IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON HUMAN-DRIVEN EDGE COMPUTING (HEC)

Received August 31, 2020, accepted September 19, 2020, date of publication September 28, 2020, date of current version October 7, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3027299*

# Privacy and Utility Preserving Trajectory Data Publishing for Intelligent Transportation Systems

**XIANGWEN LIU AND YUQUAN ZHU**
School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China

Corresponding author: Xiangwen Liu (liuxw@ujs.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant U1736216, Grant 61702233, and Grant 61702229; in part by the Graduate Student Scientific Research Innovation Project of Jiangsu Province under Grant KYCX17_1809; in part by the Project funded by the China Postdoctoral Science Foundation under Grant 2019M651738; and in part by the Innovation and Entrepreneurship Training Program of Jiangsu University under Grant 201910299506x.

**ABSTRACT** Nowadays, the extensive collection and storage of massive personal GPS data in intelligent transportation systems every day provide great convenience for trajectory data analysis and mining research, thus bringing valuable information for real-life applications. Yet, protecting personal privacy is also more challenging in the smart environment. When trajectories of individuals are published together with their sensitive attributes such as disease, income etc., one can use partial trajectory knowledge for identity, sensitive locations, and sensitive values of target individuals. We present $(\alpha, K)_L$-privacy model and an anonymization scheme aimed at *I*dentifying and *E*liminating *V*iolating privacy *S*ubtrajectories (IEVS), to prevent privacy disclosure while preserving the accuracy and high quality of published trajectories. In particular, IEVS employs three anonymization techniques, i.e., trajectory splitting, location suppression, and sensitive value generalization to eliminate all subtrajectories violating $(\alpha, K)_L$-privacy principle. Experiments show our scheme is effective to improve the data utility of anonymized trajectories when compared with previous work.

**INDEX TERMS** Privacy preservation, trajectory data publishing, splitting, generalization, ITS.

## I. INTRODUCTION

The rapid development of the Internet of things (IoT) and big data technology has spawned many new smart application domains for the urban environment. The massive amount of information collected in IoT is shared across assorted platforms and applications to predict the planning and development of cities, thus accelerating the construction process of smart cities [1]. The intelligent transportation system (ITS) [2], for example, one of the important application domains in smart cities, generates great amounts of real-time GPS data every day. Such abundant spatiotemporal information, organized as trajectories, reflects the historical traffic conditions of a city. Analyzing and mining behavior patterns from trajectories via big data technology can support decision-making for urban planning and development, such as improving traffic congestion, optimizing freight movements, facilitating residents' travel, etc. [3], to make cities smarter.

Sharing traffic trajectories for data mining brings many advantages to multiple applications in real life. However, the release of raw trajectories may pose serious privacy concerns even though the identity information of trajectory users is removed, especially when these trajectories are cross-referenced with the explicitly published spatiotemporal data from users through various social networking services in smart cities. For instance, an adversary sees that the target individual posts the information of the gourmet shops he visited on the social network platform one day. If a published trajectory dataset includes all the trajectory information of this day, and there exists a trajectory containing the location-time data of the target's visit to the gourmet shops, the adversary can associate this trajectory with the target accurately. He can further obtain the target's sensitive information contained in the trajectory, such as the home address, travel habit, health condition, personal interest, etc. Moreover, the adversary may

The associate editor coordinating the review of this manuscript and approving it for publication was Lu Liu.

infer the sensitive values via trajectories if trajectory data is published with other sensitive attributes of an individual, which poses more serious privacy disclosure.

In the era of big data, adversaries can easily get more and more spatiotemporal data published by trajectory users through location-based social networking sites, so trajectory data is more likely to expose individuals' privacy. To cope with the problem, trajectory $k$-anonymization, an attack tolerant principle, is presented to constrain the identification probability of privacy-related information. Although many approaches based on $k$-anonymity have been proposed to protect location and trajectory privacy, most of them don't explicitly discuss data quality [4]. As a result, the published trajectories are useless for data analysis and mining [5]. In the widely employed $k$-anonymization techniques for trajectory publishing, grouping $k$ co-localized trajectories and generalizing them to form a $k$-anonymized aggregate trajectory changes the entire trajectory footprint, while over distorting trajectory data. Generalizing $k$ sampling points or $k$ separated trajectory segments in different trajectories [6], [7] for the $k$-anonymization of local areas or partial trajectories cannot avoid privacy disclosure when the adversary holds some location-time data of the target in different generalized segments or areas. Moreover, generalization cannot preserve accurate trajectories for publishing. The other trajectory $k$-anonymization approach is suppressing some sampling points or trajectory segments to achieve $k$ indistinguishable trajectories for the protection of identity and the associated sensitive values. But the related schemes [8]–[10] don't refer to the protection of sensitive location information in trajectory data, thus resulting in the disclosure of sensitive location information [11]. Additionally, only employing suppression for trajectory anonymization may incur excessive trajectory distortion, especially when trajectories are published together with sensitive attributes because more locations or trajectory segments are suppressed to protect sensitive values.

To address the above problems, this paper aims to design a novel trajectory anonymization scheme in a combined data publishing scenario, where trajectory data without modification are released with sensitive attributes. Our contributions are summarized as follows.

- We formalize three types of privacy requirements in our $(\alpha, K)_L$-privacy model for privacy preservation when publishing trajectory data with sensitive attributes, to resist identity linkage, sensitive location, and sensitive value linkage attack.
- We design an anonymization scheme named IEVS and implement the related algorithms to publish trajectory data satisfying $(\alpha, K)_L$-privacy requirement. IEVS employs techniques of trajectory splitting with location suppression and sensitive value generalization to preserve data utility of the anonymized dataset as much as impossible.
- A set of experiments are implemented to evaluate the effectiveness of our scheme.

The rest of our paper is organized as follows. Section 2 discusses related work. Section 3 formulates the problem. We detail the IEVS scheme in Section 4. In Section 5, a set of experiments are run on a synthetic dataset to evaluate our scheme. Finally, we conclude this paper in Section 6.

## II. RELATED WORK

### A. STATIC RELATIONAL DATA ANONYMIZATION

$k$-anonymity [12] first applies to static relational data anonymization. It can hold back the identity linkage attack through creating equivalence classes, every of which includes at least $k$ records with identical quasi-identifiers (*QID*s). Some other extensive versions based on $k$-anonymity include: $l$-diversity [13] requires $l$ distinct sensitive values in each equivalence class to protect against attribute disclosure, $t$-closeness [14] gives strict limitation on the distance of distribution of a sensitive value between any equivalence class and the overall dataset, $(\alpha, k)$-anonymity [15] constrains the occurrence frequency of every sensitive value in each equivalence class, which is available for the anonymization of the dataset with average distribution sensitive values, and the multi-sensitive bucketization model [16] is presented for achieving the $k$-anonymity of the relational dataset with multi-sensitive attributes. Considering different requirements of privacy protection about different individuals, Xiao and Tao [17] first present the notion of personalized privacy protection. They set a guarding node (i.e., a generalized sensitive value) for the sensitive value of each individual to meet personalized anonymity. PE$(\alpha, k)$-anonymity model [18] is proposed to unify individual- and sensitive value-oriented anonymity for personalized anonymization on relational data publishing.

All the above methods for relational data anonymization are not suitable for trajectories with characteristics of high dimensionality, sparsity, and time-sequence. The reason is that the anonymization, which makes the *QID*s of at least $k$ trajectories identical with each other in a cluster, would cause excessive information loss.

### B. TRAJECTORY DATA ANONYMIZATION

#### 1) ON LOCATION PRIVACY PROTECTION

Some studies [4], [6], [19], [20] focus on the protection of stop points of a trajectory instead of the whole trajectory to reduce the amount of unnecessary distortion. In [6], generalization is employed to replace the positions of stop points with coarse $l$-diverse zones. Han and Tsai [19] propose four privacy risk levels for stop locations and consider both spatial and semantic closeness in semantic location translation. Naghizade *et al.* [4] think that stop points are the most sensitive part of the trajectory and propose an efficient algorithm based on the method of perturbation, where sensitive stop points are substituted by either moves from the same trajectory or a minimal detour. In [20], a taxonomy tree for semantic attributes of all sampling points is first built, then the sensitive stop points are replaced with alternative places

of interest (POIs) in the tree according to their sensitivities set by trajectory users.

Location $k$-anonymity achieves local generalized areas with at least $k$ locations in different trajectories. So the adversary can still analyze the movement mode of the trajectory and even infers the trajectory associated with the target when the adversary knows partial location-time data of the target in different generalized areas.

#### 2) ON TRAJECTORY PRIVACY PROTECTION

##### *a: CLUSTER-BASED TRAJECTORY ANONYMIZATION*

Most of the clustering-based approaches are employed under the assumption that the attacker knows the whole trajectory information of mobile objects. NWA [7] first proposes $(k, \delta)$-anonymity model to gather at least $k$ trajectories close to each other into a cylinder of radius $\delta$ and generate a representative trajectory through space translation. In [21], *swaplocation* is devised to change the location sequence of trajectories in each clustered group by microaggregation. Lin *et al.* [22] construct the $k$-anonymized trajectory dataset based on a clustering method to resist the identity disclosure of mobile individuals in a road-network. Dong and Pi [5] propose a novel scheme TOPF to generate cluster groups based on the frequent path in a road-network. TOPF first removes infrequent roads in each trajectory and uses frequent path to build clusters containing at least $k$ trajectories. Consequently, privacy is ensured by releasing only a set of selected representative trajectories. Huo *et al.* [6] select trajectory clusters by graph partition according to the spatial distance of trajectories to reduce the information loss. Gao *et al.* [23] propose the notion of trajectory angle to take the trajectory direction into account when evaluating the similarities of trajectories to form $k$-anonymized trajectory clusters. $(k, \Delta)$-anonymity model [24], an extended model of $(k, \delta)$-anonymity, uses clustering and space translation to get personalized $k$-anonymized trajectories with respect to different user-defined thresholds $k$ and $\Delta$. Furthermore, the WCOP-SA algorithm in [24] partitions trajectories into several segments during the anonymization process aiming at improving data utility. In [25], machine learning algorithms are applied to cluster the trajectories and a variation of the $k$-means algorithm is developed to preserve the privacy in overly sensitive datasets.

