Hindawi
Security and Communication Networks
Volume 2020, Article ID 8892954, 13 pages
https://doi.org/10.1155/2020/8892954

Research Article

A Dynamic Privacy Protection Mechanism for Spatiotemporal Crowdsourcing

Tianen Liu,[1] **Yingjie Wang**,[1] **Zhipeng Cai**,[2] **Xiangrong Tong**,[1] **Qingxian Pan**,[1] **and Jindong Zhao**[1]

[1]*School of Computer and Control Engineering, Yantai University, Yantai 264005, China*
[2]*Department of Computer Science, Georgia State University, Atlanta 30303, GA, USA*

Correspondence should be addressed to Yingjie Wang; towangyingjie@163.com

Received 25 May 2020; Revised 26 June 2020; Accepted 29 July 2020; Published 28 August 2020

Academic Editor: Xiaolong Xu

In spatiotemporal crowdsourcing applications, sensing data uploaded by participants usually contain spatiotemporal sensitive data. If application servers publish the unprocessed sensing data directly, it is easy to expose the privacy of participants. In addition, application servers usually adopt the static publishing mechanism, which is easy to produce problems such as poor timeliness and large information loss for spatiotemporal crowdsourcing applications. Therefore, this paper proposes a spatiotemporal privacy protection (STPP) method based on dynamic clustering methods to solve the privacy protection problem for crowd participants in spatiotemporal crowdsourcing systems. Firstly, the working principles of a dynamic privacy protection mechanism are introduced. Then, based on k-anonymity and l-diversity, the spatiotemporal sensitive data are anonymized. In addition, this paper designs the dynamic k-anonymity algorithm based on the previous anonymous results. Through extensive performance evaluation on real-world data, compared with existing methods, the proposed STPP algorithm could effectively solve the problem of poor timeliness and improve the privacy protection level while reducing the information loss of sensing data.

1. Introduction

With the widespread use of wireless communication technologies and smart mobile terminals, location-based services (LBS) are becoming more and more popular [1, 2]. In many spatiotemporal crowdsourcing applications, participants receive corresponding rewards by submitting their own sensing tasks to crowdsourcing application servers [3]. However, the submitted sensing data contain the participants' spatiotemporal data [4, 5]. If the crowdsourcing application server publishes these spatiotemporal data without processing, the participant's privacy information will be obtained by attackers [6, 7]. More importantly, attackers can infer the participant's recent medical service or entertainment venue by locating his spatiotemporal information to understand his health status, preferences, time, and scope of the outing [8]. Therefore, in a spatiotemporal crowdsourcing application, it is especially important to protect the spatiotemporal information of participants. The privacy protection technology based on spatiotemporal crowdsourcing has also become a research hotspot in the field of spatiotemporal crowdsourcing systems [9].

In order to ensure that participants' spatiotemporal private information is not leaked when publishing data, a large amount of work on spatiotemporal privacy protection is devoted to disturbing and anonymizing the spatiotemporal data that may reveal personal whereabouts. To et al. [10] proposed a protection framework based on differential privacy. The workers in spatiotemporal crowdsourcing first submit their real location information to the truthful mobile service provider, and the mobile service provider uses a grid-based method to construct the private spatial decompositions (PSDs) for the original location information and adds Laplace noise to process workers' real location data for privacy protection purposes. Vu et al. [11] proposed a privacy protection mechanism based on local sensitive hashing to group participants' positions. Each group contains at least k participants to achieve spatial anonymity. The

ideal partition of spatial data under low time complexity is realized, and participants' location information is protected in a spatiotemporal crowdsourcing application scenario. In [12], the problem of insufficient diversity of k-anonymity algorithm to the participants' sensitive locations is solved, and the probability of participants' access to the sensitive locations is limited or the probability analysis based on adversary knowledge is used to ensure the location diversity.

However, most researchers currently only consider the data publishing in static scenario. The attackers use historical publishing results to reveal sensitive information during static publishing, for example, to compare with the results of the previous publishing. In spatiotemporal crowdsourcing applications, many data analysis applications actually involve dynamic data publishing. For example, in order to plan travel routes for special vehicles (cash trucks, ambulances, fire engines, etc.), it is necessary to issue a sensing task to collect road traffic jams [13, 14]. For such a spatiotemporal sensitive task, application server needs to dynamically publish sensing data submitted by participants to improve the timeliness of the task. Dynamic sensing data change constantly over time, so it is often necessary to anonymize and dynamically publish sensing data at different times. However, most anonymity algorithms are invalid when dealing with the dynamic publishing of spatiotemporal data [15]. The previous anonymity result cannot be effectively utilized. Because of the big data scenario, the time complexity of algorithms is high, and the timeliness is poor [16]. Moreover, most researchers proposed privacy preserving for participant's location information but failed to consider that attackers can also infer other private information based on participant's spatiotemporal information. According to these problems, we research the privacy protection for spatiotemporal privacy information in spatiotemporal crowdsourcing systems, and the following issues should be improved further:

(1) In the process of dynamic data publishing, the results after anonymization should be effectively utilized instead of unifying the anonymization of incremental data with previous data to improve the timeliness of dynamic publishing of big data

(2) In the process of anonymizing the location attribute of participants, the time attribute is added to effectively avoid the background knowledge attack and homogeneity attack against the location attribute

In order to solve the above problems, we propose a spatiotemporal privacy protection method for spatiotemporal crowdsourcing systems. The contributions of this paper are shown as follows:

(1) A dynamic publishing algorithm based on spatiotemporal data privacy protection is designed by improving k-anonymity. When incremental data arrive, the anonymization result of the last time will be utilized to solve the timeliness problem of dynamic publishing.

(2) Based on the traditional position coordinate, a time axis is added to form the spatiotemporal information of participants, and the anonymization of participants' spatiotemporal information is carried out by applying k-anonymity and l-diversity methods, so as to solve the background knowledge attack and homogeneity attack problems.

(3) In order to verify the effectiveness of the proposed privacy protection method, the comparison experiments with k-anonymity and variable centroid location aggregation (VCLA) [17] algorithms are conducted on two real-world datasets.

The structure of the paper is as follows. Section 2 introduces the related works of spatiotemporal privacy protection. Section 3 introduces the proposed spatiotemporal privacy protection method for spatiotemporal crowdsourcing systems. In Section 4, the real-world datasets and the existing anonymity algorithms are used for evaluating the performance of the proposed method. Section 5 concludes the paper and presents the future work.

2. Related Works

In this section, we will introduce the related works about privacy protection methods for spatiotemporal data and dynamic publishing of sensitive data under a participatory sensing environment. Participatory sensing (PS) refers to the formation of a mobile Internet through daily mobile devices, where data are sensed, collected, analyzed, or screened by the public and professional users and then uploaded to the participatory sensing network [18]. With the popularization of mobile terminals and the rapid development of wireless sensor technology, the application of PS is becoming more and more common in real life. For example, in [19], Chen et al. studied the energy-efficient task offloading in mobile edge computing (MEC). However, in the process of task offloading, the privacy of participants will be exposed. In order to deal with the problems that participants' privacy will be exposed during the task offloading process, Xu et al. [20] put forward a two-phase offloading optimization strategy for joint optimization of offloading utility and privacy in edge computing. Further, Xu et al. [21] discussed the problem that transmitted information is vulnerable to attack and may cause incomplete data during task offloading. A blockchain-enabled computation offloading method was proposed to ensure data integrity. In the implementation process of these participatory sensing applications, sensing tasks uploaded by participants will mark personal spatiotemporal data, which brings great risks to the privacy security and personal safety of participants. Therefore, while people enjoy the convenience brought by LBS, their privacy is also at risk of being exposed [22].

