This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2919833, IEEE Transactions on Dependable and Secure Computing

Sensitive Label Privacy Preservation with Anatomization for Data Publishing

Lin Yao, Zhenyu Chen, Xin Wang, Dong Liu, and Guowei Wu

Abstract—Data in its original form, however, typically contain sensitive information about individuals. Directly publishing raw data will violate the privacy of people involed. Consequently, it becomes increasingly important to preserve the privacy of published data. An attacker is apt to identify an individual from the published tables, with attacks through the record linkage, attribute linkage, table linkage or probabilistic attack. Although algorithms based on generalization and suppression have been proposed to protect the sensitive attributes and resist these multiple types of attacks, they often suffer from large information loss by replacing specific values with more general ones. Alternatively, anatomization and permutation operations can de-link the relation between attributes without modifying them. In this paper, we propose a scheme Sensitive Label Privacy Preservation with Anatomization (SLPPA) to protect the privacy of published data. SLPPA includes two procedures, table division and group division. During the table division, we adopt entropy and mean-square contingency coefficient to partition attributes into separate tables to inject uncertainty for reconstructing the original table. During the group division, all the individuals in the original table are partitioned into non-overlapping groups so that the published data satisfies the pre-defined privacy requirements of our $(\alpha, \beta, \gamma, \delta)$ model. Two comprehensive sets of real-world relationship data are applied to evaluate the performance of our anonymization approach. Simulations and privacy analysis show our scheme possesses better privacy while ensuring higher utility.

Index Terms—Privacy preservation, Anatomization, Sensitive label

1 Introduction

The collection of digital data by governments and corporations has created tremendous opportunities for knowledge-based decision making. More and more application domains rely on these published data to provide enriched services to users, for applications such as marketing, social psychology, and homeland security [1]. The popularity of such applications also raises the challenge in determining the right accessibility of published data. On the one hand, it is beneficial to release data publicly for data mining and analysis activities. For example, various agencies and organizations often publish the data on public health for demographic research or other purposes. On the other hand, the publication of data may entail a privacy threat for the users if some sensitive information is released. According to a recent study, approximately 87% of the population in the United States can be identified by a given data set [2]. Therefore, it is critical to conserve the privacy of published data, especially the sensitive information.

The original data typically has four types of attributes: explicit identifier, quasi-identifier, sensitive attribute, and non-sensitive attribute [3]. *Explicit Identifier (EI)* is used to identify an individual uniquely, and is thus often removed from the published tables. Although each specific *Quasi-Identifier (QI)* cannot uniquely identify an individual, the combination of a few QIs can possibly achieve the goal. *Sensitive Attribute (SA)* contains the private or specific information of each individual. *Non-sensitive attribute* can be known for the public without any concern. Based on the background knowledge of QIs or SAs, an attacker is likely to launch the following attacks, record linkage, attribute linkage, table linkage and probabilistic attack [4]. With a record linkage attack, a target user can be identified from a specific record inside the published tables. Attribute linkage occurs when some attributes (e.g., salary and disease) are revealed, and these attributes can be linked to an individual. Table linkage aims to infer that the victim's record exists in the published data. In the probabilistic attack, the probabilistic belief on the victim's SA may be different after the published data is accessed.

1.1 Motivation

There is a challenge to protect users' privacy and prevent the identity disclosure of users from the published tables. To provide these protections, some original records should be modified before being published through some anonymization operations. k-anonymity is a popular methodology to protect the privacy with the relational data published. It is first adopted in [5] to protect the released data from disclosure by anonymizing each individual from at least $k-1$ other ones. That is, even if a user's QIs are known, the probability of identifying the user is not larger than $1/k$. Generalization and suppression are two common methods to achieve k-anonymity [4] [3]. In the generalization, some attribute values are replaced by a broader category such as a parent value in the taxonomy of an attribute. In the suppression, certain values of the attributes are replaced by

- *Lin Yao, Zhenyu Chen Dong Liu, and Guowei Wu are with Dalian University of Technology, key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, China. Lin Yao is with the DUT-RU International School of Information Science & Engineering, Dalian University of Technology. Zhenyu Liu, Dong Liu and Guowei Wu are also with the School of Software, Dalian University of Technology.*
- *Xin Wang is with Department of Electrical and Computer Engineering, Stony Brook University, New York State, U.S.A.*

special symbolic characters such as $\star$ and $\#$. As another method, perturbation distorts a dataset by adding noise, exchanging values or generating synthetic data while keeping some statistical properties [2]. Despite that these methods help protect the user privacy, their modification of QIs and SAs often result in considerable information loss [6], which largely reduces the accuracy of data analysis.

Unlike generalization, suppression and perturbation, anatomization and permutation do not make any modification of QIs or SAs, but de-associate the correlation between attributes by separating them into different tables or data sets. Extensive experiments conducted in [7] [8] [2] have proven that anatomy significantly outperforms the other techniques in both the effectiveness of data analysis and the computation cost. Not only that the anatomized tables permit highly accurate aggregate search with lower average error, but also the query accuracy of anatomy does not decay severely as the dimensionality increases. Among all the approaches based on anatomy [8] [9] [10] [11], only slicing [8] considers the correlation among attributes during the partitioning of data horizontally and vertically. However, the clustering algorithm taken by slicing causes the uneven distribution of attributes for achieving vertical partition, which causes the privacy leakage. Consequently, slicing can only resist attribute linkage attack and table linkage attack.

To address the above challenges, this paper aims to design an effective data anonymization approach that the published data are usable for accurate analysis while preventing the disclosure of sensitive data labels. Our scheme will protect the privacy of users and ensure that published data can resist the record linkage attack, attribute linkage attack, table linkage attack and probabilistic attack. To achieve our goal, we propose two major techniques, *table division* and *group division*. During the table division, the original table is partitioned vertically by dividing attributes into different tables based on their QI weights and the correlation between QIs. As entropy theory is considered to be an objective way for weight determination [12], we use the entropy of each QI to guide our table division that a set of QIs will not be put into the same table if this will release more secrete information. Furthermore, we try to ensure that the correlation between attributes is maximized in the same table but minimized in different ones. After the table division, the original data are separated into different tables. During the group division, all records in the original table will also be partitioned into non-overlapping groups and the group identifier will be added into the corresponding records in the separate tables so that each group in the published tables can satisfy the pre-defined four privacy requirements of our proposed $(\alpha, \beta, \gamma, \delta)$ model.

1.2 Contributions and Organization

In this paper, we propose a novel scheme to preserve the privacy of data with a single SA, named Sensitive Label Privacy Preservation based on Anatomization for publishing tabular data (SLPPA). Given all the above considerations, this paper has the following contributions:

- We design and implement an anatomization technique to publish data with two processes: table division and group division. To the best of our knowledge, our scheme is the first that performs table division based on both the QI weight and the correlation between QIs. In addition, we design and implement two algorithms for table division and group division to meet the requirements of our $(\alpha, \beta, \gamma, \delta)$ privacy model.
- We formalize four types of privacy requirements in our $(\alpha, \beta, \gamma, \delta)$ model for the published social data to resist the record linkage, attribute linkage, table linkage and probabilistic attack. Our privacy analysis demonstrates that this model can protect record privacy, sensitive privacy, presence privacy, and probabilistic privacy. The four parameters can be predefined by the data owner before data anonymization.
- We evaluate the performance through extensive simulations based on two real-world data sets. Compared with **Slicing** [8], the time consumption of SLPPA is reduced to less than $\frac{1}{100}$ that of **Slicing** while keeping a high utility. Compared with **ASN** [6], our SLPPA has a better data utility and privacy. In addition, compared with **ASN** and **Slicing**, SLPPA has more even number of attributes in different tables and groups.

The remainder of this paper is organized as follows: In Section 2, we discuss the related work. Privacy attacks and privacy model are introduced in Section 3. In Section 4, we present the details of our approach. Utility and privacy analysis are given in Section 5. Simulations are presented in Section 6. Finally, we conclude our work in Section 7.

2 Related Work

Existing approaches to prevent the privacy leakage of the published data are categorized into the following set of anonymity operations: generalization and suppression, anatomization and permutation, and perturbation [4] [13]. Algorithms for privacy preserving in data publishing differ in their choices of anonymity operations.