##### *b: GENERALIZATION-BASED TRAJECTORY ANONYMIZATION*

To resist subtrajectory linkage attack, at least $k$ trajectory segments are generalized into areas by employing the generalization-based method, assuming that partial trajectory segments are the background knowledge of adversaries. In [26] three generalization-based algorithms SEQANON, SD-DEQANON, and U-SEQANON are designed for location distance, semantic similarity, and user-specified utility requirements respectively. To resist record linkage attack and probabilistic attack, Gramaglia *et al.* [27] propose $k^{\tau,\epsilon}$-anonymity model and develop the `kte-hide` algorithm to ensure that the adversaries holding the trajectory segment of the target individual in time interval $\tau$ cannot link the individual with less than $k$ trajectories in the next following time interval $\epsilon$. Tu *et al.* [28] propose a strong privacy protection scheme where a trajectory cannot distinguish with any other $k-1$ trajectories and any generalized area contains $l$-diverse POI information with $t$-closeness in the dataset, to resist re-identification attack and semantic attack, respectively.

Generally, the trajectories within a cluster are substituted with a representative trajectory after the anonymization based on clustering. So the published trajectory dataset is a set of representative trajectories, which can be used for aggregation analysis but not for behaviour pattern discovery and the mining of association rules [5]. Similarly, the generalized trajectory dataset cannot provide accurate location-time points, which has the same limitation of usage with the cluster-based approach.

##### *c: SUPPRESSION-BASED TRAJECTORY ANONYMIZATION*

Trajectory anonymization based on suppression deletes partial trajectories or locations leading to privacy leakage to achieve a subset of raw trajectories. Terrovitis and Mamoulis [29] propose an anonymization approach, by removing the minimum number of POIs from trajectories, to prevent trajectories from the identity linkage attack by multi-attackers holding the knowledge of temporal POI. By employing local suppression, Chen *et al.* [8] propose $(K, C)_L$-privacy model for trajectory anonymization. The model guarantees that the identification probability of the target trajectory is not higher than $1/k$ and the identification probability of the sensitive value is not higher than $C$. Furthermore, Chen *et al.* employ local suppression to achieve higher data utility compared with the method of global suppression presented in [9]. The personalized anonymity scheme on trajectories PPTD [30] uses local suppression and generalization to provide different degrees of privacy protection on trajectories and the associated sensitive values according to the preset privacy levels. Al-Hussaeni *et al.* [10] propose the ITSA algorithm to incrementally anonymize trajectory data streams based on global suppression.

Our $(\alpha, K)_L$-privacy model extends the $(K, C)_L$-privacy model [8], [9] by the addition of the protection of sensitive locations. To preserve more location-time points in trajectories for high data utility, we design a novel anonymization scheme IEVS, which employs trajectory splitting and location suppression to anonymize trajectories locally and globally respectively and employs generalization to anonymize sensitive values. Considering that our work is mostly related to the researches in [8] and [9], we use the schemes in [8] and [9] as the benchmarks to assess our IEVS scheme in the experiments. This paper is the extension of our previous work in [31].

## III. PROBLEM DEFINITION

In this section, we first introduce some notions about a trajectory dataset, then propose the attack model, and $(\alpha, K)_L$-privacy model followed for trajectory anonymization.

### A. TRAJECTORY DATASET

*Definition 1 (Trajectory): A trajectory $t$ of length $n$ is described as a sequence $t = l_1 \rightarrow l_2 \rightarrow \ldots \rightarrow l_n$ of $n$ locations visited by a trajectory user in chronological order, where $l_i$ $(1 \leq i \leq n)$ represents a POI in the map. The length of $t$ is denoted as $|t|$, i.e., $|t| = n$.*

We address the problem of privacy preservation in a combined data publishing scenario, where the trajectory data of an individual is published together with his/her sensitive attribute. We give the notion of trajectory record for the appropriate form of data.

*Definition 2 (Trajectory Record): A trajectory record $tr$ of a specific data user contains his/her identity $id$, trajectory $t$ and sensitive attribute value $s$, described as*

$$tr = \langle id, t, s \rangle$$

The identity information $id$ should be removed in the trajectory preprocessing phase. It is replaced with a number for a clear description in the following sections. The value of sensitive attribute $s$ may involve the privacy of the trajectory user. We only discuss a single sensitive attribute in this paper.

*Definition 3 (Trajectory Dataset): A trajectory dataset $T$ is a set of trajectory records, i.e., $T = \cup tr$. $|T|$ represents the number of trajectory records in $T$. Each user is associated with one trajectory record in the dataset.*

*Definition 4 (Subtrajectory): Let $t = l_1 \rightarrow l_2 \rightarrow \ldots \rightarrow l_n$ be a trajectory in trajectory dataset $T$. $t^* = l_1^* \rightarrow l_2^* \rightarrow \ldots \rightarrow l_m^*$ $(m \geq 1)$ is a subtrajectory of $t$, denoted as $t^* = sub(t)$, if and only if there are $l_1^* = l_{p_1}, l_2^* = l_{p_2}, \ldots, l_m^* = l_{p_i}$ $(1 \leq p_1 < \ldots < p_i \leq n)$ and $m < n$. All trajectories containing $t^*$ in $T$ is denoted as $T(t^*)$.*

Some locations are considered sensitive in a trajectory (sensitive stop points along a path, e.g., a fact that an individual visits an AIDS detection center). Some values of the sensitive attribute associated with trajectories also may be of sensitivity. Sensitive values are usually set by experts in the concerned field. We consider two types of sensitive information concerned with personal privacy in a trajectory dataset, sensitive locations and sensitive values, denoted as $SL$ and $SA$ respectively.

For example, Table 1 shows a trajectory dataset $T$ comprised of 6 trajectory records (*Records*#1 to 6). $a \rightarrow d$ is a subtrajectory of trajectories in *Records*#1, 2 and 5, with length 2. Locations $f$ and $g$ are considered sensitive. So the set of sensitive locations about $T$ is $\{f, g\}$. The domain of the sensitive attribute *Disease* of dataset $T$ is $\{gastritis, flu, HIV, cancer, fever\}$. *HIV* and *cancer* are usually considered sensitive in these values. So the set of sensitive values of dataset $T$ is $\{HIV, cancer\}$.

### B. PRIVACY ISSUES

An adversary can easily launch background knowledge attack once he holds some or all nonsensitive locations about the target individual, which may disclose the identity and sensitive information of the target, thereby leading to privacy threats.

*Definition 5: (Background Knowledge Attack). Assuming that an adversary's background knowledge about target $v$, $bk_v$, is a subtrajectory of trajectory $t_v$ and $bk_v$ is composed of up to $L$ nonsensitive locations $(L < |t_v|)$, three types of private information may be disclosed.*

#### 1) IDENTITY DISCLOSURE

*In trajectory dataset $T$, if the number of trajectory records containing subtrajectory $bk_v$, i.e., $|T(bk_v)|$, is small, the adversary can easily make the association between the trajectory record and target $v$ in $T$ with high probability, and further, obtain the sensitive locations and sensitive value of $v$.*

#### 2) SENSITIVE LOCATION DISCLOSURE

*In $T$, if the identification probability, i.e., the probability of associating a sensitive location $sl$ to the trajectory records containing $bk_v$, i.e., $P(sl, T(bk_v)) = \frac{|T(sl) \cap T(bk_v)|}{|T(bk_v)|}$ is high, the fact that target $v$ ever visiting $sl$ can be easily inferred.*

#### 3) SENSITIVE VALUE DISCLOSURE

*In $T$, if the identification probability, i.e., the probability of associating a sensitive value $s$ to the trajectory records containing $bk_v$, i.e., $P(s, T(bk_v)) = \frac{|T(s) \cap T(bk_v)|}{|T(bk_v)|}$ is high, the fact that target $v$ owning sensitive value $s$ can be easily disclosed.*

### C. PRIVACY MODEL

To protect privacy from background knowledge attack, $(\alpha, K)_L$-privacy model is proposed for anonymized trajectory publishing.