In LBS, using anonymous technology to solve the location privacy problem of participants has been widely studied [23]. The k-anonymity technology was firstly proposed by Samarati and Sweeney [24]. The parameter k specifies the maximum risk of information disclosure that users can bear. It requires at least k indistinguishable records on the quasi-identifier in published data, so that attackers cannot identify the specific individual that the privacy information belongs to, so as to protect personal privacy. In

[25], the clustering-based k-anonymity strategy is adopted to protect the privacy disclosure of wearable owners when they upload sensing data. In [26], a k-anonymous location privacy protection method based on coordinate transformation was proposed for the problem that the third-party truthful server (TTP) was often untruthful in real life [27]. The anonymous server receives the coordinate-converted participant location and constructs an anonymous area without knowing the user's actual location, thereby protecting the participant's location privacy. In [28], the optimal k value of the current user is determined according to the user's environment and social attributes, and a location protection k-anonymity method based on the truthful chain was proposed to protect the location privacy of participants while ensuring the quality of service.

However, k-anonymity cannot cope with the background knowledge attack and homogeneity attack. Machanavajjhala et al. [29] firstly proposed l-diversity to improve k-anonymity. Each k-anonymity group in the published data sheet contains at least l different sensitive attribute values, so that the probability that an attacker infers a certain record privacy information will be less than $1/l$. In [30], considering the identity attributes of participants, it is ensured that each anonymous set at least has $k-1$ participants, and each anonymous set has p different sensitive values. In [31], k-anonymity and l-diversity were adopted as privacy models, and an anonymization method based on genetic algorithm clustering was proposed. The basic operator of genetic algorithm is improved to protect the personal sensitive information contained in the published report.

However, when requesting LBS services, the location of most participants is always related to time [32]. The above works only protect the location attribute of participants but do not associate the location attribute of participants with the time attribute. Trajectory anonymity refers to the sequence of user location information in a continuous period of time, which anonymizes and protects the user's location attribute and time attribute together. In [33], the trajectory privacy protection method based on user demand was proposed. By dividing different time intervals and setting different privacy protection parameters for different trajectories, the anonymous trajectory equivalence class is constructed. In [34], the Hilbert curve was used to extract the distribution characteristics of trajectory data each time, and the personalized differential privacy publishing mechanism was designed according to the individual needs for different degrees of privacy. In [35], a collaborative trajectory privacy protection scheme for continuous query was proposed to confuse attackers by issuing false query, thus confusing users' actual trajectory. In [36], a trajectory privacy protection algorithm based on trajectory shape diversity was proposed by combining k-anonymity and l-diversity to solve the trajectory privacy leakage problem that may be caused by the high similarity between trajectories in the anonymous set.

In the research of privacy protection data publishing (PPDP), the first proposed model was mainly used for static publishing, that is, only considering the one-time publishing of data, and the above research was mainly conducted for the static data publishing [37]. However, in many spatiotemporal crowdsourcing applications, a large amount of data stays in a changing state, and dynamic data publishing occurs from time to time [38]. In order to solve the problem that static publishing cannot resist link attacks and critical missing attacks, Wang and Fung [39] firstly studied the possible privacy leaks of data redistribution and proposed a method to prevent privacy leakage. The main idea of this method is to hide the true connection relationship between the two publishing versions, thereby weakening the global quasi-identifier. Xiao [40] firstly proposed the privacy protection model m-invariance for dynamic data publishing, whose key is to introduce pseudogeneralization technology to ensure that any QI group records in different data publishing versions have the same sensitive attribute value. In [41], because of the problem that the privacy protection association rule mining algorithm is not applicable to the dynamic change database, the incremental privacy protection data mining algorithm based on granularity calculation was proposed, and the incremental update algorithm was used to solve the problem of frequent item set calculation of incremental transaction database. In [42], a differential privacy histogram publishing method based on fractal dimension mining technology was proposed. The method used fractal dimension to cluster datasets and counted the values of each class. Laplace noise was added to data before publishing to achieve differential privacy. However, the above methods cannot cope with the privacy protection for spatiotemporal information, and it is difficult to adapt to the issue of dynamic privacy protection in spatiotemporal crowdsourcing applications. Even if the above methods consider participants' location information, attackers could also infer participants' privacy through time information. More importantly, the above methods are invalid for real-time data tasks.

Based on the above discussions, a dynamic publishing method for spatiotemporal privacy protection under the participatory sensing environment is proposed. By combining k-anonymity and l-diversity, the proposed dynamic publishing method could protect the privacy information of participants and reduce the information loss.

3. Dynamic Privacy Protection Algorithm

In this section, a dynamic privacy protection mechanism for spatiotemporal sensitive information is researched, and the working principles of the three main parts of the dynamic privacy protection mechanism are introduced. The proposed algorithms and corresponding explanations are given through an example.

3.1. Dynamic Privacy Protection Mechanism. The mechanism is divided into three parts: participants, TTPs, and application server.

(i) Participants: in spatiotemporal crowdsourcing applications, participants are responsible for the collection and uploading of sensing data [43]. Sensing

data uploaded by participant p_i, $1 \le i \le n$, and p_i include following attributes: $< \mathrm{id}_i, \mathrm{data}_i, \mathrm{time}_i, x_i, y_i >$, where id_i is the identity attribute of p_i, data_imeans completed sensing tasks uploaded by p_i, and $< \mathrm{time}_i, x_i, y_i >$ indicates real-time attribute and location attribute contained in data_i, denoted by d_i. It is a sensitive attribute and requires to anonymize. In the dynamic privacy protection publishing mechanism, participants submit sensing data in batches.

(ii) TTPs: in this mechanism, participants firstly upload sensing data to TTPs, and TTPs preprocesses the sensing data to extract sensitive data (i.e., participants' real spatiotemporal data) [44]. Then, using k-anonymity, the real spatiotemporal data are anonymized. The sensing data that do not satisfy the anonymity condition are stored in buffer pool and anonymize with the next incremental data. More importantly, when incremental data arrive, the corresponding equivalence classes will be added if the adaptive threshold is satisfied by utilizing the previous anonymity results. Finally, the cluster center value $\overline{u_i}, 1 \le i \le r$, is sent to the application server.

(iii) Application server: for avoiding the background knowledge attack and the homogeneity attack against k-anonymity, cluster center values need to be clustered again based on l-diversity idea. Application server anonymizes u_i according to the time attribute. Each cluster contains at least l cluster center values, and then the newly generated cluster center value $\overline{t_i}, 1 \le i \le c$, is published. After anonymization processing on TTPs and application server, the results are shown in Table 1, where $U = U_{i=1}^{r} M_i$, $M_i = U_{j=1}^{r_i} u_{ij}$, $L = U_{i=1}^{c} T_i$, and $T_i = U_{j=1}^{c_i} t_{ij}$, both r and c, respectively, represent the number of position clusters and time clusters.u_i represents a cluster containing spatiotemporal sensitive data. r_i and c_i represent the number of spatiotemporal sensitive data in the ith cluster. u_{ij} andt_{ij} represent the spatiotemporal sensitive data included in a cluster.

The dynamic publishing privacy protection mechanism proposed in this paper is different from the traditional spatiotemporal crowdsourcing process. In the process of traditional spatiotemporal crowdsourcing, requesters firstly publish tasks and then recruit participants to complete the task. In the process of uploading tasks by the participants, traditional spatiotemporal crowdsourcing does not consider the privacy of participants. More importantly, the static publish of tasks will reduce users' experience. The working process of the proposed dynamic publishing privacy protection mechanism is shown in Figure 1. In Step 1, participants send collected sensing data (including spatiotemporal sensitive information) to TTPs by secure wireless networks. In Step 2, TTPs reprocess the sensing data and extract spatiotemporal sensitive data. The spatiotemporal sensitive data are used by k-anonymity to anonymize. If the clustering condition is not met, Step 3 is performed to temporarily store the corresponding spatiotemporal sensitive data into buffer pool. If the clustering condition is met, Step 4 is performed, and TTP sends the anonymity result to the application server. In Step 5, application server clusters based on l-diversity for the time attribute of anonymity results. In Step 6, application server publishes the sensing data containing anonymity sensitive spatiotemporal data. In Step 7, when other participants submit sensing data, incremental data are sent to TTPs together with the sensing data temporarily stored in buffer pool. In Step 8, sensing data are dynamically anonymized by utilizing the previous anonymity results. Perform the above process until participants no longer submit sensing data.