Generalization and Suppression. Generalization and suppression are the most common anonymity operations used to implement k-anonymity for privacy protection. Generalization replaces some QI values with a broader category such as a parent value in the taxonomy of an attribute. To reach k-anonymity, Samarati [14] [15] [16] adopted the idea of full-domain generalization and the QI value was generalized to the same level in a given taxonomy tree structure. In [17] [18], a node in the taxonomy tree structure was generalized to its parent node. A minimal full-domain generalization was proposed to achieve k-anonymity with the siblings of the generalized node remaining unchanged [19]. Suppression replaces some attributes with special symbolic characters such as $\star$ and $\#$ to anonymize the publishing data [20] [21] [22]. In most situations, generalization is combined with suppression. In [23] [24] [25], a top-down greedy heuristic algorithm was used to choose the best specialization attributes to anonymize the QIs according to user requirements. In [26], record linkage and attribute linkage were prevented through the generalization of quasi-identifiers by setting the range values and record elimination. In [27], authors adopted the idea of clustering and multi-sensitive bucketization to publish microdata with multiple numerical sensitive attributes. The attribute was selected automatically for generalization and suppression based on the coefficient between attributes in [28]. In [29], sensitive attribute generalization and trajectory local suppression were combined to achieve a tailored personalized privacy model for trajectory data publishing.

Anatomization and Permutation. Anatomization and permutation aim to de-link the relation between attributes without modifying them. The anatomization approaches dissociate the correlation between QIs and SAs and generate several separate tables with non-overlapping attributes. Permutation shares the same principle of anatomization and is used to de-associate the relationship between a quasi-identifier and a numerical sensitive attribute by dividing the records into groups and shuffling their sensitive values within each group [30] [31]. Anatomy [7] releases all the quasi-identifiers and sensitive information directly in two separate tables and both tables have one common attribute, the group identifier. Based on the idea of anatomy, anonymized groups are generated to preserve the structural and tabular data utility of the social network [6] [32]. A linear-time algorithm for computing anatomized tables is designed to meet the ℓ-diversity privacy requirement of sensitive attribute combining with a grouping mechanism in the published social graph [33]. A new approach, called Slicing, is proposed in [8] to preserve the privacy of published data. It satisfies the privacy requirement of ℓ-diversity by partitioning attributes into columns, where each column contains a subset of attributes. Slicing also partitions the tuples into buckets and each bucket contains a subset of tuples. In [9], Slicing was adopted to preserve the privacy of published data, where each attribute has one column. In [10], a novel method named t-closeness slicing is designed to better protect transactional data against various attacks. The anonymization algorithm in [11] takes advantage of both anatomization and enhanced slicing to protect the privacy of multiple sensitive attributes, while obeying the principles of k-anonymity and l-diversity. To resist the presence attack, an improved version of anatomy called permutation anonymization [34] partitions the original table into groups to satisfy the ℓ-diversity. In [35], [36], a light-weight data privacy method is proposed to use a pseudo random permutation to scramble the original data.

Perturbation. Perturbation tries to protect the privacy by distorting the data set while keeping some statistical properties. Perturbation distorts the data by adding noise, swapping values, or generating synthetic data [4]. Adding noise was used in [37] [38] [39] [40] [41]. An method based on ϵ-differential privacy is introduced in [42] to protect the privacy of the record owner by removing or adding a single record in the published data to resist the probabilistic attack. To achieve ϵ-differential privacy in [43], each row of an adjacency matrix was projected into a low dimensional space using random projection, and then the projected matrix was perturbed with random noise. A randomization algorithm for data sanitization was also proposed to make the published data satisfy the ϵ-differential privacy in [44]. With data swapping, sensitive attribute values are exchanged among records to protect the privacy of statistical information [45]. Synthetic data generation replaces the original data with some samples points which are from a pre-defined statistical model. Random edge perturbation was used to resist structural identification attack and protect the publishing data privacy in [46]. In [47], the authors aimed to modify the shortest path length in graphs and maintain the structure of the graph to protect the sensitive information. Condensation was an alternative synthetic data generation approach to preserve the privacy [48] [49].

Summary of Related Work. Techniques based on generalization, suppression, or perturbation often modify the attribute values, which may reduce the usability of data for analysis. To avoid the issue, we adopt anatomization technique in this paper. Although Slicing [8] can achieve vertical partition by grouping attributes into columns based on the correlation among the attributes and horizontal partition by grouping randomly permutating or sorting sensitive attribute in each column to break the linkage between different columns, it cannot protect the privacy of the published data as it does not consider record linkage attack and probabilistic attack. Other approaches [9] [10] [11] based on Slicing consider each attribute as a column by ignoring the correlation between attributes. To address these challenges, we are the first to take into account both the QI weight and the association between QIs during the table division to maximize the correlation between attributes in the same table and minimize the correlation in different tables. In addition, we implement group division to make the published data meet the four types of privacy requirements specified by our $(\alpha, \beta, \gamma, \delta)$ model to prevent the record linkage, attribute linkage, table linkage and probabilistic attack respectively. Slicing, however, can only resist attribute linkage and table linkage.

3 Privacy Attacks and Privacy Model

In this section, we first introduce four privacy attacks and then present our privacy model.

Most of published data are stored in the tabular format such as that in Table 1. Each individual in Table 1 contains three kinds of attributes (EI, QIs and SA). The EI *Name* can uniquely identify an individual. QIs include *Gender*, *Job*, *Age* and *Zipcode*. The SA *Salary* should be protected.

TABLE 1
The Tabular Data of Financial Transaction Network

Name	Gender	Job	Age	Zipcode	Salary
Alice	F	Teacher	30	11100	4500
Ben	M	Engineer	33	12200	6000
Cary	F	Engineer	45	12200	4700
David	M	Teacher	42	11100	4500
Eric	M	Doctor	32	25300	6700
Frank	M	Police	44	23200	4000
Gina	F	Doctor	30	11100	6400
Henry	M	Doctor	33	12200	6000

Preserving data privacy is an essential task in order to allow such data to be published for different research and analysis purposes. In this paper, we focus on protecting the privacy of individuals in a table T, and our main goal is to avoid two categories of attacks [4]. In the first category, an attacker focuses on linking records, attributes, or tables to find an exact target victim. The attacking methods are called respectively as record linkage, attribute linkage and table linkage. In the second category, an attacker can not link records, attributes or tables to a target victim exactly, but the probabilistic belief of the attacker on the sensitive information of the target victim may be changed according to the background knowledge and the published data. This is usually referred as the probabilistic attack. To explain the attacks, we assume Table 2 is an example illustrating various attacks, where EI has been removed for each record.

TABLE 2
An Example Illustrating Various Attacks

Gender	Job	Age	Zipcode	Salary
F	Teacher	30	11100	4500
M	Engineer	33	12200	6000
F	Enginner	45	12200	4700
M	Teacher	42	11100	4500
M	Doctor	32	25300	6700
M	Police	44	23200	4000
F	Doctor	30	11100	6400
M	Doctor	33	12200	6000

1) **Record linkage attack.** An attacker could identify a tuple or a small number of tuples from the published anonymous data T^* according to the QIs of the target victim. For example, if an attacker knows that Cary is an engineer with the age over 40 and Eric is a doctor with the age over 30, he can deduce that the 3-th tuple belongs to Cary and Eric's tuple is the 5-th or 8-th in Table 2.
2) **Attribute linkage attack.** The attacker may not precisely identify the victim's tuple, but could infer the victim's sensitive information from T^* according to the category the victim belongs to. For example, if an attacker only knows that Alice is a teacher, he can not uniquely identify Alice's record. However, he can infer that Alice's salary must be 4500 from Table 2.
3) **Table linkage attack.** Both record linkage and attribute linkage are based on the assumption that the attacker has known the presence of the victim's record in T^*. Table linkage aims to infer whether the victim's record is present in T^*. For example, we assume that a strong attacker can collect all the information in Table 1 except the SA. If Frank's record is removed from Table 2, the attacker can easily infer that Frank is not in T^* because no police exists in the attribute *Job*.
4) **Probabilistic attack.** The above three attacks focus on records, attributes, and tables linking to a target victim exactly. The probabilistic attack means that an attacker can infer some information from the difference between the prior and posterior beliefs. For example, if an attacker knows that the prior belief of $Salary = 4500$ in Table 1 is $\frac{2}{8} = \frac{1}{4}$. If the 1-th and 4-th tuples that correspond to $Salary = 4500$ are removed from Table 2, the posterior belief of $Salary = 4500$ will be 0. Then, the attacker can deduce that Table 2 may be not the anonymized table of Table 1. In order to resist this attack, the difference between the prior and posterior beliefs has to be small [50].