*Definition 6 ($(\alpha, K)_L$-Privacy Model): Given a trajectory dataset $T$, $T = \{\langle 1, t_1, s_1 \rangle, \ldots, \langle n, t_n, s_n \rangle\}$, the set of sensitive locations $SL$ and sensitive values $SA$, thresholds $K$, $L$ and $\alpha$ $(0 \leq \alpha \leq 1)$, $T$ is said to be an $(\alpha, K)_L$-privacy of a trajectory dataset if and only if: (i) $|T(st)| \geq K$, for every subtrajectory $st$, composed of up to $L$ nonsensitive locations in $T$, (ii) $P(sl, T(st)) \leq \alpha$, for every sensitive location $sl \in SL$ contained in $T(st)$, and (iii) $P(s, T(st)) \leq \alpha$, for every sensitive value $s \in SA$ contained in $T(st)$.*

$(\alpha, K)_L$-privacy model guarantees that an adversary, holding any subtrajectory $st$ composed of up to $L$ nonsensitive locations in a trajectory dataset, cannot associate any trajectory user with fewer than $K$ trajectory records in the published dataset, nor can associate any sensitive location $sl$ or value $s$ with the probability of higher than threshold $\alpha$.

Reversely, if a subtrajectory $st'$ in dataset $T$ doesn't meet at least one of the above three conditions of Definition 6, $st'$ violates $(\alpha, K)_L$-privacy. The notions of violating subtrajectory and minimal violating subtrajectory (MVST) are described as follows.

*Definition 7 (Violating Subtrajectory and Minimal Violating Subtrajectory): Given a trajectory dataset $T$, $st'$ is a subtrajectory of some trajectory in $T$. $st'$ is said to be a violating subtrajectory w.r.t. $(\alpha, K)_L$-privacy if (i) $0 < |T(st')| < K$, or (ii) $P(sl, T(st')) > \alpha$, or (iii) $P(s, T(st')) > \alpha$, where $sl$ and $s$ are respectively a predefined sensitive location and value in $T$. Furthermore, $st'$ is a minimal violating*

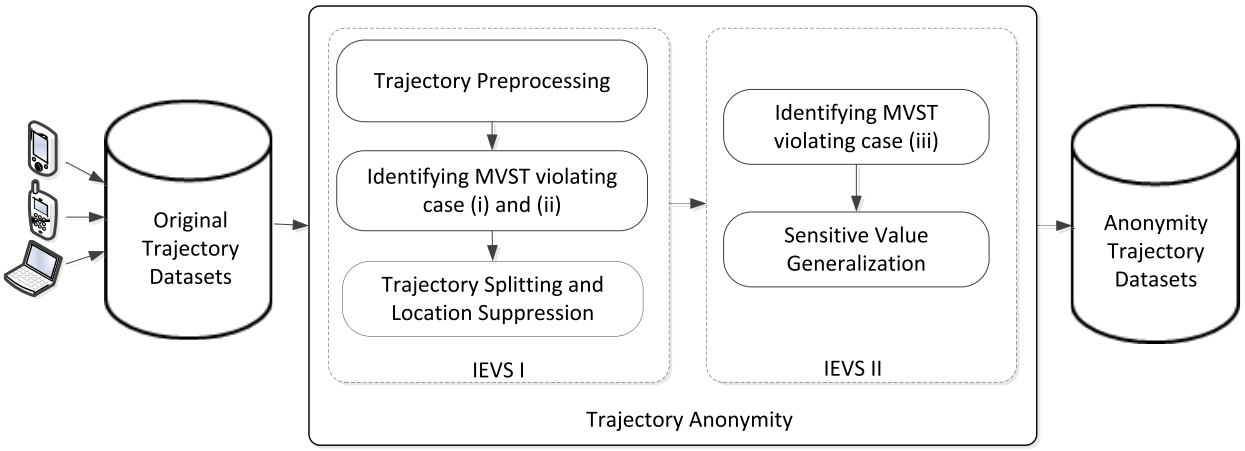


**FIGURE 1.** IEVS framework for $(\alpha, K)_L$-privacy trajectory dataset.

*subtrajectory if every proper subtrajectory of $st'$ is not a violating subtrajectory.*

*Definition 8 (Problematic Trajectory): Given a trajectory dataset $T$, a trajectory $t$ and an MVST $st$ in $T$. $t$ is called a problematic trajectory respecting $st$ if $t \in T(st)$.*

**TABLE 1.** Raw trajectory dataset *T*.

| Rec. | Trajectory | Disease |
|---|---|---|
| 1 | $a \rightarrow b \rightarrow c \rightarrow d \rightarrow g$ | gastritis |
| 2 | $b \rightarrow a \rightarrow d \rightarrow f$ | flu |
| 3 | $b \rightarrow d \rightarrow c$ | HIV |
| 4 | $a \rightarrow c$ | cancer |
| 5 | $e \rightarrow a \rightarrow d \rightarrow c$ | cancer |
| 6 | $a \rightarrow g \rightarrow b$ | fever |

For example, given $\alpha = 0.5, K = 2, L = 2, SL = \{f, g\}$ and $SA = \{HIV, cancer\}$. In Table 1, $t_1 = c \rightarrow d$ is an MVST, because $|T(t_1)| < K$, and none of its subtrajectories, $c$ or $d$, is a violating subtrajectory. $t_2 = a \rightarrow b$ is also an MVST, beacause $P(g, T(t_2)) = 1 > \alpha$. $t_3 = a \rightarrow c$ is an MVST too, beacause $P(cancer, T(t_3)) = \frac{2}{3} > \alpha$. The problematic trajectories respecting $a \rightarrow c$ are trajectories of *Records*#1, 4 and 5. $t_4 = e \rightarrow a$ is a violating subtrajectory, but not an MVST, because $e$, one of the subtrajectory of $t_4$, is an MVST.

### D. PROBLEM STATEMENT

Trajectory dataset $T$ is an $(\alpha, K)_L$-privacy of the trajectory dataset iff no violating subtrajectory is contained in $T$. It is sufficient to meet with $(\alpha, K)_L$-privacy by eliminating all MVST in $T$, which reduces the workload of violating subtrajectory enumeration [8], [9]. Aiming at this, we propose IEVS scheme to identify and eliminate all MVST. In particular, splitting is employed first to locally eliminate the MVST generated by Cases (*i*) and (*ii*) of Definition 7, supplemented by global suppression. Then sensitive value generalization is employed to eliminate those generated by Case (*iii*) of Definition 7. By comparison of suppression method used in [8] and [9], the IEVS scheme can preserve locations as much as possible by local recoding of trajectory splitting and reduce the information loss of trajectories by sensitive value generalization. As a result, in the published dataset, every trajectory user cannot be deduced to link with his record with the probability of higher than $1/k$ and each of sensitive locations and values cannot be inferred with the probability of higher than threshold $\alpha$, on the condition that an adversary holds the subtrajectories composed of up to $L$ nonsensitive locations, which can be thought as the trajectory anonymization.

Subtrajectories composed of nonsensitive locations in $T$ can be regarded as *QIDs*. Sensitive information includes sensitive locations and values. The performance of our proposed IEVS scheme cuts off the one-to-one relationship between *QIDs* and sensitive information, therefore the anonymization of the trajectory dataset for privacy protection is achieved.

*Definition 9 (Trajectory Dataset Anonymization): Given trajectory dataset $T$ and $(\alpha, K)_L$-privacy requirement, the goal of anonymization of $T$ is to achieve a sanitized version $T^*$ of $T$ by using the IEVS scheme such that $T^*$ not only satisfies $(\alpha, K)_L$-privacy but also preserves high data quality.*

## IV. IEVS SCHEME

In this section, the framework for the IEVS scheme is first introduced. Then the function of each component is detailed.

### A. FRAMEWORK OVERVIEW

IEVS framework is applied in the off-line mode, where trajectory records are anonymized after data collection, and then follows data publishing for data analysis and application. The architecture of the IEVS framework is shown in Fig. 1. We mainly focus on trajectory anonymity. It consists of two modules IEVS I and II, based on the idea of identifying and eliminating MVST.

### B. IEVS I

We identify all the MVST violating $K$-anonymity or sensitive location $\alpha$-privacy after the trajectory preprocessing phase in

IEVS I, then eliminate them by local trajectory splitting and global location suppression.

#### 1) TRAJECTORY PREPROCESSING

Trajectory preprocessing first removes trajectory users' *ID*s from trajectory dataset $T$, then creates the set of sensitive locations $SL$, the set of sensitive values $SA$, and the set of trajectory users respecting nonsensitive locations $TO$. $TO$ is described as

$$TO = \cup TO_{nsl} = \cup\{\langle i, tm\rangle | \langle i, t, s\rangle \in T, \langle nsl, tm\rangle \in t\}$$

where $nsl$ is a nonsensitive location in $T$, $TO_{nsl}$ is the set of trajectory users ever visiting $nsl$, and $tm$ is the timestamp reflecting when the user associating *Record#i* visited $nsl$.

The purpose of establishing the trajectory user set in the preprocessing phase is to improve the efficiency of the algorithm for MVST identification. If we directly visit trajectory dataset $T$ to compute the number of a subtrajectory in $T$, we must traverse all trajectories of $T$, which results in high time cost. While by visiting the trajectory user set corresponding to each nonsensitive location in $T$, scanning $T$ is avoided. Therefore, the runtime of the MVST identification algorithm is decreased.

#### 2) IDENTIFYING MVST

We need to identify all the MVST that are (*i*) composed of nonsensitive locations of up to length $L$, and (*ii*) not satisfied with $K$-anonymity or sensitive location $\alpha$-privacy. Algorithm Iden-MV (Algorithm 1) shows the details of identifying the MVST set.

Algorithm 1 inputs: (1) trajectory user set $TO$ generated in trajectory preprocessing; (2) sensitive location sets respecting each trajectory record $i$, $SL = \langle i, SL_i\rangle$; (3) thresholds $K$, $\alpha$, and $L$. Algorithm 1 outputs the MVST set $B$ violating $K$-anonymity or sensitive location $\alpha$-privacy.

