Search because of the addition of confusing dummy locations. Even if the Improved Clustering algorithm selects denser regions to build anonymous groups, these locations are more easily detected because they do not take into account the location semantics and therefore do not necessarily match the user's behavioral characteristics.

VII. Conclusion

In this paper, we combine a self-organizing neural network with a k-anonymity algorithm and deploy it for the first time to construct a hidden region for LBS location privacy protection.

This approach can transform high-dimensional data into two-dimensional coordinates while being able to maintain the topology of the data. In addition, we construct a location semantic tree based on the location semantics within the cloaking region, generate a set of candidate dummy locations based on the generalization level of the semantic labels in this semantic tree, and select different numbers of the best locations from them to add to the cloaking region. This method greatly reduces the time overhead, while reducing information loss and improving data quality, and experiments demonstrate the effectiveness of the method proposed in this paper. In the future, we plan to analyze the probability of considering historical queries when generating fake locations. Meanwhile, in this paper, we mainly study the location privacy protection for the snapshot query case, and the next step will be to consider the location privacy protection for continuous queries, i.e., trajectory privacy protection.

VIII. Conclusion

References

[1] Jiang, H., Li, J., Zhao, P., Zeng, F., Xiao, Z., & Iyengar, A. "Location privacy-preserving mechanisms in location-based services: A comprehensive survey." ACM Computing Surveys (CSUR) 54.1 (2021): 1-36.

[2] Jiang, J., Han, G., Wang, H., & Guizani, M. "A survey on location privacy protection in wireless sensor networks." Journal of Network and Computer Applications 125 (2019): 93-114.

[3] Sweeney, Latanya. "k-anonymity: A model for protecting privacy." International journal of uncertainty, fuzziness and knowledge-based systems 10.05 (2002): 557-570.

[4] Yazdanjue N, Fathian M, Amiri B. Evolutionary algorithms for k-anonymity in social networks based on clustering approach[J]. The Computer Journal, 2020, 63(7): 1039-1062.

[5] Wang J, Cai Z, Yu J. Achieving personalized $ k $-anonymity-based content privacy for autonomous vehicles in CPS[J]. IEEE Transactions on Industrial Informatics, 2019, 16(6): 4242-4251.

[6] Iyer, Rohan, et al. "Spatial K-anonymity: A privacy-preserving method for COVID-19 related geospatial technologies." arXiv preprint arXiv:2101.02556 (2021).

[7] Gruteser, Marco, and Dirk Grunwald. "Anonymous usage of location-based services through spatial and temporal cloaking." Proceedings of the 1st international conference on Mobile systems, applications and services. 2003.

[8] Zhang, W., Yin, G., Sha, Y., & Yang, J. "Protecting the moving user's locations by combining differential privacy and-anonymity under temporal correlations in wireless networks." Wireless Communications and Mobile Computing (2021).

[9] Liang, Yuting, and Reza Samavi. "Optimization-based k-anonymity algorithms." Computers & Security 93 (2020): 101753.

[10] Uphaus, P., Beringer, B., Siemens, K., Ehlers, A., & Rau, H. "Location-based services–the market: success factors and emerging trends from an exploratory approach." Journal of Location Based Services 15.1 (2021): 1-26.

[11] Yang, X., Gao, L., Zheng, J., & Wei, W. "Location privacy preservation mechanism for location-based service with incomplete location data." IEEE Access 8 (2020): 95843-95854.

[12] Nisha, N., Natgunanathan, I., Gao, S., & Xiang, Y. "A novel privacy protection scheme for location-based services using collaborative caching." Computer Networks (2022): 109107.

[13] Gursoy, M. E., Liu, L., Truex, S., & Yu, L. "Differentially private and utility preserving publication of trajectory data." IEEE Transactions on Mobile Computing 18.10 (2018): 2315-2329.

[14] Wen, R., Zhang, R., Peng, K., & Wang, C. "Protecting Locations with Differential Privacy against Location-Dependent Attacks in Continuous LBS Queries." 2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). IEEE, 2021.

[15] Ren, Y., Li, X., Miao, Y., Deng, R., Weng, J., Ma, S., & Ma, J. (2022). "DistPreserv: maintaining user distribution for privacy-preserving location-based services." IEEE Transactions on Mobile Computing (2022).