3.2. Static Publishing Anonymous Protection. In order to protect the spatiotemporal privacy of participants, k-anonymization is used to anonymize participants' time and location attributes together. In the spatiotemporal crowdsourcing application, because of the different dimensions of time and location attributes of participants, we standardize the spatiotemporal sensitive data $d_i, 1 \le i \le n$, by using the standard deviation method expressed by equation (1). d_{ik} represents the real spatiotemporal information of the kth dimension of the ith data. d'_{ik} represents normalized spatiotemporal information of d_{ik} that is shown by the following equation:

$$d'_{ik} = \frac{d_{ik} - \min\{d_{jk}\}}{\max\{d_{jk}\} - \min\{d_{jk}\}}, \quad 10 \le j \le n, 1 \le k \le 3. \tag{1}$$

The distance between participants p_i and p_j is calculated by equation (2). The distance includes spatial distance and temporal distance between participants p_i and p_j:

$$\mathrm{dis}(p_i, p_j) = \sqrt{\sum_{k=1}^{3} \left(d'_{ik} - d'_{jk}\right)^2}, \quad 1 \le i, j \le n. \tag{2}$$

In order to easily find the center points of position cluster and time cluster, we calculate the global centroid $\overline{d}$ of the actual spatiotemporal dataset for anonymization by the following equation:

$$\overline{d} = < \frac{\sum_{i=1}^{n} \mathrm{time}_i}{n}, \frac{\sum_{i=1}^{n} x_i}{n}, \frac{\sum_{i=1}^{n} y_i}{n} > \tag{3}$$

In order to reduce the information loss and increase the privacy protection, we set the adaptive threshold expressed as follows:

$$\mathrm{ave}_j = \sum_{k=1}^{3} \sum_{i=1}^{r} \left(d'_{ik} - \overline{d'_{jk}}\right), \quad 1 \le j \le |G| \text{ or } |L|, \tag{4}$$

where r indicates that there is r spatiotemporal data in the cluster. The static publishing anonymity protection based on k-anonymity is shown in Algorithm 1.

Algorithm 1 describes that participants send sensing data to TTPs. The TTPs firstly process the sensing data and extract participant's real spatiotemporal information (represented by set A) as sensitive data for privacy

Table 1: Anonymization results.

Anonymization results	TTPs	Application server
Clustering result set	U	L
Each cluster in result set	$M_i, 1 \le i \le r$	$T_i, 1 \le i \le c$
Spatiotemporal sensitive data included in a cluster	$u_{ij}, 1 \le j \le r_i$	$t_{ij}, 1 \le j \le c_i$
Cluster center value	$\overline{u_i}, 1 \le i \le r$	$\overline{t_i}, 1 \le i \le c$

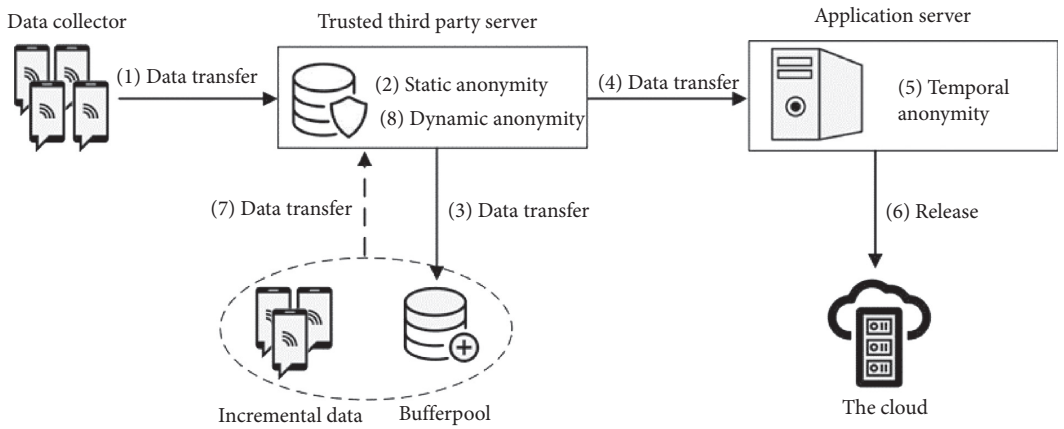

Figure 1: The framework of a dynamic publishing mechanism for privacy protection.

protection. The input of Algorithm 1 is k-anonymity-specified parameter k and the participant's real spatiotemporal dataset A. The output of Algorithm 1 is the anonymity result set U and buffer pool dataset B. Calculate the global centroid $\overline{d}$ in Step 1. Step 2 initializes parameters, and count is the number of new split clusters. Steps 4–11 describe that the number of points in the new split cluster is k. Step 5 selects a point d_{sma} with the largest distance to the global center point $\overline{d}$. d_{sma} indicates the new cluster center point and will be deleted from A (Step 6). Steps 7–11 select $k-1$points that have the smallest distance with $d_{\text{sma}}$to form a new cluster N_{count}. Update the center point d_{count} of N_{count} (Step 8), and select the point d_{smi} with the smallest distance to d_{count}(Step 9). Step 10 adds d_{smi} to cluster N_{count} and removes it from A. Steps 12–19 extend the cluster N_{count}, and in order to reduce information loss, we set the adaptive threshold ave (Step 13). If the point d_{cmi} (Step 14) in A satisfies the adaptive threshold, it will be added to the cluster N_{count} (Step 16), and the centroid d_{count} of N_{count} (Step 17) is updated. In Step 20, N_{count} is added to U, and the number of cluster is updated. If there are remaining data in A, it is stored in the buffer pool B (Step 22). In Step 23, the output of Algorithm 1 is returned.

The real spatiotemporal information contained in the sensing data uploaded by participants is anonymized by TTPs and returned to anonymity result set U. Send the center point $\overline{u_i}$ to the application server. Then, we illustrate the anonymity process of spatiotemporal data more visually by some data examples in experiments. As shown in Table 2, the first column is the class ID being run by Algorithm 1, the second and third columns are the real spatiotemporal information of participants, and the fourth and fifth columns are the anonymity spatiotemporal information. We can see that each equivalence class contains at least 3 points.

3.3. Improved Static Publishing Anonymity Protection Based on l-Diversity. However, k-anonymity is vulnerable to background knowledge attack and homogeneity attack. Therefore, when the application server publishes anonymity results, we adopt l-diversity to improve the algorithm. Application server receives the cluster center value u_i sent by TTPs, anonymizes the time attribute based on l-diversity, and calculates the time center value by the following equation:

$$\overline{t} = \frac{\sum_{i=1}^{m} \text{time}_i}{m}, \tag{5}$$

where m refers to the number of spatiotemporal data anonymized by TTPs, i.e., $m = \sum_{i=1}^{r} |M_i|$.