In Algorithm 1, $C_m$, $B_m$, and $U_m$ represent the candidate MVST set, MVST set, and non-violating subtrajectories of length $m$ respectively. Algorithm 1 first sets the value of $m$ to 1 and initializes $C_1$ with all nonsensitive locations to identify all MVST of length 1 (line 1). Then for each trajectory $st \in C_m$, Algorithm 1 gets all trajectory users $TO_{st}$ who ever visited the locations in $st$ sequentially by computing the intersection of trajectory user sets of all locations in $st$ (line 4). If $st$ exists in the trajectory record dataset (line 6), we calculate $|TO_{st}|$, the number of users visiting $st$, and $\frac{|TO_{st}(sl)|}{|TO_{st}|}$, the percentage of users containing sensitive location $sl$ in $TO_{st}$. If $st$ satisfies both $K$-anonymity and $\alpha$-constraint respecting each sensitive location $sl$, i.e., $|TO_{st}| \geq K$ and $\frac{|TO_{st}(sl)|}{|TO_{st}|} \leq \alpha$, $st$ is added to $U_m$ for creating candidate MVST set $C_{m+1}$, otherwise, $st$ is added to $B_m$ (lines 7-11). Next, the candidate MVST set $C_{m+1}$ of length $m+1$ is generated in two steps (lines 14-20). First, the Cartesian product of $U_m$ and $U_m$ (denoted as $U_m \widetilde{\times} U_m$ in Algorithm 1) is conducted with the consideration of temporal sequentiality. Second, all the super trajectories of the identified MVST are deleted from $C_{m+1}$.

**Algorithm 1** Iden-MV

**Input:** Trajectory user set $TO$, sensitive locations $SL$, thresholds $K$, $\alpha$ and $L$

**Output:** MVST set $B$ violating $K$-anonymity or sensitive location $\alpha$-privacy

1: $C_1 \leftarrow$distinct locations in $TO$, $B_1 = U_1 = \emptyset$, $m = 1$;
2: **while** $m \leq L$ and $C_m \neq \emptyset$ **do**
3:  **for** each trajectory $st \in C_m$ **do**
4:  Compute $TO_{st}$, i.e., $\cap TO_l$, for every nonsensitive location $l \in st$;
5:  Compute $|TO_{st}(sl)|$, for every $sl \in SL_i$ where $i$ is a user in $TO_{st}$;
6:  **if** $|TO_{st}| > 0$ **then**
7:  **if** $|TO_{st}| < K$ or $\frac{|TO_{st}(sl)|}{|TO_{st}|} > \alpha$ **then**
8:  $B_m \leftarrow st$;
9:  **else**
10:  $U_m \leftarrow st$;
11:  **end if**
12:  **end if**
13:  **end for**
14:  $m{+}{+}$;
15:  $C_m \leftarrow U_{m-1} \widetilde{\times} U_{m-1}$;
16:  **for** each trajectory $st \in C_m$ **do**
17:  **if** $st$ is a super sequence of $st^*$ where $st^* \in B_{m-1}$ **then**
18:  Remove $st$ from $C_m$;
19:  **end if**
20:  **end for**
21: **end while**
22: **return** $B = B_1 \cup B_2 \cup \ldots \cup B_L$;

For example, given $K = 2$, $\alpha = 0.5$, $L = 2$, $SL = \{f, g\}$, the MVST set of length 1 generated from Table 1 $B_1 = \{e\}$. So all nonsensitive locations except location $e$ are put into $U_1$ to generate the candidate MVST $C_2$ by $U_1 \widetilde{\times} U_1$. The result MVST set $B = \{e, a \rightarrow b, c \rightarrow d, b \rightarrow a\}$.

#### 3) ELIMINATING MVST

To eliminate the MVST set $B$, the anonymity technique of trajectory splitting is preferentially applied. The principal idea of splitting is that each problematic trajectory containing an MVST is segmented at a certain location so that the result trajectories after segmentation no longer include the MVST. For example, in Table 1, if $b \rightarrow d \rightarrow c$ is an MVST, the corresponding problematic trajectory, *Record#* 3, can be split into two trajectories $b$ and $d \rightarrow c$ at location $b$ or $b \rightarrow d$ and $c$ at location $d$, to eliminate the MVST $b \rightarrow d \rightarrow c$.

When we eliminate an MVST $bt$ in dataset $T$, we can split the problematic trajectories $T(bt)$ at any location except the last one of $bt$. The locations except the last one are the candidate split positions of $T(bt)$ respecting $bt$. However, can all these candidate split positions be used as the split ones? If so, how to determine the most suitable position? The discussion of these two considerations is followed in turn.

Every splitting operation separates each original problematic trajectory into two trajectories, which causes the loss of some subtrajectories of the original trajectory. The number of these subtrajectories thus reduces. As a result, the remaining number of them in the dataset may be fewer than $K$ or the identification probabilities about the sensitive locations associating with them may be higher than threshold $\alpha$. Therefore, these subtrajectories become new violating subtrajectories. For example, if *Record*# 3 in Table 1 is split at location $b$, a new MVST $b \rightarrow c$ generates because of violating $K$-anonymity assuming that $K = 2$ (the number of $b \rightarrow c$ becomes 1 after splitting).

Assuming that we split the problematic trajectories $T(bt)$ in dataset $T$ to eliminate an MVST $bt$, $T_r$ represents the remaining trajectories in $T$, i.e., $T_r = T - T(bt)$, and the lost subtrajectories, composed of up to $L$ nonsensitive locations, are denoted as $CQ$. For a sequence $q \in CQ$, if $q$ doesn't belong to $B$ or doesn't contain any of the sequence of $B$, i.e., $q \notin B \wedge q' \neq sub(q)$ (where $q'$ is a sequence of $B$), and $q$ satisfies: $T_r(q) < K$ or $\frac{|T_r(sl) \cap T_r(q)|}{|T_r(q)|} > \alpha$ (where $sl$ is a sensitive location of $T_r(q)$ on the premise of $T_r(q) > 0$), $q$ is a new MVST of dataset $T$.

When splitting at a certain location causes the generation of the new MVST, it cannot guarantee to eliminate all MVST in $|B|$ iterations. To preserve the effectiveness of trajectory anonymization, the location can't be used as a split position. This is what we adopt to determine if a candidate split location can be a split position. Next it comes to the second problem.

If there exists more than one split location, the **anonymity gain** metric is defined to find the optimal split position for balancing between privacy and data utility.

From the respect of **privacy protection**, when the problematic trajectories are split at location $p$, the more the number of instances of eliminated MVST, the better the effect of trajectory anonymization. Thus, the metric of **privacy protection gain**, denoted as $PG_{spt}(p)$, is described as

$$PG_{spt}(p) = NUM_{spt} \tag{1}$$

where $NUM_{spt}$ represents the number of instances of eliminated MVST after the problematic trajectories are split at location $p$.

From the respect of **information loss**, splitting a trajectory can preserve all the locations of it, but would affect the co-appearance of distinct locations in the original trajectory, which causes the information loss of count queries and frequent patterns. These two are the bases for utility metrics in many related works [32], [33]. To measure this distortion, we use $IL_{spt}(p)$, the number of lost subtrajectories of length 2, after a trajectory $t$ is split into $t_1$ and $t_2$ at location $p$, for information loss metric, defined as

$$IL_{spt}(p) = NUM_b - NUM_a \tag{2}$$

where $NUM_b$ is the number of subtrajectories of $t$ with length 2, $NUM_a$ is the sum number of subtrajectories of $t_1$ and $t_2$ with length 2.

We compute the **anonymity gain** $AG_{spt}(p)$ to find the optimal split position by

$$AG_{spt}(p) = \frac{PG_{spt}(p)}{IL_{spt}(p)} = \frac{NUM_{spt}}{NUM_b - NUM_a} \tag{3}$$

Obviously, a greater $PG_{spt}(p)$ denotes larger privacy protection gain and a less $IL_{spt}(p)$ denotes lower information loss.

If there is no split position to separate problematic trajectories for the elimination of an MVST, location suppression [9] is applied as the complement to trajectory splitting. The main idea of location suppression is removing all the instances of a location in an MVST from the dataset such that the MVST is eliminated. For an MVST with a minimum length of 2, more than one location can be selected as the suppression location to remove. When all the instances of a location $p$ in an MVST are removed from dataset $T$, the more instances of removed MVST and the fewer amounts of removed instances of location $p$ bring optimal trajectory anonymization. So the **anonymity gain** metric for optimal location suppression $AG_{sup}(p)$ is defined by

$$AG_{sup}(p) = \frac{NUM_{sup}}{NUM'_b} \tag{4}$$

where $NUM_{sup}$ is the number of instances of eliminated MVST in $B$ if $p$ is removed in the dataset, and $NUM'_b$ is the number of instances of location $p$.

The Elim-MV algorithm (Algorithm 2) depicts the details of eliminating MVST. It inputs MVST set $B$ and the original trajectory dataset $T$, thresholds $K$, $\alpha$, and $L$. It outputs the anonymized dataset $T'$ satisfying $K$-anonymity and sensitive location $\alpha$-privacy.