[16] Alotaibi, M., Ibrahem, M. I., Alasmary, W., Al-Abri, D., & Mahmoud, M. "UBLS: User-Based Location Selection Scheme for Preserving Location Privacy." 2021 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE, 2021.

[17] Qiu, G., Guo, D., Shen, Y., Tang, G., & Chen, S. "Mobile semantic-aware trajectory for personalized location privacy preservation." IEEE Internet of Things Journal 8.21 (2020): 16165-16180.

[18] Slijepčević, D., Henzl, M., Klausner, L. D., Dam, T., Kieseberg, P., & Zeppelzauer, M. "k-anonymity in Practice: How generalisation and suppression affect machine learning classifiers." Computers & Security 111 (2021): 102488.

[19] Zheng, L., Yue, H., Li, Z., Pan, X., Wu, M., & Yang, F. "K-anonymity location privacy algorithm based on clustering." IEEE Access 6 (2018): 28328-28338.

[20] Ashkouti F, Sheikhahmadi A. "DI-Mondrian: Distributed improved Mondrian for satisfaction of the L-diversity privacy model using Apache Spark." Information Sciences 546 (2021): 1-24.

[21] Li, F., Chen, Y., Niu, B., He, Y., Geng, K., & Cao, J. "Achieving personalized k-anonymity against long-term observation in location-based services." 2018 IEEE Global Communications Conference (GLOBECOM). IEEE, 2018.

[22] Z. Tu, K. Zhao, F. Xu, Y. Li, L. Su and D. Jin, "Beyond k-anonymity: protect your trajectory from semantic attack." 2017 14th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON). IEEE, 2017.

[23] B. Niu, Q. Li, X. Zhu, G. Cao and H. Li, "Achieving k-anonymity in privacy-aware location-based services." IEEE INFOCOM 2014-IEEE conference on computer communications. IEEE, 2014.

[24] Shaobo Zhang, Xiong Li, Zhiyuan Tan, "A caching and spatial K-anonymity driven privacy enhancement scheme in continuous location-based services." Future Generation Computer Systems 94 (2019): 40-50.

[25] Siddiqie, S., Mondal, A., & Reddy, P. K. "An improved dummy generation approach for enhancing user location privacy." International Conference on Database Systems for Advanced Applications. Springer, Cham, 2021.

[26] Li, Chao, and Balaji Palanisamy. "Reversible spatio-temporal perturbation for protecting location privacy." Computer Communications 135 (2019): 16-27.

[27] Wu, Z., Li, G., Shen, S., Lian, X., Chen, E., & Xu, G. "Constructing dummy query sequences to protect location privacy and query privacy in location-based services." World Wide Web 24.1 (2021): 25-49.

[28] J. Tang, H. Zhu, R. Lu, X. Lin, H. Li and F. Wang, "DLP: Achieve customizable location privacy with deceptive dummy techniques in lbs applications." IEEE Internet of Things Journal 9.9 (2021): 6969-6984.

[29] Pan, X., Nie, S., Hu, H., Yu, P., & Guo, J. "Reverse nearest neighbor search in semantic trajectories for Location based Services." IEEE Transactions on Services Computing (2020).

[30] N. Nisha, I. Natgunanathan and Y. Xiang, "An Enhanced Location Scattering Based Privacy Protection Scheme," in IEEE Access, vol. 10, pp. 21250-21263, 2022.

[31] Zhao P, Li J, Zeng F, et al. ILLIA: Enabling $ k $-anonymity-based privacy preserving against location injection attacks in continuous LBS queries[J]. IEEE Internet of Things Journal, 2018, 5(2): 1033-1042.

[32] Tu Z, Zhao K, Xu F, et al. Protecting trajectory from semantic attack considering ${k} $-anonymity, ${l} $-diversity, and ${t} $-closeness[J]. IEEE Transactions on Network and Service Management, 2018, 16(1): 264-278.

[33] Tan, Z., Wang, C., Yan, C., Zhou, M., & Jiang, C. "Protecting privacy of location-based services in road networks." IEEE Transactions on Intelligent Transportation Systems 22.10 (2020): 6435-6448.