Algorithm 2 describes the anonymous releasing based on l-diversity for time attribute. The input is l-diversity parameter l and the output set U of Algorithm 1, and the output is anonymous set L. Step 1 and Step 2 take the time set T and the position set O out of U, respectively. Step 3 calculates the global central value $\overline{t}$ of time set T, and the number of initial clusters is count = 1. Steps 4–15 describe that the number of points in the newly generated cluster is l. Step 5 initializes time cluster T_{count} and location cluster O_{count}. Step 6 finds the t_{tma} with the largest distance to the global central value $\overline{t}$ by equation (2). Step 7 adds t_{tma} to the new cluster T_{count}, and the coordinate cluster O_{tma}corresponding to the subscript t_{ma} is added to the new cluster O_{count}. Then, t_{tma} is deleted from the time set T. The $l-1$ points with the smallest distance are selected to join the cluster (steps 8–12). Step 13 updates the time center

Input: k-anonymous parameter k, the actual spatiotemporal dataset A from participants
Output: aggregation result U, buffer pool dataset B

(1) Calculate the global centroid $\overline{d}$ of A by equation (3)
(2) count = 1, $U = \varphi$
(3) **while** $|A| \geq k$ do
(4) $N_{\text{count}} = \varphi$
(5) sma = $\text{argmax}_{i \in A} \text{dis}(d_i, \overline{d})$
(6) $N_{\text{count}} = N_{\text{count}} \cup d_{\text{sma}}$, $A = A/d_{\text{sma}}$
(7) **for** $j \longleftarrow 1$ to $k-1$ **do**
(8) Update the centroid $\overline{d_{\text{count}}}$ of N_{count} by equation (3)
(9) smi = $\text{argmin}_{i \in A} \text{dis}(d_i, \overline{d_{\text{count}}})$
(10) $N_{\text{count}} = N_{\text{count}} \cup d_{\text{smi}}$, $A = A/d_{\text{smi}}$
(11) **end for**
(12) **while** $|N_{\text{count}}| < 2k-1$ **do**
(13) Calculate the average distance $\text{ave}_{\text{count}}$ of N_{count} by equation (4)
(14) cmi = $\text{argmin}_{i \in A} \text{dis}(d_i, \overline{d_{\text{count}}})$
(15) **if** $\text{dis}(d_{\text{cmi}}, \overline{d_{\text{count}}}) < \text{ave}_{\text{count}}$ then
(16) $N_{\text{count}} = N_{\text{count}} \cup d_{\text{cmi}}$, $A = A/d_{\text{cmi}}$
(17) Update the centroid d_{count} of N_{count} by equation (3)
(18) **end if**
(19) **end while**
(20) $U = U \cup N_{\text{count}}$, count = count + 1
(21) **end while**
(22) $B = \text{A}$
(23) **return** U, B

Algorithm 1: Static publishing anonymity protection based on k-anonymity.

Table 2: Three anonymous examples.

Class ID	Time	Location	Anonymized time	Anonymized location
1	20:47:36	(21.3675, −157.9388)	20:38:22	(21.3166, −157.8616)
1	21:46:33	(21.2866, −157.8129)	20:38:22	(21.3166, −157.8616)
1	19:20:57	(21.2958, −157.8331)	20:38:22	(21.3166, −157.8616)
2	23:31:00	(45.5894, −122.7524)	23:00:57	(46.3272, −122.5448)
2	22:00:00	(45.7801, −122.5400)	23:00:57	(46.3272, −122.5448)
2	23:31:50	(47.6122, −122.3419)	23:00:57	(46.3272, −122.5448)
3	18:54:05	(30.4810, −97.8295)	19:15:09	(32.0273, −97.4996)
3	19:33:26	(32.7368, −97.3271)	19:15:09	(32.0273, −97.4996)
3	19:17:48	(32.8640, −97.3421)	19:15:09	(32.0273, −97.4996)
4	18:46:48	(30.2016, −97.6671)	19:02:11	(31.5155, −97.4498)
4	19:09:06	(32.6804, −97.3746)	19:02:11	(31.5155, −97.4498)
4	19:03:47	(32.8382, −97.0045)	19:02:11	(31.5155, −97.4498)
4	19:09:03	(30.3417, −97.7530)	19:02:11	(31.5155, −97.4498)
5	19:50:21	(59.3238, 18.0977)	17:25:48	(59.3232, 18.0543)
5	15:49:08	(59.3457, 18.0587)	17:25:48	(59.3232, 18.0543)
5	16:38:04	(59.3055, 17.9892)	17:25:48	(59.3232, 18.0543)
5	17:28:22	(59.3122, 18.0796)	17:25:48	(59.3232, 18.0543)
5	17:23:05	(59.3288, 18.0461)	17:25:48	(59.3232, 18.0543)

value $\overline{t_{\text{count}}}$ of cluster T_{count}. The output of Algorithm 2 in Step 14 is L_{count}, and the Cartesian product of time center value sets $\overline{t_{\text{count}}}$ and position cluster O_{count}. Steps 16–23 describe that if there is any remaining point in time set T, the cluster with the smallest distance (Step 18) is found by equation (2) and added to the cluster (Step 19), then the time center value $\overline{t_{\text{lmi}}}$ of the cluster T_{lmi} is updated (Step 20). In Step 24, the output of Algorithm 2 is returned.

The following is a more visual illustration of releasing spatiotemporal data based on l-diversity. As shown in Table 3, the first column is the group ID being run by Algorithm 2, the second column is the class ID being run by Algorithm 1, the third and fourth columns are the anonymous spatiotemporal information being run by Algorithm 1, and the fifth and sixth columns are the anonymous spatiotemporal information anonymized by the application server. We can see that each 2-equivalence group ($l = 2$) contains at least two 3-equivalence classes ($k = 3$), where the anonymous time attribute is the same and the anonymous location attribute is different.

Input: l-diversity parameter l, aggregation result U from Algorithm 1
Output: aggregation result L

(1) Take time set T out of U
(2) Take location set O out of U
(3) Calculate the global centroid $\overline{t}$ by equation (5), count = 1
(4) **while** $|T| \geq l$**do**
(5) $T_{\text{count}} = \varphi$, $O_{\text{count}} = \varphi$
(6) tma = $\text{argmax}_{i \in T} \text{dis}(t_i, \overline{t})$
(7) $T_{\text{count}} = T_{\text{count}} \cup t_{\text{tma}}$, $O_{\text{count}} = O_{\text{count}} \cup O_{\text{tma}}$, $T = T/t_{\text{tma}}$
(8) **for** $j \longleftarrow 1$ to $l - 1$ **do**
(9) Update the centroid $\overline{t_{\text{count}}}$ of T_{count} by equation (3)
(10) tmi = $\text{argmin}_{i \in T} \text{dis}(t_i, \overline{t_{\text{count}}})$
(11) $T_{\text{count}} = T_{\text{count}} \cup t_{\text{tmi}}$, $O_{\text{count}} = O_{\text{count}} \cup O_{\text{tmi}}$, $T = T/t_{\text{tmi}}$
(12) **end for**
(13) Update the centroid $\overline{t_{\text{count}}}$ of T_{count} by equation (3)
(14) $L_{\text{count}} = \overline{t_{\text{count}}} \times O_{\text{count}}$, count = count + 1
(15) **end while**
(16) **while** $|T| > 0$**do**
(17) **for** $i \in |T|$**do**
(18) lmi = $\text{argmin}_{j \in L} \text{dis}(t_i, \overline{t_j})$
(19) $T_{\text{lmi}} = T_{\text{lmi}} \cup t_i$, $O_{\text{lmi}} = O_{\text{lmi}} \cup O_i$
(20) Update the centroid $\overline{t_{\text{lmi}}}$**of** T_{lmi}
(21) $L_{\text{lmi}} = \overline{t_{\text{lmi}}} \times O_{\text{lmi}}$
(22) **end for**
(23) **end while**
(24) **return** L

Algorithm 2: Static publishing anonymity protection based on l-diversity.

Table 3: 3-Anonymity, 2-diversity examples.