To resist the above attacks, only removing explicit identifier such as *Name* is often not enough to protect against privacy disclosure. QAs and SA may reveal some private information. Therefore, we define our $(\alpha, \beta, \gamma, \delta)$ privacy model in this paper. Given a network's tabular data T, the anonymous data $T^* = \{g_1, g_2, \cdots, g_k\}$ consists of different groups g_j. We say T^* satisfies $(\alpha, \beta, \gamma, \delta)$ privacy model if each group g_j satisfies the $\alpha - record - association$, $\beta - sensitive - association$, $\gamma - presence$ and $\delta - probability$ requirements. To illustrate the meanings of the parameters α, β, γ and δ, we will show some examples using the information from the Table 3, which is anonymized from Table 1, and uses *GID* as the group identifier.

Definition 1 ($\alpha - record - association$) T^* satisfies $\alpha - record - association$ if the probability of inferring QIs of each individual v_i satisfies $Pr[AQI(v_i)] \leq \alpha$ for $0 \leq \alpha \leq 1$, where $AQI(v_i)$ represents v_i's QIs.

In Table 1, the QIs of Alice is $AQI(Alice) = \{F, Teacher, 30, 11100\}$. After anonymization, the original records are split into different groups. Alice belongs to g_1 in Table 3. Table 3(a) has four tuples associated with g_1: $(30, 11100)$, $(33, 12200)$, $(45, 12200)$, and $(42, 11100)$. Table 3(b) also has four tuples $(F, Teacher)$, $(M, Engineer)$, $(F, Engineer)$, and $(M, Teacher)$ associated with g_1. There are $4! = 24$ combinations from the two tables to reconstruct QIs of members in g_1. One example combination is

$$\{(F, Teacher, 30, 11100), (M, Engineer, 33, 12200),\\ (F, Engineer, 45, 12200), (M, Teacher, 42, 11100)\}.$$

As the number of combinations that contain $AQI(Alice)$ is $3! = 6$, the probability of inferring QIs of Alice is $Pr[AQI(Alice)] = \frac{6}{24} = \frac{1}{4}$.

Definition 2 ($\beta - sensitive - association$) T^* satisfies $\beta - sensitive - association$ if the probability of inferring SA of each individual v_i satisfies $Pr[ASA(v_i)] \leq \beta$ for $0 \leq \beta \leq 1$, where $ASA(v_i)$ is used to represent v_i's SA.

To explain $Pr[ASA(v_i)]$, for Alice, $ASA(Alice) = \{4500\}$ belongs to g_1 in Table 3(c) has two counts. Thus we have $Pr[ASA(Alice)] = \frac{2}{4} = \frac{1}{2}$.

Definition 3 ($\gamma - presence$) T^* satisfies $\gamma - presence$ if for each individual v_i, the probability of inferring that v_i belongs to T^* satisfies $Pr[v_i \in T^*] \leq \gamma$ with $0 \leq \gamma \leq 1$.

To explain $Pr[v_i \in T^*]$, we first find the probability of each combination in g_1,

$$Pr[(F, Teacher, 30, 11100, 4500)] = \frac{1}{4} \times \frac{1}{4} \times \frac{2}{4} = \frac{1}{32}.$$

Similarly, we can get the probability of each combination in g_2,

$$Pr[(M, Doctor, 33, 12200, 6700)] = \frac{1}{4} \times \frac{2}{4} \times \frac{1}{4} = \frac{1}{32}.$$

In summary, $Pr[v_i \in T^*]$ must be smaller than the probability of the most frequent combination among all the groups.

Definition 4 ($\delta - probability$) T^* satisfies $\delta - probability$ if for each individual v_i, the probability difference of v_i' SA before and after the anonymization in each group meets $\mid Pr[ASA(v_i)] - Pr[ASA(v_i) \mid g_j] \mid \leq \delta$ with $0 \leq \delta \leq 1$.

For $ASA(Alice) = \{4500\}$, $Pr[4500] = \frac{2}{8}$ holds in Table 1. After anonymization, we can compute $Pr[4500|g_1] = \frac{2}{4}$ and $Pr[4500|g_2] = 0$ from Table 3(c). Then, the probability difference of inferring Alice's SA before and after the anonymization $\mid Pr[ASA(Alice)] - Pr[ASA(Alice) \mid g_1]$ is $|\frac{2}{8} - \frac{2}{4}| = \frac{1}{4}$ and $\mid Pr[ASA(Alice)] - Pr[ASA(Alice) \mid g_2]$ is $|\frac{2}{8} - 0| = \frac{1}{4}$.

TABLE 3
An Example of Published Table

(a) QI table

GID	Age	Zipcode	Count
1	30	11100	1
1	33	12200	1
1	45	12200	1
1	42	11100	1
2	32	25300	1
2	44	23200	1
2	30	11100	1
2	33	12200	1

(b) QI table

GID	Gender	Job	Count
1	F	Teacher	1
1	M	Engineer	1
1	F	Engineer	1
1	M	Teacher	1
2	M	Doctor	2
2	M	Police	1
2	F	Doctor	1

(c) SA table

GID	Salary	Count
1	4500	2
1	6000	1
1	4700	1
2	6700	1
2	4000	1
2	6400	1
2	6000	1

Given a dataset T and four privacy parameters α, β, γ and δ, our goal is to anonymize T into T^* with an insurance of privacy by meeting the $(\alpha, \beta, \gamma, \delta)$ privacy model. These four parameters can be pre-defined by the data owner before the anonymization.

4 Sensitive Label Privacy Preservation with Anatomization (SLPPA)

Our main research goal is to protect the privacy of the published data from the background knowledge attacks while not compromising the utility of published data. In this section, we first introduce our basic framework and then elaborate the details of our SLPPA scheme.

4.1 Overview

Our SLPPA scheme includes two processes, table division and group division, as shown in Fig.1. By separating the original table into multiple tables, we can increase the uncertainty of reconstructing the original table from the published tables. By separating the original table into multiple groups, we can ensure that each group in the published tables can meet our $(\alpha, \beta, \gamma, \delta)$ privacy model and thereby resist the above four types of attacks. SLPPA follows a sequence of procedures to conserve the privacy of the published data:

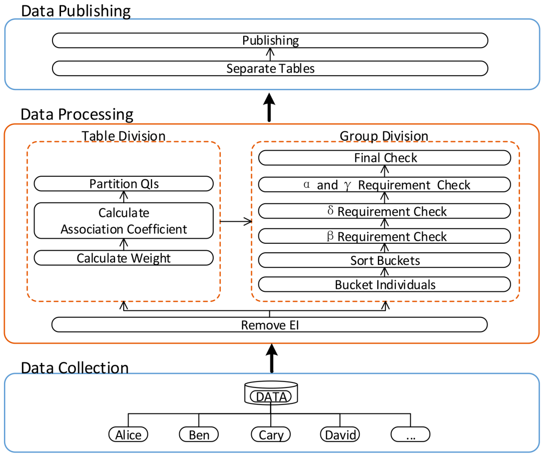


Fig. 1. Architecture of our SLPPA

TABLE 4
An Example of Table Division

(a) QI table

Age	Zipcode	Count
30	11100	2
33	12200	2
45	12200	1
42	11100	1
32	25300	1
44	23200	1

(b) QI table

Gender	Job	Count
F	Teacher	1
M	Engineer	1
F	Engineer	1
M	Teacher	1
M	Doctor	2
M	Police	1
F	Doctor	1

(c) SA table

Salary	Count
4500	2
6000	2
4700	1
6700	1
4000	1
6400	1

1) EI is first removed from the original Table 1 to generate Table 2.
2) During the table division, the SA is first put into a separate table, and then the QIs are partitioned into several other tables. We adopt the concept of entropy [51] to evaluate the QI weight and mean-square contingency coefficient [8] to determine the association strength between QIs. With these operations, Table 2 is divided into a few tables as shown in Table 4.
3) During the group division, we first divide the original data into different buckets each containing records of the same SA value, and rank the buckets in a descending order based on the bucket size. When dividing individuals in the buckets into different groups, we consider the four privacy parameters.
4) After individuals are divided into different groups, their group identifiers ($GIDs$) will be added into the tables which contain different attributes. Finally, these tables will be published as shown in Table 3.