Algorithm 2 first eliminates all the MVST of length 1 in $B$ by removing all the instances of locations in $B$ from $T$ (lines 1-2). Next, for every remaining trajectory $bt$ in $B$, the Check-SP algorithm (Algorithm 3) is called to get the split position $p$ with maximal anonymity gain when trajectory splitting is done on $T(bt)$ (line 7). If $p$ doesn't exist (i.e., splitting trajectories in $T(bt)$ at any nonsensitive location in $bt$ lead to the generation of the new MVST), location suppression is applied to eliminate $bt$ (lines 8-11). In particular, line 9 first computes the suppression location $p'$ with maximal anonymity gain when location suppression is done on $T(bt)$, and then line 10 deletes all instances of $p'$ in $T$. $bt$ is thus eliminated by removing it from $B$ in line 11. Otherwise, each trajectory in $T(bt)$ is split into two segments $t_1$ and $t_2$ at location $p$, and the sensitive attribute value respecting $t$ is added to $t_1$ and $t_2$ respectively to form two new trajectory records (line 14). In the end, $bt$ is deleted from $B$ (line 18). The anonymized trajectory dataset $T'$ is thus achieved.

We design the Check-SP algorithm (Algorithm 3) to find the optimal split location for eliminating an MVST. Algorithm Check-SP inputs MVST $bt$, the MVST set $B$ and original dataset $T$, thresholds $\alpha$, $K$, $L$. It outputs the split position $p$ with maximal anonymity gain.

**Algorithm 2** Elim-MV

**Input:** Dataset $T$, MVST $B$, sensitive locations $SL$, sensitive values $SA$, thresholds $K$, $L$, $\alpha$

**Output:** The anonymized dataset $T'$ satisfying $K$-anonymity and sensitive location $\alpha$-privacy

1: Remove the instances of all size-one MVST in $B$, from $T$;
2: Remove all size-one MVST from $B$;
3: **for** each $bt \in B$ **do**
4: **if** $T(bt) = \emptyset$ **then**
5: $B = B \setminus \{bt\}$;
6: **else**
7: $p$=ISLA($bt$, $B$, $T$);
8: **if** $p =$ NULL **then**
9: Compute suppression location $p'$ in $bt$ by Formula (4);
10: Remove all instances of $p'$ from $T$;
11: Remove the MVST containing $p'$ from $B$;
12: **else**
13: **for** each $t \in T(bt)$ **do**
14: Split $t$ at $p$ into $t_1$ and $t_2$ and add $SA(t)$ to $t_1$ and $t_2$;
15: **end for**
16: **end if**
17: **end if**
18: $B = B \setminus \{bt\}$;
19: **end for**
20: $T' \leftarrow T$;
21: **return** $T'$;

**Algorithm 3** Check-SP

**Input:** An MVST $bt$, dataset $T$, MVST set $B$, sensitive locations $SL$, thresholds $K$, $L$ and $\alpha$

**Output:** Split position $p$ of $bt$;

1: $P \leftarrow$all locations of $bt$ except the last one;
2: **for** each $p \in P$ **do**
3: **for** each trajectory $t \in T(bt)$ **do**
4: $CQ_t \leftarrow$the lost subtrajectories composed of up to $L$ nonsensitive locations in $t$ after splitting $t$ at $p$;
5: Remove all trajectories, together with their super sequences, in $B$ from $CQ_t$;
6: **for** each $cq \in CQ_t$ **do**
7: $CQ = CQ \cup cq$;
8: $cq.num$++;
9: **end for**
10: **end for**
11: **if** $(|T(cq)| - cq.num) < K$ or $P(sl, T(cq) - cq.num) > \alpha$ for any $cq \in CQ$ **then**
12: $AG_{spt}(p) = -1$;
13: Go to line 2;
14: **else**
15: Compute $AG_{spt}(p)$ by Formula (3);
16: **end if**
17: **end for**
18: **if** $AG_{spt}(p) = -1$ for any $p \in P$ **then**
19: **return** NULL;
20: **else**
21: **return** the $p$ with the maximum $AG_{spt}(p)$;
22: **end if**

In Algorithm 3, all locations in $bt$ except the last one are put into the set of candidate split positions $P$ first (line 1). Next, each location $p \in P$, is checked whether it can be as the split position or not, that is, whether a new MVST generates when splitting trajectories at $p$, which details in the next two steps. We compute the new candidate MVST $CQ$ in the first step. Specifically, for each problematic trajectory $t \in T(bt)$, we compute the lost trajectory sequences $CQ_t$ when $t$ is split at $p$ (line 4). Then we delete the two types of trajectories, the trajectories in $B$ and the super trajectories respecting trajectories in $B$, from $CQ_t$ (line 5). Next, we merge all $CQ_t$ into $CQ$ and count the number of each distinct trajectory in $CQ$ by $num$ (lines 6-9). In the second step, for each sequence $cq$ in $CQ$, we check if the number of $cq$ and the identification probability of $cq$ in the trajectories are more than $K$ and less than $\alpha$ respectively. If not, $p$ cannot be as a split position (lines 11-12). Otherwise, the anonymity gain of $p$ is calculated (line 15). We return the split position $p$ with the maximum anonymity gain or NULL if none of the locations in $bt$ is a split position (lines 18-22).

For example, we eliminate the MVST $a \rightarrow d \rightarrow c$ in the trajectory record dataset $T$ of Table 1 (given $L = 3$). The problematic trajectory $T(a \rightarrow d \rightarrow c)$ is *Record*#5 where location $e$, the MVST with length 1, has been removed. So the candidate split positions $P = \{a, d\}$. If *Record*#5 is split at location $a$, the lost subtrajectories $CQ = \{a \rightarrow d, a \rightarrow c\}$. Location $a$ is the split position for neither of the sequences in $CQ$ is a new MVST. But location $d$ cannot be a split position because one of its lost subtrajectories $d \rightarrow c$ becomes a new MVST if *Record*#5 is split at $d$.

### C. IEVS II

In the IEVS II module, we need to eliminate all MVST violating sensitive value $\alpha$-privacy in dateset $T'$ generated after the execution of the IEVS I module. Different with what we do in IEVS I, in IEVS II we adopt the method of generalization to replace the sensitive values with inaccurate values (i.e., the generalized values) to make the identification probability of them no higher than the preset probability constraint threshold $\alpha$, thus achieving $(\alpha, K)_L$-privacy about sensitive values. In this sense, the MVST respecting sensitive values are eliminated. So we need first to find all the instances of sensitive values which are identified with the probability of higher than $\alpha$ by the adversary with nonsensitive location sequences of up to length $L$. Then we generalize them according to a predefined generalization tree of the sensitive attribute.

To find the instances of sensitive values to be generalized, we should identify the subtrajectories violating sensitive value $\alpha$-privacy in $T'$. The instances of sensitive values, which are contained in the same trajectory records with these subtrajectories, are what we need to generalize. To identify all MVST violating sensitive value $\alpha$-privacy in $T'$, we reconstruct the trajectory user set $TO'$ respecting non-sensitive locations in $T'$ and modify the Iden-MV algorithm as follows. First, the user set $TO$ is replaced with $TO'$, and sensitive value set $SA$ is added into the input. Second, line 5 is modified as "*Compute* $|TO'_{st}(s)|$, *for each* $s \in SA$;". Third, line 7 is modified as "*if* $\frac{|TO'_{st}(s)|}{|TO'_{st}|} > \alpha$ *then*". The modified Iden-MV algorithm is denoted by Iden-MV$'$. The output is the MVST violating sensitive value $\alpha$-privacy in $T'$, denoted as $B'$.

Next, a taxonomy tree is built up for all values of the sensitive attribute to generalize the sensitive values identified with the probability of higher than $\alpha$. The relevant concepts are defined as follows.

*Definition 10 (Taxonomy Tree for Sensitive Attribute): Given the sensitive attribute S in a trajectory dataset T, D(S) represents the domain of S. A taxonomy tree GT for D(S) is defined as a 2-tuple, GT* $= \langle N, h \rangle$, *where*

- *N is the set of all nodes of GT. There are two types of nodes in GT: leaf node and internal node. The set of leaf nodes is D(S). For any node n in N, if n is not a leaf node, it must be an internal node. The set of internal nodes is denoted as IN which represents the set of generalized values.*
- *h:* $IN \rightarrow D(S)$ *denotes a reflection between internal nodes and leaf nodes. Given an internal node inode, h(inode) represents all the leaf nodes that can be generalized to inode, and* $|h(inode)|$ *denotes the number of them.*

*Definition 11 (Replacing Node): Given two nodes* $n_1$, $n_2$ *in N,* $n_2$ *is called a replacing node of* $n_1$ *and denoted as* $n_2 = r(n_1)$ *iff* $h(n_1) \subset h(n_2)$.

*Definition 12 (Generalization Level): Given two nodes* $n_1$, $n_2$ *in N and* $n_2 = r(n_1)$, *if* $n_1$ *is a leaf node, the length of the shortest path from* $n_2$ *to* $n_1$ *is the generalization level when* $n_1$ *is generalized to* $n_2$, *denoted as* $l(n_1, n_2)$.

For example, a taxonomy tree for the sensitive attribute *Disease* in Table 1 is shown in Fig. 2. *respiratory infection* and *immune system disease* are the replacing nodes of *flu*. The generalization level of *respiratory infection* and *immune system disease* are 1 and 2, respectively.

We then give the notion of *guarding node*, which is used as a generalized value to replace the corresponding sensitive value.