Group ID	Class ID	Time	Location	Anonymized time	Anonymized location
1	3	19:15:09	(32.0273, −97.4996)	18:34:23	(32.0273, −97.4996)
1	3	19:15:09	(32.0273, −97.4996)	18:34:23	(32.0273, −97.4996)
1	3	19:15:09	(32.0273, −97.4996)	18:34:23	(32.0273, −97.4996)
1	4	19:02:11	(31.5155, −97.4498)	18:34:23	(31.5155, −97.4498)
1	4	19:02:11	(31.5155, −97.4498)	18:34:23	(31.5155, −97.4498)
1	4	19:02:11	(31.5155, −97.4498)	18:34:23	(31.5155, −97.4498)
1	4	19:02:11	(31.5155, −97.4498)	18:34:23	(31.5155, −97.4498)
1	5	17:25:48	(59.3232, 18.0543)	18:34:23	(59.3232, 18.0543)
1	5	17:25:48	(59.3232, 18.0543)	18:34:23	(59.3232, 18.0543)
1	5	17:25:48	(59.3232, 18.0543)	18:34:23	(59.3232, 18.0543)
1	5	17:25:48	(59.3232, 18.0543)	18:34:23	(59.3232, 18.0543)
1	5	17:25:48	(59.3232, 18.0543)	18:34:23	(59.3232, 18.0543)
2	1	20:38:22	(21.3166, −157.8616)	21:49:40	(21.3166, −157.8616)
2	1	20:38:22	(21.3166, −157.8616)	21:49:40	(21.3166, −157.8616)
2	1	20:38:22	(21.3166, −157.8616)	21:49:40	(21.3166, −157.8616)
2	2	23:00:57	(46.3272, −122.5448)	21:49:40	(46.3272, −122.5448)
2	2	23:00:57	(46.3272, −122.5448)	21:49:40	(46.3272, −122.5448)
2	2	23:00:57	(46.3272, −122.5448)	21:49:40	(46.3272, −122.5448)

3.4. Dynamic Publishing Anonymity Protection. For static one-release mechanisms, k-anonymity and l-diversity are valid. However, in real life, application servers usually publish sensing data dynamically. Therefore, in this section, we improve k-anonymity and l-diversity to accommodate dynamic publishing mechanism. Algorithm 3 describes the dynamic publishing anonymity protection.

Algorithm 3 describes how TTPs use the previous anonymity result to solve the problem of dynamic publishing when participants submit sensing data in different time periods. The input of Algorithm 3 is k-anonymous parameter k, the clustering result U of Algorithm 1, incremental dataset I (that is, the sensing data submitted by participants), and buffer pool dataset B. The output of the algorithm is the clustering result D and buffer pool dataset

Input: k-anonymous parameter k, aggregation result U from Algorithm 1, incremental dataset I, buffer pool dataset B
Output: aggregation result D, buffer pool dataset B'

(1) Calculate global dataset $W=I+B$
(2) **for** $i \longleftarrow 1$ to r **do**
(3) Take the centroid set $\overline{U}$ out of U
(4) smi = argmin$_{j\in\overline{U}}$dis$(\overline{u_j}, tW_i)$
(5) r ave = argmin$_{e\in|u_{\text{smi}}|}$dis$(u_e, \overline{u_{\text{smi}}})$
(6) **if** dis$(\overline{u_{\text{smi}}}, tW_i) \le r$ ave **then**
(7) $M_{\text{smi}} = M_{\text{smi}} \cup W_i$, $W = W/W_i$
(8) Update the centroid $\overline{u_{\text{smi}}}$ of u_{smi}
(9) $D = U$
(10) **end if**
(11) **end for**
(12) **for** $i \longleftarrow 1$to r **do**
(13) **if** $|M_i| > 2k$ **then**
(14) **Callback Algorithm** 4$\longrightarrow$
(15) **input**: k-anonymous parameter k, cluster M
(16) **output**: aggregation result G
(17) **end if**
(18) $D = U \cup G$, $B' = W$
(19) **end for**
(20) **return** D, B'

Algorithm 3: Dynamic publishing anonymous protection.

B'. The global dataset W is the incremental data I and the buffer pool data B (Step 1). Steps 2–11 describe the process of adding data from dataset W that meets the adaptive threshold condition to the last clustering result, where r represents the number of clusters of U (Table 1). First, the cluster center set $\overline{U} = U_{i=1}^{r}\overline{u_i}$ in the clustering result U is taken out (Step 3), and Step 4 finds the subscript of the smallest cluster center point smi to point W_i. Then, adaptive threshold values r ave are set by equation (4) (Steps 5 and 6), where r ave is the average distance between point u_e and center point $\overline{u_{\text{smi}}}$ in clusteru_{smi}. If the point in dataset W meets the adaptive threshold, join the corresponding cluster and delete the point from W (Step 7), update the central value $\overline{u_{\text{smi}}}$ of cluster u_{smi} (Step 8), and assign the updated U to the output result D in Algorithm 3 (Step 9). Steps 12–19 describe that if the number of points in cluster M_i is greater than or equal to $2k$, then Algorithm 4 is called to split M_i. The clustering result D is denoted by $D = U \cup G$, where U is the number of points in cluster $M_i$less than $2k$, and G is the output of Algorithm 4 and temporarily stores the remaining data in W to buffer pool B'(Step 18). In Step 20, the output of Algorithm 3 is returned.

Algorithm 4 describes that if the number of points in the cluster is greater than or equal to $2k$, the cluster is split. The input of Algorithm 4 is k-anonymous parameter k and cluster M. The output of Algorithm 4 is the clustering result G of the new split. Step 1 calculates the global center point $\overline{d}$ of cluster M by equation (3). Step 2 initializes parameters, count is the number of new split clusters, and G is the output of Algorithm 4. Steps 3–13 describe that the number of points in the new split cluster is k. Step 5 selects a point d_{sma} with the largest distance to the global center point $\overline{d}$. Take d_{sma} as the new cluster center point and delete it from M (Step 6). Steps 7–11 select k-1 points that have the smallest distance with d_{sma} to form a new clusterN_{count}, update the center point $\overline{d_{\text{count}}}$of N_{count} (Step 8), and select the point $d_{\text{smi}}$with the smallest distance to $\overline{d_{\text{count}}}$ (Step 9). Step 10 adds d_{smi} to cluster N_{count} and removes it from M. In Step 12, the newly generated cluster N_{count} is added to the output result G, and the number of clusters increases. If there are remaining points in cluster M, add them to the new cluster closest to them (steps 14–20). Step 16 finds a cluster N_{cmi} having the smallest distance with the remaining point d_i, add $d_i$to N_{cmi} (Step 17), and update the center point $\overline{d_{\text{cmi}}}$ of cluster N_{cmi}. In Step 21, the output G of Algorithm 4 is returned.

4. Experiments and Result Analysis

In this section, we use real-world datasets, including Gowalla's Friendship Network dataset and Kaggle's New York Taxi Travel Time dataset. Table 4 shows the number of attributes and data points and the density of data points contained in datasets. We compare the proposed STPP algorithm with k-anonymity and VCLA algorithms in terms of running time, information loss, and privacy protection. The hardware environment of the experiments is an AMD A8-5550M APU with Radeon (tm) HD Graphics @ 2.10 GHz equipped with 4 GB RAM and running the Win 10 OS.