4.2 Privacy Requirement

As discussed before, the group division must ensure that each group satisfies the pre-defined privacy requirements of our $(\alpha, \beta, \gamma, \delta)$ model. In this sub-section, we define α, β, γ and δ privacy requirements. To explain them clearly, we take Table 3 as an example.

α **Requirement:** For each group, the probability of inferring the right QIs of v_i, $Pr[AQI(v_i)]$, is smaller than α.

We define c_j^1 and c_j^2 as the number of times the most frequent records appear in the same group g_j of the two tables. We can draw a conclusion that the probability of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2919833, IEEE Transactions on Dependable and Secure Computing

inferring the right $AQI(v_i)$ in g_j must be smaller than the most frequent combinations in g_j, with

$$Pr[AQI(v_i)] \leq Max(\frac{c_j^1}{n_j}, \frac{c_j^2}{n_j}), \tag{1}$$

where n_j is the number of individuals in g_j and the function Max is used to return the largest value from two elements, $\frac{c_j^1}{n_j}$ and $\frac{c_j^2}{n_j}$.

If there are k separate tables for QIs, we assume that $c_j^1, c_j^2, \cdots, c_j^k$ are the maximum number of repetitions in the corresponding tables. Equation (1) can be modified as

$$Pr[AQI(v_i)] \leq Max(\frac{c_j^1}{n_j}, \ldots, \frac{c_j^k}{n_j}). \tag{2}$$

To satisfy the $\alpha - record - association$ privacy, each g_j must satisfy

$$Max(\frac{c_j^1}{n_j}, \ldots, \frac{c_j^k}{n_j}) \leq \alpha \quad s.t. \quad \forall g_j \in T^*. \tag{3}$$

β **Requirement:** For each group, the probability of inferring the right SA of v_i, $Pr[ASA(v_i)]$, is smaller than β.

With the sensitive attribute stored in a separate table, $Pr[ASA(v_i)]$ can not be greater than the ratio between c_j (i.e., the number of times the most frequent record appears in the table) and the total number of records in g_j:

$$Pr[ASA(v_i)] \leq \frac{c_j}{n_j}. \tag{4}$$

To satisfy the $\beta - sensitive - association$ privacy, each g_j formulation must satisfy

$$\frac{c_j}{n_j} \leq \beta \quad s.t. \quad \forall g_j \in T^*. \tag{5}$$

γ **Requirement:** The probability of inferring that v_i belongs to a group is smaller than γ.

After k separate tables for QIs and one table for SA are published, there are totally $(n_j)^{k+1}$ combinations of records in each g_j. Then, the probability of inferring that v_i belongs to g_j must be smaller than the ratio between the most frequent combination in g_j and the total number,

$$Pr[v_i \in g_j] \leq \frac{c_j^1 \times c_j^2 \cdots \times c_j^k \times c_j}{(n_j)^{k+1}}. \tag{6}$$

To satisfy the $\gamma - presence$ privacy, each g_j must satisfy

$$\frac{c_j^1 \times c_j^2 \cdots \times c_j^k \times c_j}{(n_j)^{k+1}} \leq \gamma \quad s.t. \quad \forall g_j \in T^*. \tag{7}$$

δ **Requirement:** For each group, the difference between the prior and posterior beliefs that an individual possesses the SA value must be smaller than δ.

$$\mid Pr[ASA(v_i)] - \frac{c_j^s}{n_j} \mid \leq \delta \quad s.t. \quad \forall g_j \in T^*, \tag{8}$$

where c_j^s is the number of times that an SA value appears in g_j.

4.3 Algorithm SLPPA

To protect the SA privacy, anatomization de-associates the correlation between attributes by separating them into different tables without modification. In our design, we first divide the attributes into multiple tables based on the QI weight and the association between QIs, and then complete the group division as shown in Fig.1.

4.3.1 Table Division

Table division aims to maximize the association between attributes in the same table and minimize the association between attributes in the different tables. For the example of Table 1, we divide attributes into three tables, two for QIs and one for the SA. As shown in Fig.1, table division includes three steps, calculating the QI weight, calculating the association coefficients between QIs and partitioning QIs.

Step 1: First, we apply entropy to measure the weight of each QI. Entropy is the expected value (average) of the information contained in each message received [51]. A QI with a higher entropy can provide more amount of useful information. Based on this QI, the attacker is more likely to deduce the individuals' privacy. The weight of each QI is calculated as

$$W_{Q_r} = -\sum_{i=1}^{m_r} p(a_{ri}) log(p(a_{ri})), \tag{9}$$

where Q_r is a QI, $\{a_{r1}, a_{r2}, \cdots, a_{rm}\}$ is a set of possible values for Q_r, $p(a_{ri})$ is the chance that a_{ri} is taken, and m_r is the number of distinct values for Q_r. Finally, we can range the attributes in a descending order according to their weights.

Step 2: We adopt the *Mean-square contingency coefficient* [8], [52] to measure the association between QIs. The association coefficient between two QIs is calculated through

$$\phi^2(Q_1, Q_2) = \frac{\sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \frac{(p(a_{ij}) - p(a_{i\cdot})p(a_{\cdot j}))^2}{p(a_{i\cdot})p(a_{\cdot j})}}{min\{m_1, m_2\} - 1}, \tag{10}$$

where m_1 and m_2 are the corresponding number of distinct values of Q_1 and Q_2. $p(a_{ij})$ is the chance that a_{ij} is taken. $p(a_{i\cdot})$ and $p(a_{\cdot j})$ are the marginal totals of $p(a_{ij})$ where $p(a_{i\cdot}) = \sum_{j=1}^{m_2} p(a_{ij})$ and $p(a_{\cdot j}) = \sum_{i=1}^{m_1} p(a_{ij})$.

Step 3: Finally, we partition the QIs into separate tables. The number of tables can be set by users. In our example, we divide the QIs into two tables. If two QIs with higher weights are put into the same table, it will provide more useful information for attackers to deduce an individual's privacy. To prevent this, we first select two QIs with the maximum weight values as the first column attributes in the two different tables. Then, we pick out a QI each time according to the descending order of the QI weights, and put it into the table where it has the maximum average association coefficient with the other columns. Our aim is to ensure that the association coefficient between attributes in the same table is maximized, but the coefficient for attributes in different tables is minimized. In addition, we don't want the amount of information provided by different tables to be too large.

With the above three steps, we get three tables in Table 4. Only Table 6 includes one SA, $Salary$. Table 4(a) and

Table 4(b) include two QIs respectively, $Age, Zipcode$ and $Gender, Job$. The last column $Count$ is used to show the number of times the corresponding attribute tuple appears in the original table.

4.3.2 Group Division

After the table division, the original data are separated into different tables. Group division is then applied to divide the original data into different groups to make each group satisfy $(\alpha, \beta, \gamma, \delta)$ privacy model. Each individual only belongs to one group. Algorithm 1 shows how the group division can be pursued.

Step 1**:** On lines $1-2$, we first divide all the original records into different buckets according to the SA value. Records with the same SA value are divided into the same bucket. Then, we range the buckets in a descending order based on the bucket size and generate a list $BuckList$.

Step 2**:** On lines $3-15$, all the original records are divided into different groups where each group must meet the $(\alpha, \beta, \gamma, \delta)$ privacy model. Because we consider only one SA and multiple QIs, we fist consider β and δ requirements related to SA. The details are as follows:

1) On line $3-4$, we initialize the group index g_j and the counter of each bucket, where the counter indicates the number of groups that have different SAs within the bucket.
2) On lines $5-11$, in order to satisfy β and δ requirements, we first pick out N different buckets in the $BuckList$ from the tail to head with $N = Max(\lceil \frac{1}{\beta} \rceil, \lceil \frac{1}{\delta} \rceil)$. Then, we select one record from each bucket and put it into g_j. When there are N different values of SA in g_j, a group g_j satisfying β and δ requirements is generated.
3) The above steps are repeated until the number of buckets in the $BuckList$ is less than N.

Step 3**:** On lines $13-18$, we add each record in the rest of $BuckList$ to one group which possesses different SA values until no reord is left. Even if there is one record left, it indicates that β and δ requirements cannot be satisfied. In this case, we should reset β or δ to make N smaller.

Step 4**:** On lines $20-41$, we check whether all the groups formed with the above steps satisfy α and γ requirements by starting from the first group. Once one group such as g_i cannot satisfy the requirements, we must combine g_i with one or a few groups from the last to the first until the combined group satisfies α and γ requirements. Finally, if any group cannot satisfy the privacy requirements, α and γ must be reset.