*Definition 13 (Guarding Node): Given a trajectory dataset T, the sensitive attribute S in T, a taxonomy tree GT for the domain of S D(S), N is the set of all nodes in GT, SA is the sensitive values to be generalized. For* $\forall s \in SA$, *if* $\exists n \in N$, $n = r(s) \wedge \frac{1}{h(n)} \leq \alpha$, *and the value of* $l(s, n)$ *is the minimum, n is said the guarding node of s.*

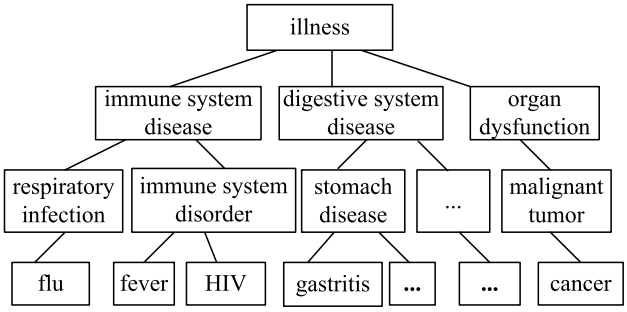


**FIGURE 2.** A taxonomy tree for sensitive attribute *Disease*.

**Algorithm 4** Gen-SA

**Input:** Trajectory dataset $T'$, taxonomy tree $GT$ and threshold $\alpha$

**Output:** The anonymized trajectory dataset $T^*$ meeting $(\alpha, K)_L$-privacy

1: Construct the trajectory user set $TO'$ respecting $T'$;
2: $B' \leftarrow$ all MVST violating sensitive value $\alpha$-privacy in $T$ generated by calling Iden-MV$'$);
3: $G_s \leftarrow \{s | \frac{|TO'_{st}(s)|}{|TO'_{st}|}\} > \alpha$, for every $s \in SA$ and every $st \in B'\}$;
4: **for** each $s \in G_s$ **do**
5: Visit $GT$ to find the guarding node $gn$ of $s$;
6: Replace $s$ with $gn$;
7: **end for**
8: The modified trajectory dataset $T'$ is denoted as $T^*$;
9: **return** $T^*$;

The algorithm Gen-SA (Algorithm 4) describes the process of sensitive value generalization. It inputs the result dataset $T'$ output by Algorithm 2, the predefined generalized tree $GT$ and threshold $\alpha$. It outputs $(\alpha, K)_L$-privacy of trajectory dataset $T^*$.

Gen-SA first identifies $B'$, MVST set violating sensitive value $\alpha$-privacy in trajectory set $T'$ (lines 1-2) and puts the sensitive values that need to be generalized into $G_s$ (line 3). Then Gen-SA replaces every $s \in G_s$ with its guarding node $gn$ by traversing taxonomy tree $GT$ (lines 4-7). $T^*$ is the result trajectory dataset meeting with $(\alpha, K)_L$-privacy requirement.

### D. COMPLEXITY ANALYSIS

In the first module IEVS I, the sub-module of trajectory preprocessing constructs the trajectory user set $TO$ respecting nonsensitive locations, which needs to scan the dataset once. So the cost is $O(|T| \cdot |t|)$, where $t$ is a trajectory in the trajectory dataset $T$. In the sub-module of identifying MVST, all sequences in each $C_i$ $(1 \leq i \leq L)$ are checked. The size of $C_1$ is the number of distinct nonsensitive locations in $T$. The upper limit of it is not more than $|d|$, the number of dimensions of $T$. The worst cost of checking locations of $C_1$ is $O(|d| \cdot |TO_{nsl}|)$, where $TO_{nsl}$ is the trajectory user set respecting a nonsensitive location $nsl$ in $T$. For $C_2$, the upper

bound of its size is not more than $|d| \cdot (|d| - 1)/2$ due to the Cartesian product of $U_1$ and $U_1$ and the pruning process in Algorithm 1. When we execute an intersection on the trajectory user sets respecting the two locations of a sequence in $C_2$, to count the number of the same users in the two sets, we first merge the two sets into one, then sort it and then count the number. So the cost is approximately $O(|TO_{nsl}|^2)$. The cost of checking sequences of $C_2$ is approximately $O(|d|^2 \cdot |TO_{nsl}|^2)$. When $i \geq 3$, the worst cost of checking $C_i$ is $O(|d|^i \cdot |TO_{nsl}|^2)$. So the worst computational cost of sub-module identifying MVST is $O(|d|^L \cdot |TO_{nsl}|^2)$. In the third sub-module of eliminating MVST, the most costly operation is to check if the MVST in $B$ can be eliminated by trajectory splitting. For each MVST in $B$, we call Algorithm 3 once, which has to visit all records in $T$. The number of MVST in $B$ is also bounded by $|d|^L$. So the cost of eliminating MVST is bounded by $O(|d|^L \cdot |T|)$. Therefore, the cost is $O(|d|^L \cdot |T|)$ in IEVS I module. In the IEVS II module, the most costly operation is to identify all MVST violating sensitive value $\alpha$-privacy from the user set $TO'_{nsl}$ respecting $T'$ created by IEVS I module, which is similar to the sub-module of identifying MVST in IEVS I. So the worst cost is $O(|d|^L \cdot |TO'_{nsl}|^2)$. Our algorithm has an $O(|d|^L \cdot |T|)$ complexity time by incorporating the two modules of trajectory anonymity.

## V. EXPERIMENTS

In this section, we conduct a series of experiments to evaluate the performance of IEVS in terms of data utility and execution time.

### A. EXPERIMENTAL SETUP

In our experiments, we use Brinkhoff generator [34], a road network based generator of moving objects, to generate moving trajectories. The city road map of Oldenburg in Germany is input to the generator. The city map is divided into 100 regions. When a user visits a region of the city, the centroid of the region is used to represent the corresponding location information of the user's trajectory. A trajectory dataset containing 20000 trajectories is generated with the average length 4.7. Seven values of the sensitive attribute are randomly associated with trajectory records and two of them are set sensitive values. The experiments are performed on a machine with Intel Core i7 CPU at 3.60GHz, 8GB RAM, and the algorithms are implemented in VC++.

We use the method of changing the value of a parameter and fixing those of the remaining parameters to conduct our experiments several times on the dataset. After the comparison of the result dataset, the final parameter values are determined. The parameter settings involved in the experiments are described as follows.

- Dataset size $|T|$. The default value of $|T|$ is 20000. When its value varies, the smaller datasets are created by random selection of trajectory records from the original dataset.
- $K$. The default value of $K$ is 10. It varies in [5, 35] in the experiment to evaluate its impact.

SELECT COUNT(*)
FROM trajectories $t$
WHERE $lq \in t$

**FIGURE 3.** An example of COUNT(*) query.

- $L$. It is set to 2 by default. We change $L$ from 2 to 5 to assess its effect on our scheme.
- $\alpha$. When we evaluate the effects of other parameters, its default value is set to 0.5. When assessing the impact of $\alpha$ on our scheme, we take the four values of 0.25, 0.33, 0.5, and 0.75 as the frequency threshold $\alpha$ and averagely assigned to all records of the trajectory dataset.
- Sensitive value size $|SA|$. The domain size of the sensitive attribute is 7. The 7 values are assigned to each trajectory record randomly. Two of the 7 values are set to the sensitive values.
- Sensitive location size. We randomly select 10 from 100 locations as the sensitive locations.

Theoretically, given the same values of thresholds $K$, $L$, and $\alpha$ ($C$ in $(K, C)_L$-privacy model), our $(\alpha, K)_L$-privacy model can provide the same effect on privacy protection of identity and sensitive values with $(K, C)_L$-privacy model. Moreover, our model provide sensitive value protection, which doesn't refer to in $(K, C)_L$-privacy model. To evaluate the data quality of $(\alpha, K)_L$-anonymized trajectory dataset, we compare our scheme with the previous related work KCL-global [9] and KCL-local [8].

### B. DATA UTILITY

We first introduce *Information Loss* (*IL*) and *Average Relative Error* (*ARE*) as the metrics for data utility. Then we compare our IEVS scheme with KCL-global [9] and KCL-local [8] against dataset size, $K$, $L$, and threshold $\alpha$, in term of *IL* and *ARE*.

#### 1) INFORMATION LOSS

Information loss is a basic metric for data utility. It has different forms of definition corresponding to different methods of anonymization. When trajectory anonymization occurs, the less information loss, the more data utility. In our method, we measure information loss by calculating the changes in the number of locations. Specifically, the trajectory information loss $IL_t$ is measured by the ratio of the number of removed locations in the anonymized dataset to original locations, which is calculated as

$$IL_t = \frac{|\,LOC_T\,| - |\,LOC_{T^*}\,|}{|\,LOC_T\,|} \tag{5}$$

where $T^*$ is the $(\alpha, K)_L$-privacy of trajectory dataset with respect to an original dataset $T$. $|LOC_{T^*}|$ and $|LOC_T|$ represent the total numbers of locations in $T^*$ and $T$, respectively.

#### 2) AVERAGE RELATIVE ERROR

Average Relative Error [32], [33], [35] measure is another way to estimate data utility in many anonymity methods.