Datasets are processed to better protect participants' spatiotemporal sensitive data. First, we randomly extract 1000 data from Friendship Network dataset as a segment, a total of five segments, as participant's sensing data to conduct comparison experiments. Then, we randomly extract 3000 data from New York City Taxi Trip dataset as a

Input: k-anonymous parameter k, cluster M
Output: aggregation result G
(1) Calculate the global centroid $\overline{d}$ of cluster M by equation (3)
(2) count = 1, $G = \varphi$
(3) **while** $|M| \geq k$ **do**
(4) $N_{\text{count}} = \varphi$
(5) sma = $\text{argmax}_{i \in M} \text{dis}(d_i, \overline{d})$
(6) $N_{\text{count}} = N_{\text{count}} \cup d_{\text{sma}}$, $M = M/d_{\text{sma}}$
(7) **for** $j \longleftarrow 1$ to $k-1$ **do**
(8) Update the centroid $\overline{d_{\text{count}}}$ of N_{count}
(9) smi = $\text{argmin}_{i \in M} \text{dis}(d_i, \overline{d_{\text{count}}})$
(10) $N_{\text{count}} = N_{\text{count}} \cup d_{\text{smi}}$, $M = M/d_{\text{smi}}$
(11) **end for**
(12) $G = G \cup N_{\text{count}}$, count = count + 1
(13) **end while**
(14) **while** $|M| > 0$**do**
(15) **for** $i \in |M|$**do**
(16) cmi = $\text{argmin}_{i \in G} \text{dis}(d_i, \overline{d_j})$
(17) $N_{\text{cma}} = N_{\text{cmi}} \cup d_i$
(18) Update the centroid $\overline{d_{\text{cmi}}}$ of N_{cmi}
(19) **end for**
(20) **end while**
(21) **return** G

Algorithm 4: Breaking up clusters.

Table 4: Attribute, quantity, and density of datasets.

Datasets	Dimensions	Quantity	Sparseness
Friendship Network dataset	5	6442892	Sparse
New York City Taxi Trip dataset	11	1458644	Dense

segment, a total of five segments, as participant's sensing data to design comparison experiments. Each segment of sensing data is uploaded to TTPs in batches dynamically. Then, the spatiotemporal sensitive data of participants, including time and location attributes, are extracted from sensing data for anonymization.

Figure 2 shows the comparison of experimental results by comparing the proposed STPP algorithm with k-anonymous and VCLA algorithms on running time. Figure 2(a) shows the experimental result on Friendship Network dataset, and Figure 2(b) shows the experimental result on New York City Taxi Trip dataset. The x-coordinate is the number of participants, and the y-coordinate is running time. It can be seen that the STPP algorithm is superior to the other two algorithms, whether it is on a small dataset where participants' spatiotemporal distance is sparse, or on a large dataset with dense spatiotemporal distance. When there are fewer participants submitting tasks, the running time of the three algorithms is not much different. It is because that the three algorithms are improved by k-anonymity algorithms, the STPP algorithm proposed in this paper does not have obvious advantages in terms of running time when there are few participants. However, when the number of participants gradually increases, STPP algorithm could better solve the problem of poor timeliness of data publishing due to the large number of participants in spatiotemporal crowdsourcing applications.

Since anonymized data are used for dynamic publishing, the difference between real spatiotemporal data and anonymized data is seen as the information loss. The information loss is expressed by the following equation:

$$\text{IL} = \sum_{i=1}^{n} \sum_{k=1}^{3} d'_{ik} - \overline{d'_{ik}}, \tag{6}$$

where $\overline{d'_{ik}}$ represents the anonymized information of d'_{ik}. k represents dimension, which includes time dimension and location dimension.

Figure 3 shows the comparison of experimental result by comparing the STPP algorithm with k-anonymous and VCLA algorithm on information loss. The x-coordinate is the number of participants on the Friendship Network dataset, and the y-coordinate is information loss. From the experimental result, it can be seen that the information loss increases with the increase of participants. Moreover, STPP algorithm is obviously better than the comparison algorithms on information loss.

Figure 4 shows the relationship between the parameter k of k-anonymous and the information loss of the STPP algorithm, where different curves represent different amounts of data. The experiments are conducted on the Friendship Network dataset. From the experimental result, it can be inferred that with the increase of k, the information loss increases gradually, which is because that increasing k leads

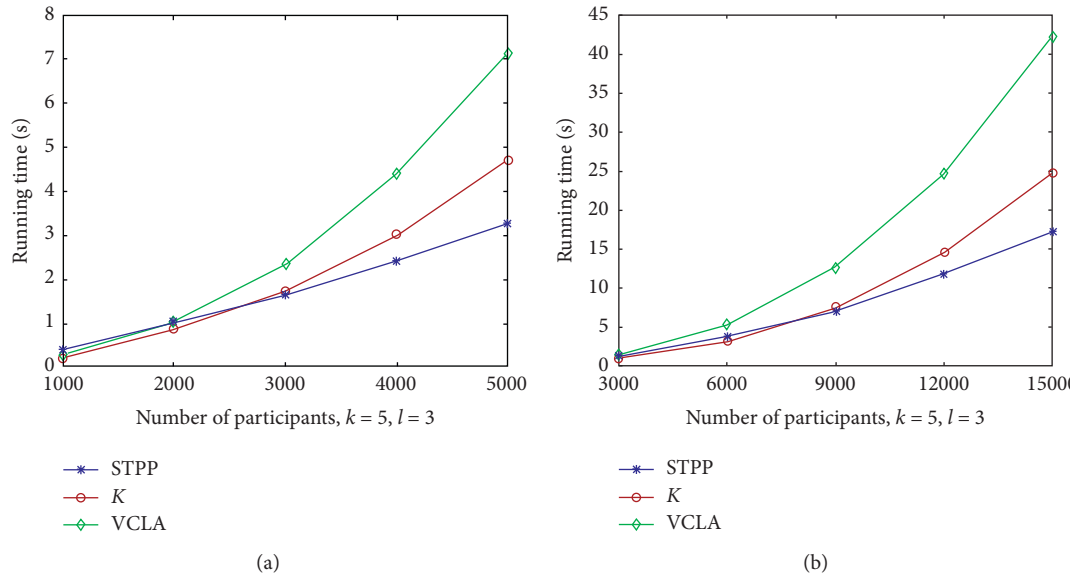

Figure 2: Comparison of experimental results of running time on (a) Friendship Network dataset and (b) New York City Taxi Trip dataset.

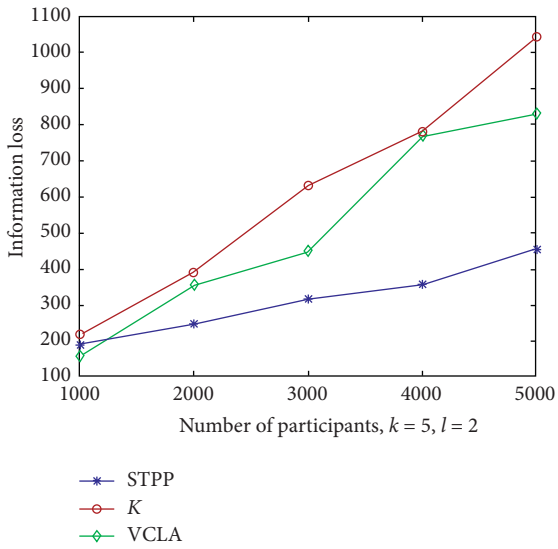

Figure 3: Comparison of experimental result on information loss.

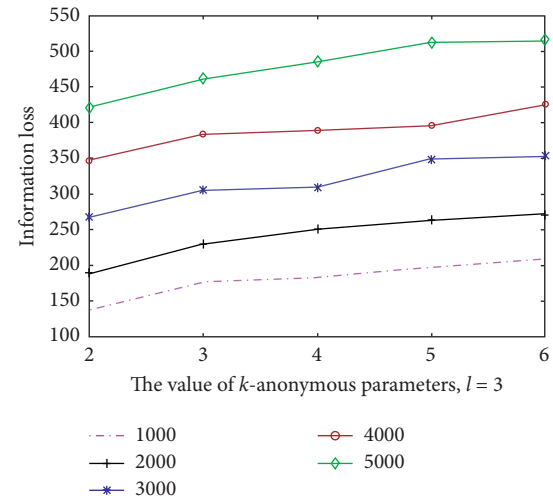

Figure 4: Information loss of STPP algorithm.

to an increase of spatiotemporal sensitive data in clusters, and IL in each cluster will increase correspondingly.