Algorithm 1 Group Division

```
Input: tabular data $T$, $\alpha, \beta, \gamma, \delta$
Output: groups $g_1, g_2, \cdots, g_k$
     ▷ Bucket Individuals
 1: divide individuals into $buckets$ according to $SA$
 2: $BuckList \leftarrow$ Sort-descending($buckets$)
     ▷ Group Individuals
 3: $j \leftarrow 1$
 4: for all $buckets$, $counter \leftarrow 0$
 5: $N \leftarrow max(\lceil \frac{1}{\beta} \rceil, \lceil \frac{1}{\delta} \rceil)$
 6: while $buckets$ in $BuckList > N$ do
 7:     pick out $N$ $buckets$ from $BuckList$ from tail to head
 8:     for each above $bucket$, pick out one individual
 9:     for other $buckets$, $counter \leftarrow counter + 1$
10:     put all individuals into $g_j$
11:     j = j+1
12: end while
13: for $bucket_i$ in $BuckList$ do
14:     if $size(bucket_i) \leq counter_i$ then
15:         put all records of $bucket_i$ into groups have different
            $SA$s within $bucket_i$
16:     else
17:         break and reset $\beta$ or $\delta$ to make $N$ smaller
18:     end if
19: end for
20: for $g_j$ in groups do
21:     if $g_j$ satisfies $\alpha$ and $\gamma$ Requirements then
22:         $j = j + 1$ and break
23:     else
24:         $k \leftarrow l, l \leftarrow$ number of groups
25:         while l $> 1$ do
26:             if $k \neq j$ then
27:                 combine $g_j$ with $g_k$
28:                 if $g_j$ satisfies $\alpha$ and $\gamma$ Requirements then
29:                     $j = j + 1, l = l - 1$ and break
30:                 else
31:                     $k = k - 1, l = l - 1$
32:                 end if
33:             else
34:                 $k = k - 1$
35:             end if
36:         end while
37:     end if
38: end for
39: if $g_1$ cannot satisfy $\alpha$ and $\gamma$ Requirements then
40:     reset $\alpha$ or $\gamma$
41: end if
```

4.4 Parameter Setting

As α, β, γ or δ is the lower bound of the corresponding privacy requirement, different data owners may have different privacy requirements. In one case, one person may set each parameter to 0.05 in order to meet the 95% confidence level. In another case, α is only set to be 0.5 to resist record linkage attack if Sex as one kind of QI represents male or female. In this section, we will give some instructions on how to set proper parameters.

To achieve β and δ requirements, we make $N = Max(\lceil \frac{1}{\beta} \rceil, \lceil \frac{1}{\delta} \rceil)$ to guarantee that the leakage probability of each SA value is less than $\frac{1}{N}$ with the total of M different SA values and $N \leq M$. During the group division, we first pick up N different buckets in the $BuckList$ from the tail to the head. After performing **Step 2**, we assume there are R buckets left. According to **Step 3**, each record in the R-th bucket will be put into the group with different SA values to achieve the β requirement. The number range of available groups is between $\lfloor \frac{M-R}{N} \rfloor$ and $\binom{M-R}{N}$. When the number of records left in the R-th bucket is smaller than $\lfloor \frac{M-R}{N} \rfloor$, there will be no record left. Otherwise, we can assert that the group division will not converge. In this case, we should reset β or δ.

Based on the definition of δ requirement, we can infer that $Pr[ASA(v_i)]$ is no more than $\frac{H}{L}$, where H is the number of most frequent SA value and L is the total number of records in the original data. We can easily get $H \leq G \leq \frac{L}{N}$ and $Pr[ASA(v_i)] \leq \frac{H}{L} \leq \frac{G}{L} \leq \frac{1}{N}$, where G is the number of groups. It is obvious that $Pr[ASA(v_i)|g_i]$ is between $\frac{1}{2N-1}$ and $\frac{1}{N}$ and $|Pr[ASA(v_i)] - Pr[ASA(v_i)|g_i]| \leq \frac{1}{N}$ can hold, which satisfies the δ requirement.

After β and δ requirements are satisfied, the records in each group have been determined. If α and γ requirements cannot be satisfied in one group, **Step 4** will be performed. In particular, if the data owner is not sure of γ, he can only set $\gamma \geq \alpha^k \times \beta$ to satisfy γ requirement once α requirement and β requirement are satisfied, which is inferred from Equations (3), (5), (6) and (7). The impact of k on the performance of our SLPPA will be evaluated in our simulations.

It is obvious that different kinds of QIs will bring different convergence capabilities. Assume that there are X different QIs and min and max represent the minimal and maximal distinct values among QIs. When α is smaller than $\frac{1}{max^{X-k+1}}$, α requirement can never be met. When α is larger than $\frac{1}{min}$, α requirement can always be met. When α is set to be between $\frac{1}{max^{X-k+1}}$ and $\frac{1}{min}$, whether α requirement can be met is decided by the correlation among QIs.

5 Utility and Privacy Analysis

The aim of our design is to protect the users' privacy while not compromising the utility of data. In this section, we first analyze the utility of the published tables based on our SLPPA, and then prove that each group can satisfy the $(\alpha, \beta, \gamma, \delta)$ privacy model.

5.1 Utility

In many data studies, there is a need to obtain the data statistics. We define the utility based on how well one can estimate count queries, i.e. the number of records that meet a query condition needs to be found. To evaluate the anonymous data utility of our method, we first explain how to respond to a counting query.

We denote Q as a query and $Q(C)$ as the query result that satisfies C, where C is a set of constraints for QIs and SA. We let Q_i represent the constraint for all the QIs in i-th separate table (e.g. $age > 30$) and Q_{SA} denote the constraint for SA (e.g. $Salary = 6700$), the result $Q(C)$ in each group can be computed as follows:

$$Q(C) = Q(Q_1) \times \cdots \times Q(Q_k) \times Q(Q_{SA}), \tag{11}$$

where the query result $Q(Q_i)$ that satisfies the constraint Q_i is equal to n/m, n is the number of records satisfying Q_i, m is the number of items in the group and k is the number of separate tables. $Q(Q_{SA})$ is equal to the number of records satisfying Q_{SA}. Repeatedly, we get the query result for each separate group with Equation (11) and the sum of all the results is the final query result that satisfies C. Obviously, the utility decreases with the number of items m and the number of separate tables k, because each multiplier is less than one.

Next we take the following example to explain how to estimate this number from the tables published.

SELECT count(*) FROM Published-data
WHERE $Age > 30$ AND $Job = Doctor$ AND
$Salary = 6700$.

This query contains several constraints. As a response, we can first find the $GIDs$ that contain $Salary = 6700$, and we find g_2 from the Table 3(c). From the group g_2, we can infer that the number of records that satisfy the query requirement is $1 \times \frac{3}{4} \times \frac{3}{4} = \frac{9}{16}$, as Table 3(c) contains one record with $Salary = 6700$, 3 records out of 4 in Table 3(a) satisfy $Age > 30$ and 3 records out of 4 in Table 3(b) satisfy $Job = Doctor$. This example shows that the probability of inferring the right record is $\frac{9}{16}$ after the group division, or we can consider the estimated number of records returned from the query is $\frac{9}{16}$. If we do not divide all records into groups, the probability of inferring the right record is $1 \times \frac{6}{8} \times \frac{3}{8} = \frac{9}{32}$ from Table 4.

From Table 1, we know that only one record satisfies this query. Therefore, grouping helps to find an answer closer to the actual value. In addition, a query can be sped up with the use of group information. As we ensure that the privacy requirements can be met when forming the groups and dividing the tables, the privacy will still be satisfied while improving the utility of the data. Next, we will prove that our SLPPA can satisfy the four privacy requirements.

5.2 Privacy

In this section, we take Table 3 as an example to prove that our SLPPA can satisfy the four privacy requirements.

Privacy for α-record-association: α privacy requirement is defined to resist the record linkage attack. In this situation, the attacker aims to infer the right QIs of the target victim (e.g. Alice) with a part of QIs or SA being the background knowledge.