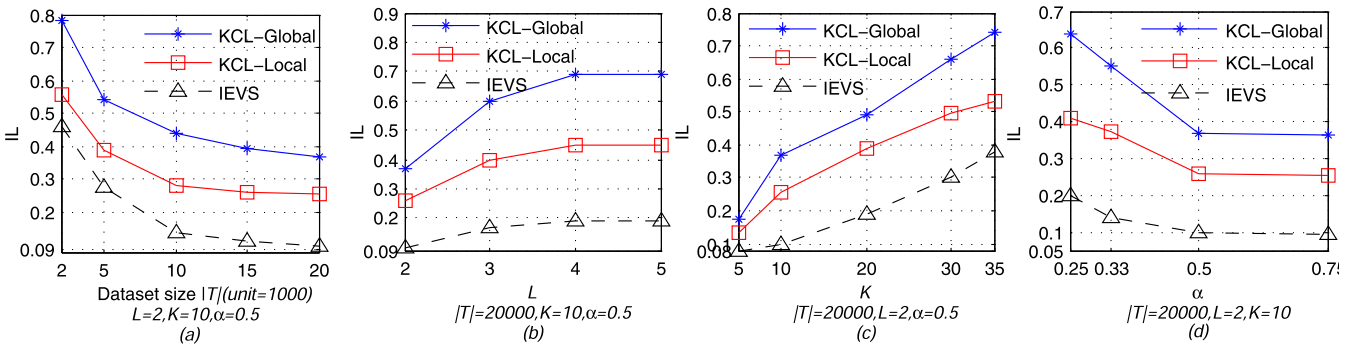


**FIGURE 4.** Information loss versus dataset size, $L$, $K$ and $\alpha$.

Given a count query $Q$, the preserving levels of the original dataset can be evaluated by comparison of the answers of queries on the original and anonymized dataset. Generally, more similar answers on both datasets imply higher data utility of an anonymized dataset. To measure ARE, we design a COUNT(*) query $Q$ (the form of the queries is shown in Fig. 3) on the original and anonymized dataset, which randomly selects 500 different ordered location sequences $LQ$ with length 2. ARE is calculated as

$$ARE_Q = \frac{\sum_{lq \in LQ} \frac{|T(lq)| - |T^*(lq)|}{|T(lq)|}}{|\ LQ\ |} \tag{6}$$

where $lq \in LQ$, $|T^*(lq)|$ and $|T(lq)|$ are the number of trajectories that contain the location sequence $lq$ in the anonymized dataset $T^*$ and the original dataset $T$, respectively.

We first compare our scheme with KCL-Global and KCL-Local on trajectory information loss, in terms of dataset size $|T|$, parameters $K$, $L$ and $\alpha$. For the equal conditions of comparison, we don't consider the temporal information when conducting the algorithms of KCL-Global and KCL-Local for trajectory anonymization, which is the same as what we do in our scheme. Note that parameter $C$ in KCL-Global and KCL-Local is equivalent to $\alpha$ in our scheme.

Fig. 4 shows the algorithm performance respecting trajectory information loss. On the same conditions in these subgraphs of Fig. 4, the information loss of IEVS is very low, which means trajectory splitting can almost eliminate all MVST violating $K$-anonymity and sensitive location $\alpha$-privacy, and thus most of the locations can be published as in the original dataset. The information loss of KCL-Global and KCL-Local is much higher than that of IEVS because KCL-Global and KCL-Local are exclusively based on location suppression for trajectory anonymization. Moreover, unlike suppression employed on trajectories in KCL-Global and KCL-Local against attribute linkage attack (which aggravates the information loss of trajectories), IEVS conducts the method of generalization to protect sensitive values, which relieves the distortion of trajectories. Besides, trajectory splitting in IEVS eliminates the MVST composed of nonsensitive locations, instead of all locations in KCL-Global and KCL-Local.

In Fig. 4(a) we observe that the information loss decreases for larger dataset size, which means more trajectory data records would not increase the number of MVST, it reduces the number of MVST instead and thus, decreases trajectory information loss. Figs. 4(b)-4(c) show that the information loss increases with $L$ and $K$. The reason is, in Fig. 4(b) larger value of $L$ indicates that more MVST generate. As a result, more trajectories are required to be anonymized, which causes more information loss consequently. In Fig. 4(c) the increase of $K$ improves the level of privacy preserving, which creates more locations violating $K-$anonymity. So the information loss increases. Fig. 4(d) shows that the information loss reduces for the increasing value of $\alpha$. The reason is that larger $\alpha$ leads to less MVST and thus, less trajectories are anonymized, which reduces the information loss.

Next, we test the ARE of our scheme and the algorithms of KCL-Global and KCL-Local by varying $|T|$, $K$, $L$, and $\alpha$. Fig. 5 shows that our IEVS scheme has lower ARE than KCL-Global and KCL-Local, which indicates trajectory splitting preserves the number of published locations, while location suppression has a greater impact on co-appearance than splitting. In Fig. 5(a), ARE decreases gradually with dataset size $|T|$. This is because as $|T|$ increases, the instances of subtrajectories increases in size, which improves the probability of the same subtrajectory in two trajectories. Next, in Fig. 5(b) we observe that the ARE improves with the maximum length of nonsensitive location sequences that the adversary has, i.e., $L$. Increasing $L$ causes more MVST and more trajectories need to be anonymized, thus in higher ARE. Then we evaluate ARE against $K$ (Fig. 5(c)). Increasing $K$ leads to more MVST of length 1, so less subtrjaectories of length 2 are preserved in the dataset, which brings higher ARE. Finally the ARE, in Fig. 5(d), decreases against the probability constraint $\alpha$. The reason is that less MVST created with the increase of $\alpha$, so lower number of anonymized trajectories improves the co-occurrence probability of locations.

### C. EFFICIENCY

Fig. 6 presents the run time of IEVS, KCL-Global, and KCL-Local against dataset size $|T|$, $K$, $L$, and probability

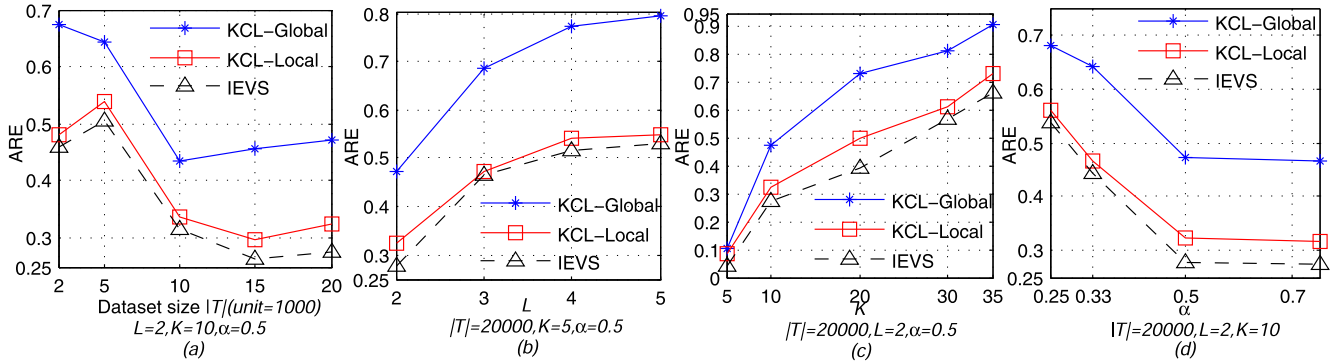


**FIGURE 5.** Average relative error versus dataset size, $L$, $K$ and $\alpha$.

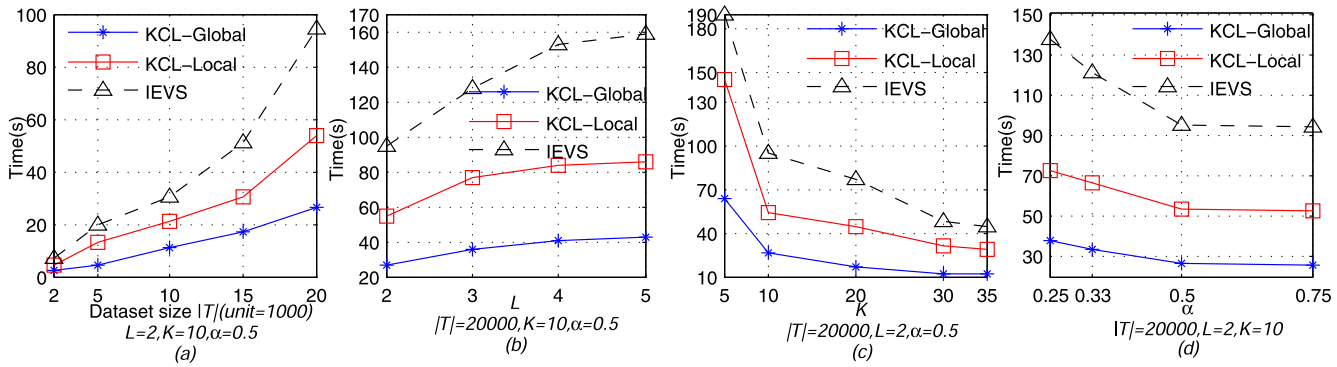


**FIGURE 6.** Execution time versus dataset size, $L$, $K$ and $\alpha$.

constraint threshold $\alpha$. Increasing the dataset size $|T|$, which means more trajectory records need to be evaluated, results in higher time cost (Fig. 6(a)). More time cost is achieved for the higher value of $L$ (Fig. 6(b)), as more MVST generate with the increasing of $L$. In Fig. 6(c), the time cost is less when the value of $K$ increases. Since more MVST of length 1 generate with the increase of $k$, the number of MVST of length 2 reduces, so the time cost is lower. Finally in Fig. 6(d), as expected, with the increasing value of $\alpha$, the runtime is decreased. KCL-Global is the fastest algorithm between the three methods, as it eliminates MVST at each loop due to employing global location suppression and is no need to check whether new MVST are generated.