For evaluating the performance of privacy protection, we use the probability of attackers' attack success to quantify and compare, that is, attackers guess the probability of participants' specific spatiotemporal data based on the published sensing data. Suppose that n sensing data are published, and spatiotemporal sensitive data $d_i, 1 \le i \le n$, are aggregated into r location clusters and c time clusters. In this paper, equation 7 is used to quantify privacy protection, where $\sum_{i=1}^{r}(1/|u_i|)/r$ and $\sum_{j=1}^{c}(1/|c_j|)/c$ represent the average probability that attackers can infer real location attribute and time attribute of each sensing data, respectively:

$$p = \frac{1}{n} \times \frac{\sum_{i=1}^{r}\left(1/\left|u_i\right|\right)}{r} \times \frac{\sum_{j=1}^{c}\left(1/\left|c_j\right|\right)}{c}. \tag{7}$$

Figure 5 shows the comparison of experimental result by comparing the STPP algorithm with k-anonymous and VCLA algorithms on privacy protection. The x-coordinate is the number of participants on the New York City Taxi Trip dataset, and the y-coordinate is the probability of attackers to infer specific spatiotemporal data of participants based on the published sensing data. From the experimental result, it can be seen that the privacy protection gradually increases with the increase of participants. It is because that if the number of participants increases, the sensing data published by the application server will increase correspondingly, which reduces the probability of attackers' attack success, since the probability of successful attack without

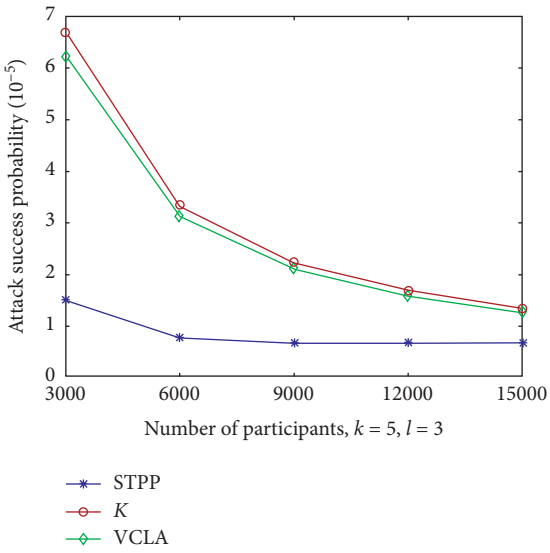

Figure 5: Experimental result on privacy protection.

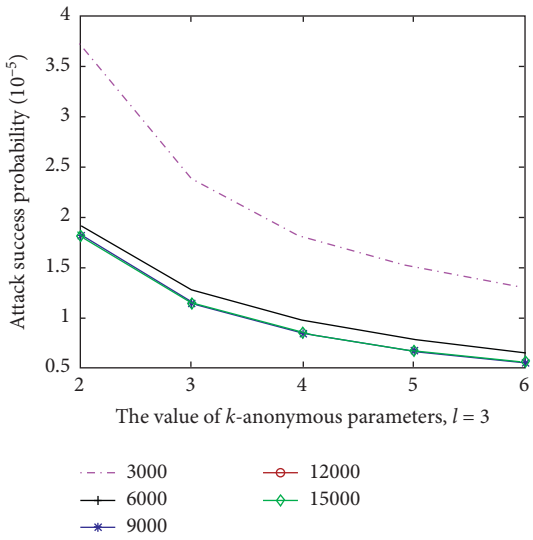

Figure 6: Privacy protection of the STPP algorithm.

background knowledge is very low (y-coordinate unit is 10^{-5}). STPP algorithm is slightly better than the comparison algorithms on privacy protection.

Figure 6 shows the relationship between the parameter k of k-anonymous and the privacy protection of the STPP algorithm. We conduct the experiments on New York City Taxi Trip dataset. From the experimental result, we can see that with the increase of k, the privacy protection increases gradually, which is because that increasing k leads to an increase of spatiotemporal sensitive data in each cluster, and the average probability is reduced that the real spatiotemporal data are inferred by attackers.

When participants upload sensing data, TTPs will temporarily store sensing data that do not meet anonymity condition into buffer pool. Then, TTPs wait for the arrival of the next incremental data, which will generate the problem of delayed publish of sensing data. Figure 7 shows the ratio of

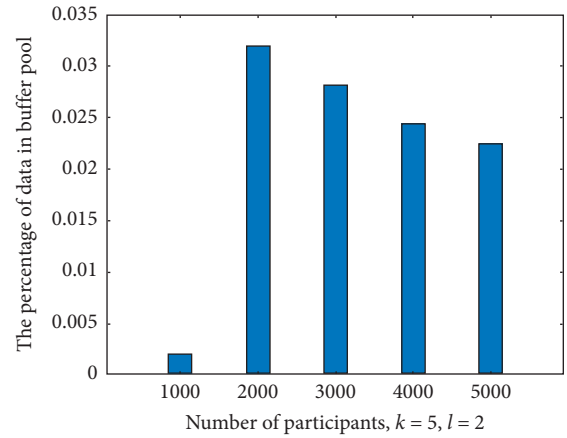

Figure 7: The proportion of sensing data in buffer pool.

buffer pool data to the number of sensing data for this publish. The x-coordinate is the number of participants on the Friendship Network dataset, and the y-coordinate is the ratio of sensing data in buffer pool. It can be seen that the proportion of data in buffer pool is very low, which proves that the sensing data in buffer pool have no great impact on delayed publish.

Through experiments on real-world datasets, we can see that the proposed STPP algorithm is superior to k-anonymous and VCLA algorithms in terms of running time, information loss, and privacy protection. STPP algorithm could solve the privacy protection problem of dynamic publishing for spatiotemporal crowdsourcing.

5. Conclusions

In the existing work, few researchers focus on privacy protection for dynamic publishing mechanism. There are few privacy protection methods for spatiotemporal sensitive data in dynamic publishing mechanism. In this paper, a dynamic publishing mechanism for spatiotemporal sensitive data privacy protection is proposed. Then, we design the dynamic k-anonymity algorithm and add the spatiotemporal data that met the adaptive threshold condition to the corresponding equivalence classes, making full use of the previous anonymous result to solve the problem of poor timeliness of static publishing. Thirdly, aiming at the shortcomings of k-anonymity, which is vulnerable to background knowledge attacks and homogeneous attacks, we anonymize participants' time attribute based on l-diversity, so as to improve privacy protection and reduce information loss. Finally, the performance of the proposed STPP algorithm is evaluated on two real-world datasets. Compared with the existing algorithms, experimental results show that STPP algorithm has lower time complexity, less information loss, and stronger privacy protection.

In the future, we will detect and process malicious participants (i.e., outliers) so as to better reduce information loss and protect participants' privacy data.

Data Availability

The experiment data used to support the findings of this study have been deposited in the GitHub repository (https://github.com/ltn21999/K_L-dynamic-privacy-protection).

Conflicts of Interest

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant nos. 61822602, 61772207, 61802331, 61572418, 61602399, 61702439, and 61773331, the China Postdoctoral Science Foundation under Grant nos. 2019T120732 and 2017M622691, the National Science Foundation (NSF) under Grant nos. 1704287, 1252292, and 1741277, and the Graduate Innovation Foundation of Yantai University (GIFYTU) under Grant nos. YDYB2024 and YDZD1908.

References

[1] Z. Cai, X. Zheng, and J. Yu, "A differential-private framework for urban traffic flows estimation via taxi companies," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6492–6499, 2019.