Suppose that there are k separate tables to contain all QIs, in the worst case, the attacker has known Alice' s $k-1$ separate tables of QIs and SA values. The probability of inferring Alice' s record, i.e. the last table of Alice, is equal to the probability of choosing a red ball from a bucket containing n_j balls, $\frac{d_r}{n_j}$, where d_r represents the number of red balls in the bucket and is less than the largest number of balls with the same color c_j^i. Then $\frac{{d_r}^i}{n_j} \leq \frac{c_j^i}{n_j}$ can hold. As we have proven $Max(\frac{c_j^1}{n_j}, \ldots, \frac{c_j^k}{n_j}) \leq \alpha$ in Section 4.2, we can get $\frac{d_r}{n_j} \leq \frac{c_j^i}{n_j} \leq Max(\frac{c_j^1}{n_j}, \ldots, \frac{c_j^k}{n_j}) \leq \alpha$, which means that the biggest probability of inferring Alice' s record is smaller than α.

We give an example to illustrate the above proof process. We ensure Equation (3), $\alpha \geq Max(\frac{c_j^1}{n_j}, \ldots, \frac{c_j^k}{n_j})$, to be met. In Table 3, $\alpha \geq \frac{1}{4}$ must be set. As the background knowledge, if the attacker knows Alice's QIs ($Age = 30$ and $Zipcode = 11100$), he can infer that Alice may be in g_1 or g_2. The probability of inferring the right QIs of Alice in g_1 or g_2 is both $\frac{1 \times 1}{4 \times 4}$. As another type of background knowledge, if the attacker only knows Alice' s SA ($Salary = 4500$), Alice must be in g_1, and the probability of inferring the right QIs is $\frac{1 \times 1}{4 \times 4}$.

In a word, our SLPPA can meet the $\alpha - record - association$ requirement for privacy no matter the attacker knows SA or a part of QIs of the victim.

Privacy for β-sensitive-association: β privacy requirement is defined to resist the attribute linkage attack. In this case, we assume that an attacker may know a part of QIs or all of QIs of the victim. The attacker aims to infer the right SA of the target victim.

In the worst case, the attacker has known all the QIs of the victim Alice. The probability of inferring Alice' s SA is equal to the probability of choosing a red ball from a bucket that contains n_j balls, $\frac{d_r}{n_j}$, where d_r is the number of red balls in the bucket and less than the largest number of balls with the same color c_j. Then, we can get $\frac{d_r}{n_j} \leq \frac{c_j}{n_j}$. As we have proven $\frac{c_j}{n_j} \leq \beta$ in Section 4.2, we can get $\frac{d_r}{n_j} \leq \frac{c_j}{n_j} \leq \beta$, which indicates that the biggest probability of inferring Alice' s SA is smaller than β.

We again give an example to illustrate the above proof process. As our SLPPA ensures $\beta \geq \frac{c_j}{n_j}$ in Equation (5) to hold, we can get $\beta \geq \frac{2}{4}$ in Table 3. No matter which group the victim belongs to, the probability of inferring his/her right SA must be smaller than $\frac{2}{4}$. So even though the attacker can know all of QIs of Alice, $\{F, Teacher, 30, 11100\}$, he can infer Alice's SA in g_1 at a probability of less than $\frac{2}{4}$. If the attacker only knows a part of QIs ($Age = 30$ and $Zipcode = 11100$), he can deduce Alice's SA in g_1 or g_2 at a probability of $\frac{2}{4}$ or 0 respectively.

In summary, our SLPPA can achieve the privacy requirement on $\beta - sensitive - association$.

Privacy for γ-presence: γ privacy requirement is defined to resist the table linkage attack. We assume that an attacker may know a part of QIs, all of QIs or SA of the target victim. The attacker attempts to infer whether the victim appears in the published tables.

Suppose that k separate tables contain all QIs and one table records SA values. The worst case is that the attacker has known all of QIs and SA of an individual (e.g. Bob). With the existence of $k+1$ separate tables, there are ${n_j}^{k+1}$ combinations for a certain group g_j. The probability of inferring Bob in g_j is equal to the probability of choosing $k+1$ red balls respectively from $k+1$ different buckets. When each bucker contains n_j balls, $\frac{d_j^1 \times d_j^2 \cdots \times d_j^k \times d_j}{(n_j)^{k+1}}$ will hold, where ${d_j}^i$ represents the number of red balls in the i-th bucket and is less than the largest number of balls with the same color ${c_j}^i$. We have $\frac{d_j^1 \times d_j^2 \cdots \times d_j^k \times d_j}{(n_j)^{k+1}} \leq \frac{c_j^1 \times c_j^2 \cdots \times c_j^k \times c_j}{(n_j)^{k+1}}$. As we have proven $\frac{c_j^1 \times c_j^2 \cdots \times c_j^k \times c_j}{(n_j)^{k+1}} \leq \gamma$ in Section 4.2, $\frac{d_j^1 \times d_j^2 \cdots \times d_j^k \times d_j}{(n_j)^{k+1}} \leq \frac{c_j^1 \times c_j^2 \cdots \times c_j^k \times c_j}{(n_j)^{k+1}} \leq \gamma$ can be obtained, which means the biggest probability of inferring Bob in g_j is smaller than β.

Take Alice for an example, if an attacker knows all the QIs of Alice, he can infer that only g_1 may include Alice, and the probability that g_1 has a record of Alice is no more than $\frac{1}{4} \times \frac{1}{4} \times \frac{2}{4} = \frac{1}{32}$. As another type of background knowledge, if the attacker knows a part of QIs ($Age = 30$ and $Zipcode = 11100$) on Alice, he can infer that Alice may belong to g_1 or g_2. The probability of g_1 has a record of Alice is no more than $\frac{1}{4} \times \frac{1}{4} \times \frac{2}{4} = \frac{1}{32}$, and the probability of g_2 has a record of Alice is no more than $\frac{1}{4} \times \frac{1}{4} \times \frac{1}{4} = \frac{1}{64}$. Consequently, the detection of the presence of Alice is $\frac{1}{2} \times (\frac{1}{32} + \frac{1}{64}) = \frac{3}{128}$. For the third background knowledge, if the attacker only knows Alice's SA ($Salary = 4500$), he can infer that g_1 has a record of Alice at the probability of $\frac{1}{4} \times \frac{1}{4} \times \frac{2}{4} = \frac{1}{32}$.

Consequently, our SLPPA meets the $\gamma - presence$ privacy requirement.

Privacy for δ-probability: δ privacy requirement is defined to resist the probabilistic attack. In this case, we assume that an attacker may know the victim's SA or the probability that the victim possesses the SA. The attacker tries to infer some useful information by comparing the difference in the probability of knowing the SA value before and after the anonymization.

In our SLPPA, we divide individuals into groups based on Equation (8). We define $Pr[ASA(v_i)]$ as the probability of $ASA(v_i)$ in the original table, $Pr[ASA(v_i) \mid g_i]$ as the probability of $ASA(v_i)$ in g_i, and $Pr[ASA(v_i) \mid g_j]$ as the probability of $ASA(v_i)$ in g_j. Based on Equation (8), we can get $\mid Pr[ASA(v_i)] - Pr[ASA(v_i) \mid g_i] \mid \leq \delta$ and $\mid Pr[ASA(v_i)] - Pr[ASA(v_i) \mid g_j] \mid \leq \delta$. We can also get

$$\mid Pr[ASA(v_i) \mid g_i - Pr[ASA(v_i) \mid g_j] \mid \leq 2\delta, \qquad (12)$$

which means the probability difference between groups in the anonymized table is smaller than 2δ. As a result, our SLPPA can meet the $\delta - probability$ privacy requirement.

We exploit the use of grouping to ensure that the number of items in a group meet various requirements in order to quantify the privacy guarantee level. The number of groups to divide is a tradeoff between increasing the data usage efficiency and conserving data privacy. If there is a need to find statistics of data, the grouping would reduce the overhead in collecting the results as shown in our example in Section 5.1. However, if an attacker learns to use the grouping to infer user data, it will also reduce the space for the trying. The use of larger groups will help better conserve privacy but also take longer time to collect information from the published tables.

If the attacker does not infer the group which the victim belongs to, the query space will be very large, which makes the probability of inferring other information on the victim be much smaller than α,β,γ,and δ. In summary, our SLPPA can prevent the attackers from discovering new information about the target victim with the probability higher than the specified thresholds of α,β,γ,and δ by taking various background knowledge into consideration.

6 Performance Evaluations

Setup: We implement our SLPPA in Python. We use a workstation running Linux version 3.19.0 with core i5, Duo 2.7GHz CPU and 4GB RAM. We use SQLite to store the data set.