## VI. CONCLUSION

People in ITS perform all their activities through advanced information communication and technology, which makes it difficult for people to hide their tracks. Personal private information involved in trajectories thus may be leaked. In this paper we study the problem of privacy-preserving trajectory data publishing. To hold back the adversary who can launch subtrajectory linkage attack to infer identities, sensitive locations and values unknown to him, we propose a novel anonymization scheme IEVS that employs trajectory splitting and location suppression on trajectories and generalization on sensitive values, to achieve higher data utility as well as our proposed $(\alpha, K)_L$-privacy requirement. Our experimental results demonstrate the effectiveness of our IEVS scheme in term of data utility.

## REFERENCES

[1] R. A. K. Jayashree and R. Babu, "A collaborative approach of iot, big data, and smart city," in *Big Data Analytics for Smart Connected Cities*, N. Dey and S. Tamane, Eds. Hershey, PA, USA: IGI Global, 2019, pp. 25–37.

[2] X. Chen, J. Ding, and Z. Lu, "A decentralized trust management system for intelligent transportation environments," *IEEE Trans. Intell. Transp. Syst.*, early access, Aug. 13, 2020, doi: 10.1109/TITS.2020.3013279.

[3] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using big data analytics," *Comput. Netw.*, vol. 101, pp. 63–80, Jun. 2016.

[4] E. Naghizade, L. Kulik, E. Tanin, and J. Bailey, "Privacy-and context-aware release of trajectory data," *ACM Trans. Spatial Algorithms Syst.*, vol. 6, no. 1, pp. 1–25, Feb. 2020.

[5] Y. Dong and D. Pi, "Novel privacy-preserving algorithm based on frequent path for trajectory data publishing," *Knowl.-Based Syst.*, vol. 148, pp. 55–65, May 2018.

[6] Z. Huo, Y. Huang, and X. Meng, "History trajectory privacy-preserving through graph partition," in *Proc. 1st Int. Workshop Mobile Location-Based Service (MLBS)*, 2011, pp. 71–78.

[7] O. Abul, F. Bonchi, and M. Nanni, "Never walk alone: Uncertainty for anonymity in moving objects databases," in *Proc. IEEE 24th Int. Conf. Data Eng.*, Apr. 2008, pp. 376–385.

[8] R. Chen, B. C. M. Fung, N. Mohammed, B. C. Desai, and K. Wang, "Privacy-preserving trajectory data publishing by local suppression," *Inf. Sci.*, vol. 231, pp. 83–97, May 2013.

[9] N. Mohammed, B. C. Fung, and M. Debbabi, "Preserving privacy and utility in RFID data publishing," Concordia Univ., Montreal, QC, Canada, Tech. Rep. 6850, 2010. [Online]. Available: https://spectrum.library.concordia.ca/6850/
[10] K. Al-Hussaeni, B. C. M. Fung, and W. K. Cheung, "Privacy-preserving trajectory stream publishing," *Data Knowl. Eng.*, vol. 94, pp. 89–109, Nov. 2014.
[11] Y. Xin, Z.-Q. Xie, and J. Yang, "The privacy preserving method for dynamic trajectory releasing based on adaptive clustering," *Inf. Sci.*, vol. 378, pp. 131–143, Feb. 2017.
[12] L. Sweeney, "$k$-anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness, Knowl. Syst.*, vol. 10, no. 5, pp. 557–570, 2002.
[13] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "$L$-diversity: Privacy beyond $k$-anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, no. 1, p. 3-es, 2007.
[14] N. Li, T. Li, and S. Venkatasubramanian, "Closeness: A new privacy measure for data publishing," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 7, pp. 943–956, Jul. 2010.
[15] R. C.-W. Wong, J. Li, A. W.-C. Fu, and K. Wang, "($\alpha$, $k$)-anonymity: An enhanced $k$-anonymity model for privacy preserving data publishing," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2006, pp. 754–759.
[16] X.-C. Yang, Y.-Z. Wang, B. Wang, and G. Yu, "Privacy preserving approaches for multiple sensitive attributes in data publishing," *Chin. J. Comput.*, vol. 31, no. 4, pp. 574–587, Sep. 2009.
[17] X. Xiao and Y. Tao, "Personalized privacy preservation," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, 2006, pp. 229–240.
[18] X. Liu, Q. Xie, and L. Wang, "Personalized extended ($\alpha$,$k$)-anonymity model for privacy-preserving data publishing," *Concurrency Comput., Pract. Exper.*, vol. 29, no. 6, p. e3886, Mar. 2017.
[19] P.-I. Han and H.-P. Tsai, "SST: Privacy preserving for semantic trajectories," in *Proc. 16th IEEE Int. Conf. Mobile Data Manage.*, Jun. 2015, pp. 80–85.
[20] Y. Dai, J. Shao, C. Wei, D. Zhang, and H. T. Shen, "Personalized semantic trajectory privacy preservation through trajectory reconstruction," *World Wide Web*, vol. 21, no. 4, pp. 875–914, Jul. 2018.
[21] R. Trujillo-Rasua and J. Domingo-Ferrer, "On the privacy offered by (k,$\delta$)-anonymity," *Inf. Syst.*, vol. 38, no. 4, pp. 491–494, Jun. 2013.
[22] D. Lin, S. Gurung, W. Jiang, and A. Hurson, "Privacy-preserving location publishing under road-network constraints," in *Database Systems for Advanced Applications* (Lecture Notes in Computer Science), vol. 5982. Berlin, Germany: Springer, 2010, pp. 17–31.
[23] S. Gao, J. Ma, C. Sun, and X. Li, "Balancing trajectory privacy and data utility using a personalized anonymization model," *J. Netw. Comput. Appl.*, vol. 38, pp. 125–134, Feb. 2014.
[24] D. Kopanaki, V. Theodossopoulos, N. Pelekis, I. Kopanakis, and Y. Theodoridis, "Who cares about others' privacy: Personalized anonymization of moving object trajectories," in *Proc. Adv. Database Technol.*, 2016, pp. 425–436.
[25] S. Shaham, M. Ding, B. Liu, S. Dang, Z. Lin, and J. Li, "Privacy preserving location data publishing: A machine learning approach," *IEEE Trans. Knowl. Data Eng.*, early access, Jan. 7, 2020, doi: 10.1109/TKDE.2020.2964658.
[26] G. Poulis, S. Skiadopoulos, G. Loukides, and A. Gkoulalas-Divanis, "Apriori-based algorithms for $k^m$-anonymizing trajectory data," *Trans. Data Priv.*, vol. 7, pp. 165–194, Aug. 2014.
[27] M. Gramaglia, M. Fiore, A. Tarable, and A. Banchs, "Preserving mobile subscriber privacy in open datasets of spatiotemporal trajectories," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, May 2017, pp. 1–9.
[28] Z. Tu, K. Zhao, F. Xu, Y. Li, L. Su, and D. Jin, "Protecting trajectory from semantic attack considering $k$-anonymity, $l$-diversity, and $t$-closeness," *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 1, pp. 264–278, Mar. 2019.
[29] M. Terrovitis and N. Mamoulis, "Privacy preservation in the publication of trajectories," in *Proc. 9th Int. Conf. Mobile Data Manage. (MDM)*, Apr. 2008, pp. 65–72.
[30] E. Ghasemi Komishani, M. Abadi, and F. Deldar, "PPTD: Preserving personalized privacy in trajectory data publishing by sensitive attribute generalization and trajectory local suppression," *Knowl.-Based Syst.*, vol. 94, pp. 43–59, Feb. 2016.
[31] X. Liu, L. Wang, and Y. Zhu, "SLAT: Sub-trajectory linkage attack tolerance framework for privacy-preserving trajectory publishing," in *Proc. Int. Conf. Netw. Netw. Appl. (NaNA)*, Oct. 2018, pp. 298–303.
[32] G. Poulis, G. Loukides, A. Gkoulalas-Divanis, and S. Skiadopoulos, "Anonymizing data with relational and transaction attributes," in *Machine Learning and Knowledge Discovery in Databases* (Lecture Notes in Computer Science), vol. 8190. Berlin, Germany: Springer, 2013, pp. 353–369.
[33] G. Loukides, A. Gkoulalas-Divanis, and B. Malin, "COAT: COnstraint-based anonymization of transactions," *Knowl. Inf. Syst.*, vol. 28, no. 2, pp. 251–282, Aug. 2011.
[34] T. Brinkhoff, "A framework for generating network-based moving objects," *GeoInformatica*, vol. 6, no. 2, pp. 153–180, 2002.
[35] M. Terrovitis, G. Poulis, N. Mamoulis, and S. Skiadopoulos, "Local suppression and splitting techniques for privacy preserving publication of trajectories," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 7, pp. 1466–1479, Jul. 2017.

**XIANGWEN LIU** is currently pursuing the Ph.D. degree in computer science and technology with Jiangsu University, Zhenjiang, China. Her research interests include privacy-preserving data publishing and data security.

**YUQUAN ZHU** received the Ph.D. degree in computer science and engineering from Southeast University, China. He is currently a Full Professor with the School of Computer Science and Communication, Jiangsu University, Zhenjiang, China. His current research interests include data mining, knowledge discovery, and machine learning.