[2] J. Li, T. Cai, K. Deng, X. Wang, T. Sellis, and F. Xia, "Community-diversified influence maximization in social networks," *Information Systems*, vol. 92, pp. 1–12, 2020.

[3] Y. Wang, Z. Cai, Z.-H. Zhan, Y.-J. Gong, and X. Tong, "An optimization and auction-based incentive mechanism to maximize social welfare for mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 3, pp. 414–429, 2019.

[4] N. A. H. Haldar, J. Li, M. Reynolds, T. Sellis, and J. X. Yu, "Location prediction in large-scale social networks: an in-depth benchmarking study," *VLDB Journal*, vol. 28, no. 5, pp. 623–648, 2019.

[5] J. Wang, Z. Cai, and J. Yu, "Achieving personalized k-Anonymity-Based content privacy for autonomous vehicles in CPS," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4242–4251, 2020.

[6] Z. Xiong, W. Li, Q. Han et al., "Privacy-preserving auto-driving: a GAN-based approach to protect vehicular camera data," in *Proceedings of 2019 IEEE International Conference on Data Mining (ICDM)*, Beijing, China, November 2019.

[7] M. Bi, Y. Wang, Y. Li, and X. Tong, *A Privacy-Preserving Mechanism Based on Local Differential Privacy in Edge Computing*, China Communications, Hong Kong, China, 2020.

[8] Y. Liang, Z. Cai, J. Yu, Q. Han, and Y. Li, "Deep learning based inference of private information using embedded sensors in smart devices," *IEEE Network*, vol. 32, no. 4, pp. 8–14, 2018.

[9] X. Xu, X. Zhang, X. Liu, J. Jiang, L. Qi, and M. Z. A. Bhuiyan, "Adaptive computation offloading with edge for 5G-envisioned internet of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[10] H. To, G. Ghinita, and C. Shahabi, "A framework for protecting worker location privacy in spatial crowdsourcing," *Proceedings of the VLDB Endowment*, vol. 7, no. 10, pp. 919–930, 2014.

[11] K. Vu, R. Zheng, and J. Gao, "Efficient algorithms for k-anonymous location privacy in participatory sensing," in *Proceeding of the IEEE INFOCOM*, pp. 2399–2407, Orlando, FL, USA, March 2012.

[12] S. B. Avaghade and S. S. Patil, "Privacy preserving for spatio-temporal data publishing ensuring location diversity using K-anonymity technique," in *Proceedings of the 2015 International Conference on Computer, Communication and Control (IC4)*, September 2015.

[13] Z. Cai, Z. Duan, and W. Li, "Exploiting multi-dimensional task diversity in distributed auctions for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, no. 99, p. 1, 2020.

[14] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Computer Networks*, vol. 171, pp. 1–14, 2020.

[15] T. Liu, Y. Wang, Y. Li, X. Tong, L. Qi, and N. Jiang, "Privacy protection based on stream cipher for spatio-temporal data in IoT," *IEEE Internet of Things Journal*, 2020.

[16] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *Proceedings of the 39th IEEE International Conference on Distributed Computing Systems (ICDCS*, Dallas, TX, USA, July 2019.

[17] X. Wang, Z. Liu, X. Tian et al., "Incentivizing crowdsensing with location-privacy preserving," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6940–6952, 2017.

[18] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering (TNSE)*, vol. 7, no. 2, pp. 766–775, 2020.

[19] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. S. Shen, "Energy efficient dynamic offloading in mobile edge computing for internet of things," *IEEE Transactions on Cloud Computing*, 2019.

[20] X. Xu, C. He, Z. Xu, L. Qi, S. Wan, and M. Z. A. Bhuiyan, "Joint optimization of offloading utility and privacy for edge computing enabled IoT," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2622–2629, 2020.

[21] X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, and W. Dou, "BeCome: blockchain-enabled computation offloading for IoT in mobile edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4187–4195, 2020.

[22] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[23] Y. Wang, Z. Cai, X. Tong, Y. Gao, and G. Yin, "Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems," *Computer Networks*, vol. 135, pp. 32–43, 2018.

[24] P. Samarati, "Protecting respondents identities in microdata release," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[25] L. Fang and L. Tong, "A clustering K-anonymity privacy-preserving method for wearable iot devices," *Security and Communication Networks*, vol. 2018, Article ID 4945152, 8 pages, 2018.

[26] S. C. Lin, A. Y. Ye, and L. Xu, "K-anonymity location privacy protection method with coordinate transformation," *Journal of Chinese Computer Systems*, vol. 37, pp. 119–123, 2016.

[27] X. Zheng, Z. Cai, and Y. Li, "Data linkage in smart internet of things systems: a consideration from a privacy perspective," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 55–61, 2018.

[28] H. Wang, H. Huang, Y. Qin, Y. Wang, and M. Wu, "Efficient location privacy-preserving K-anonymity method based on the credible chain," *ISPRS International Journal of Geo-Information*, vol. 6, no. 6, p. 163, 2017.

[29] A. Machanavajjhala, J. Gehrke, D. Kifer et al., "L-diversity: privacy beyond k-anonymity," in *Proceedings of the 22nd International Conference on Data Engineering*, April 2006.

[30] T. Dargahi, M. Ambrosin, M. Conti, and N. Asokan, "ABAKA: a novel attribute-based k-anonymous collaborative solution for LBSs," *Computer Communications*, vol. 85, pp. 1–13, 2016.

[31] A. Abdrashitov and A. Spivak, "Sensor data anonymization based on genetic algorithm clustering with L-Diversity," in *Proceedings of the 18th Conference of Open Innovations Association and Seminar on Information Security and Protection of Information Technology (FRUCT-ISPIT)*, April 2016.

[32] Y. Wang, Z. Cai, G. Yin, Y. Gao, X. Tong, and G. Wu, "An incentive mechanism with privacy protection in mobile crowdsourcing systems," *Computer Networks*, vol. 102, pp. 157–171, 2016.

[33] Z. Hu, J. Yang, and J. Zhang, "Personalized trajectory privacy protection method based on user-requirement," *International Journal of Cooperative Information Systems*, vol. 27, no. 3, 2018.

[34] F. Tian, S. Zhang, L. Lu et al., "A novel personalized differential privacy mechanism for trajectory data publication," in *2017 Proceedings of the International Conference on Networking & Network Applications (NaNA)*, October 2017.

[35] T. Peng, Q. Liu, D. Meng et al., "Collaborative trajectory privacy preserving scheme in location-based services," *Information Sciences*, vol. 387, pp. 165–179, 2017.

[36] D. Sun, Y. Luo, G. Fan et al., "Privacy protection algorithm based on trajectory shape diversity," *Journal of Computer Applications*, vol. 36, no. 6, pp. 1544–1551, 2016.

[37] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but No track: privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1868–1878, 2017.

[38] X. Xu, B. Shen, X. Yin et al., "Edge server quantification and placement for offloading social media services in industrial cognitive IoV," *IEEE Transactions on Industrial Informatics*, no. 99, p. 1, 2020.

[39] K. Wang and B. C. M. Fung, "Anonymizing sequential releases," in *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Philadelphia, PA, USAACM, Philadelphia, PA, USA, August 2006.

[40] X. Xiao, "M-invariance: towards privacy preserving re-publication of dynamic datasets," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Beijing, China, June 2007.

[41] S. Cheng, C. Xu, and H. Dan, "Research on incremental privacy preserving data mining," *Application Research of Computers*, vol. 3, no. 8, 2018.

[42] F. Yan, X. Zhang, C. Li et al., "Differentially private histogram publishing through Fractal dimension for dynamic datasets," in *Proceedings of IEEE 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1542–1546, Wuhan, China, June 2018.

[43] Y. Wang, Z. Cai, Z. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, 2020.

[44] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 665–673, 2018.