Dataset: To evaluate the performance of our SLPPA, we conduct our studies on two data sets, **Adult** and **Census-Income**, which can be obtained from UCI Machine Learning Repository [53]. They are two of the most popular public data sets since 2007. **Adult** is extracted from the U.S. Census database with 48842 instances. **Census-Income** is extracted from the population surveys conducted by the U.S. Census Bureau with 199523 instances. To compare SLPPA with ASN and Slicing, we adopt the same QIs in these two schemes, as shown in Table 5 and Table 6. These QIs can uniquely identify the individuals in each data set. We randomly select $(5K, 10K, 15K, 20K, 25K)$ tuples from **Adult**, and $(50K, 100K, 150K, 190K)$ tuples from **Census-Income**.

TABLE 5
Attributes of Adult

Attribute	Number of Distinct Values
Age	72
Workclass	7
Education	16
Marital	7
Occupation	14
Relationship	6
Race	5
Sex	2
Hours-Per-Week	94
Country (SA)	41

TABLE 6
Attributes of Census-Income

Attribute	Number of Distinct Values
Age	91
Worker	9
Industry	52
Education	17
WageHour	1240
Marital	7
Major	24
Mace	5
Sex	2
Employment	8
Gains (SA)	132

ASN also takes two procedures, table division and group division. During the table division, each attribute constitutes a separate table. During the group division, all the individuals in the original table are partitioned into non-overlapping groups so that the published data satisfy the pre-defined privacy requirement to resist the attribute linkage and table linkage attacks. In **Slicing**, k-medoid method is first used to cluster attributes into k separate tables, and then Mondrian algorithm is used for grouping tuples into different groups to meet the ℓ-diversity requirement. In our **SLPPA**, to partition attributes into separate tables, we adopt entropy and mean-square contingency coefficient to guide the process. The increase of correlation between attributes within a table and reduction of correlation across tables help to better preserve the utility and privacy. During the group division, all the individuals in the original table are partitioned into non-overlapping groups so that the published data satisfies the pre-defined $(\alpha, \beta, \gamma, \delta)$ model to meet more strict privacy requirement. **Slicing** and **ASN** only take attribute linkage and table linkage into consideration, while SLPPA aims to prevent record linkage, attribute linkage, sensitive linkage and probability attack.

The following metrics are used to evaluate the three schemes:

• *Time Consumption(TC)*: Time consumption is used to measure the time taken to perform table division and group division.

• *Relative Error of Count Query(RECQ)*: Count query is usually used to return the number of rows which match a specified criteria. For a specific count query Q, *RECQ* is defined to evaluate the information loss before and after anonymization in Equation 13,

$$RECQ = \frac{\mid Q(T) - Q(T^{*}) \mid}{Q(T)}, \qquad (13)$$

where $Q(T)$ and $Q(T^{*})$ represent the query results from the original table and the published tables respectively.

6.1 Performance Comparison of Table Division and Group Division

In this set of simulations, the corresponding parameters are set in Table 7.

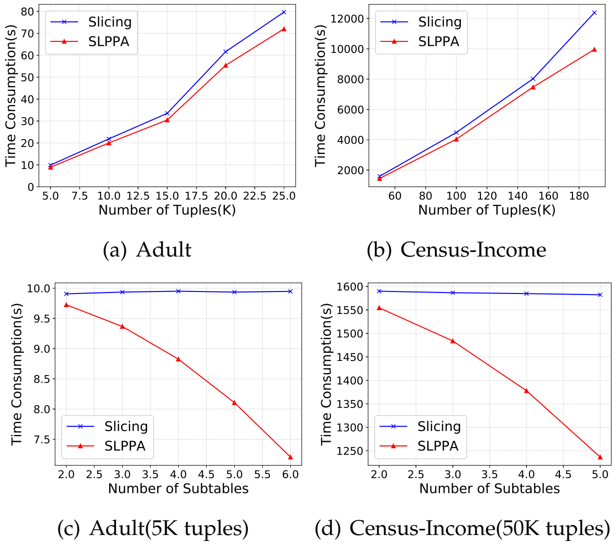
TABLE 7
Parameters of Division

Algorithm	Parameter
ASN	$\alpha = 0.005, \beta = 0.05$
Slicing	$\ell = 20, k = 3$
SLPPA	$\alpha = 0.5, \beta = 0.05, \gamma = 0.005, \delta = 0.05, k = 3$

Time consumption of table division: As **ASN** simply puts each attribute into a separate table, there is no algorithm used for table division. We only compare the time consumed by **SLPPA** and **Slicing**. In Fig.2(a) and Fig.2(b), the time consumption of both schemes increases with the size of the data set when we divide all QIs into 3 tables. However, **SLPPA** takes slightly less time. Although both schemes adopt the mean-square contingency coefficient to compute the association between QIs, the calculation of QI weight in SLPPA can reduce the time slightly taken for finding the coefficient. Fig.2(c) and Fig.2(d) show that the time consumption of our **SLPPA** decreases with the number of subtables k, while **Slicing** maintains a stable consumption. Our scheme first determines the first column of each subtable based on the attribute weight and computes the correlation between other attributes and the first column to achieve the table division, while **Slicing** first computes the correlation between all attributes and then clusters attributes based on their correlation. Fig.2 shows that **SLPPA** can resist the four attacks without taking more extra time, while **ASN** only can resist presence leakage attack and sensitive association leakage attack.

Attribute distribution of table division: Fig.3 shows the attribute distribution after the table division. In this study, we divide QIs into 3 tables in Fig.3(a), Fig.3(b), Fig.3(c) and Fig.3(d). In Fig.3(a) and Fig.3(b), we adopt the attributes listed in Table 5 and Table 6. In Fig.3(c) and Fig.3(d), we add 4 extra QIs {*Final-Weight, Education-Num, Capital-Gain, Capital-Loss*} into **Adult** and 3 extra QIs {*Capital-Gain, Capital-Loss, Tax-Filter-Status*} into **Census-Income**. These four figures show that **SLPPA** has a more even attribute distribution. For example, three tables of **SLPPA** contain 2, 3 and 4 attributes respectively in Fig.3(a). The clustering algorithm in **Slicing** may produce some isolated nodes. Particularly, 3 tables of **Slicing** in Fig.3(c) contain only one attribute. To satisfy the privacy requirements in Table 7, we also partition all attributes into four tables and **SLPPA**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2919833, IEEE Transactions on Dependable and Secure Computing

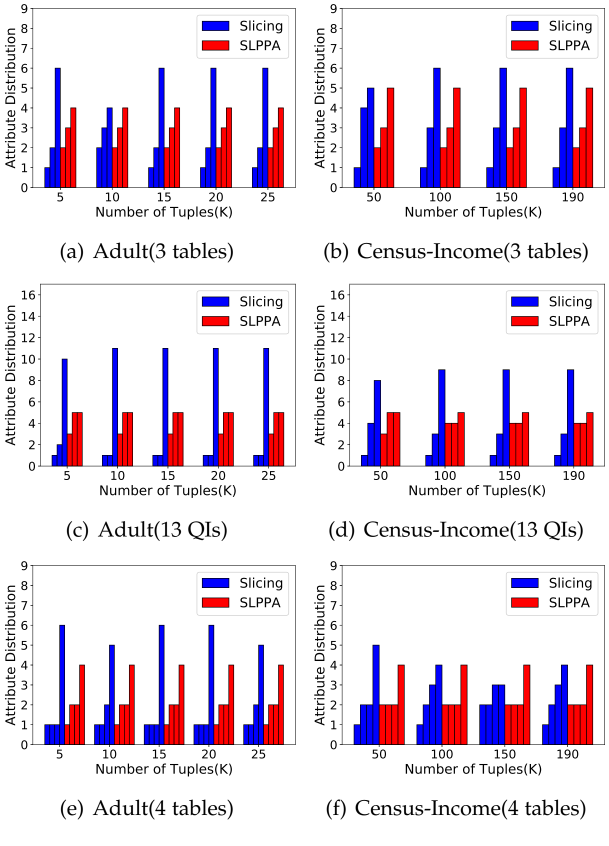


(a) Adult (b) Census-Income

(c) Adult(5K tuples) (d) Census-Income(50K tuples)

Fig. 2. Time Consumption of Table Division

still has a more even attribute distribution in Fig.3(e) and Fig.3(f).



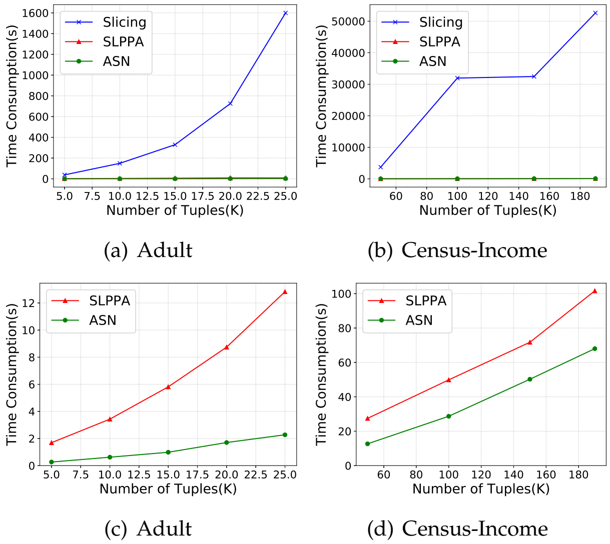
(a) Adult(3 tables) (b) Census-Income(3 tables)

(c) Adult(13 QIs) (d) Census-Income(13 QIs)

(e) Adult(4 tables) (f) Census-Income(4 tables)

Fig. 3. Attribute Distribution of Table Division

Time consumption of group division: Fig.4 shows the time consumption of group division in **SLPPA**, **Slicing** and **ASN**. We can see that **Slicing** possesses the highest amount of time consumption because it takes a recursive Mondrian algorithm to divide groups. As discussed before, **ASN** consumes the least amount of time because it only considers attribute linkage and table linkage during the group division. In order to make a better distinction between **ASN** and **SLPPA**, Fig.4(a) and Fig.4(b) are enlarged into Fig.4(c) and Fig.4(d).

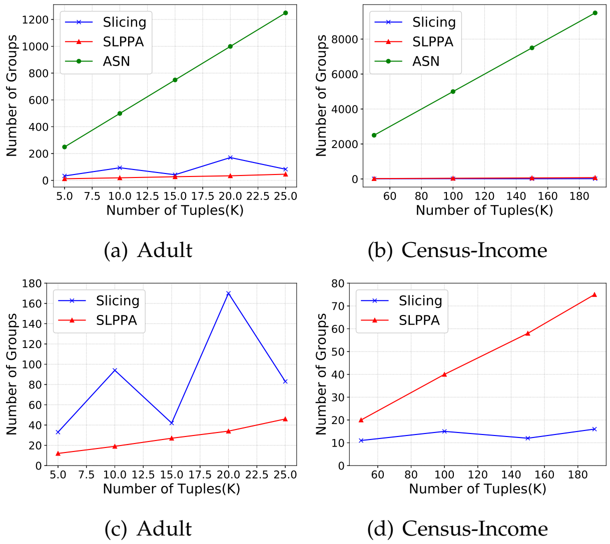


(a) Adult (b) Census-Income

(c) Adult (d) Census-Income

Fig. 4. Time Consumption of Group Division

Number of groups after group division: In Fig.5(a) and Fig.5(b), **ASN** has the biggest number of groups after group division, implying that each group contains fewer items. Too many groups in **ASN** may increase the difficulty and reduce the accuracy of count query. To make the distinction between Slicing and SLPPA clearer, Fig.5(c) and Fig.5(d) are enlarged from Fig.5(a) and Fig.5(b). Fig.5(c) and Fig.5(d) show that **SLPPA** generates more groups under a big data set and the number increases with the number of tuples regularly. **Slicing** has a larger fluctuation in Fig.5(c) and Fig.5(d).



(a) Adult (b) Census-Income

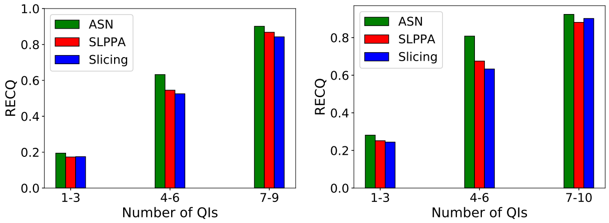
(c) Adult (d) Census-Income

Fig. 5. Number of Groups After Group Division

6.2 Comparison of Information Loss

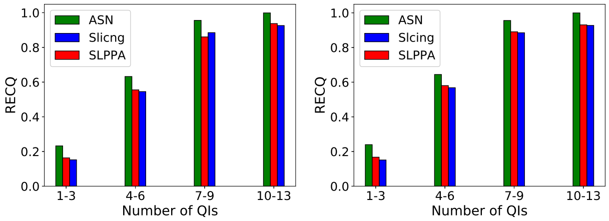
In our studies, we randomly select 1000 count queries for **Adult** and **Census-Income** to evaluate the information loss.

We adopt the same parameters in Table 7 to get Fig.6 with three cases, the count queries including random 1 to 3 QIs, 4 to 6 ones, and 7 to 9(or 10) ones. **ASN** has the biggest RECQ because **ASN** generates the most number of separate tables by putting each attribute into a separate table and the most number of groups. **Slicing** has the smallest RECQ because the large variation in attribute distribution of Slicing in Fig.3 may make a separated table contain too many QIs, bringing some convenience to the queries. However, there is only a small difference of RECQ between **SLPPA** and **Slicing**. In addition, RECQ increases with the number of QIs in Fig.6. Even when the number of QIs increases, Fig.7 still has the result similar to that in Fig.6.



(a) Adult (b) Census-Income

Fig. 6. Comparison of Relative Error



(a) Adult (b) Census-Income

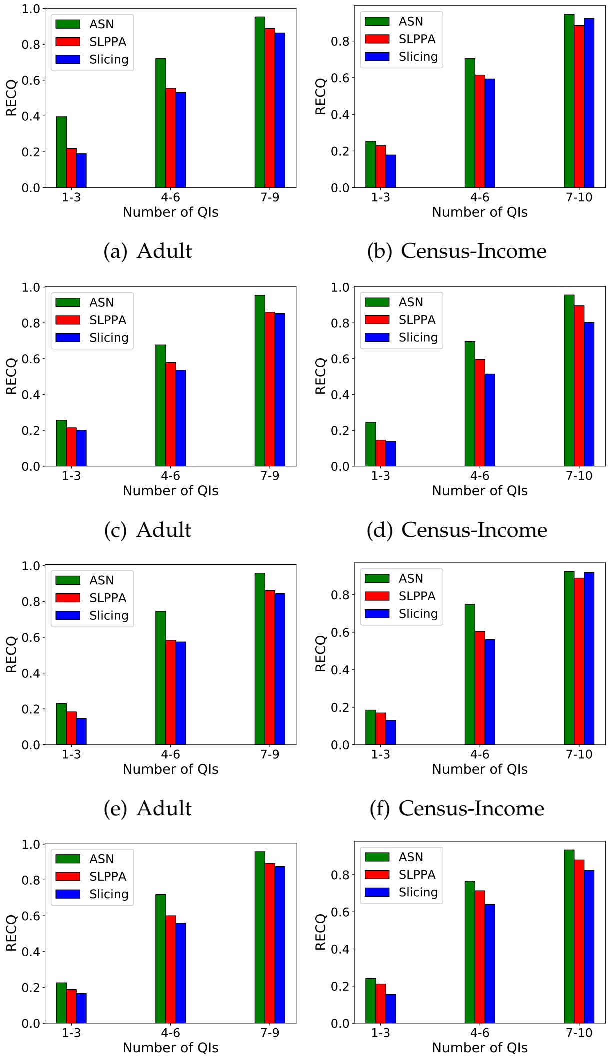
Fig. 7. Comparison of Relative Error With more QIDs

TABLE 8
Parameters of Information Loss

Figure 8	Parameters Difference
(a)(b)	$\alpha = 0.05$ in SLPPA
(c)(d)	$\beta = 0.1$ in ASN and SLPPA $\ell = 10$ in Slicing
(e)(f)	$\gamma = 0.01$ in SLPPA $\alpha = 0.01$ in ASN
(g)(h)	$\delta = 0.02$ in SLPPA

We evaluate the effect of the parameters in **SLPPA**, **Slicing** and **ASN** on the information loss. Unless specified, the default parameters are set in Table 8. Fig.8 shows the information loss has no obvious difference compared with Fig.6. In Fig.9, the data utility decreases with the number of separate tables. Fig.10 shows that the data utility decreases with the group size, which has been discussed in Equation 11.

Summary: From our results, **ASN** takes the least amount of time but has the highest information loss. Compared with **Slicing**, our **SLPPA** takes less time, especially in group division. **SLPPA** also has a more even number of attributes



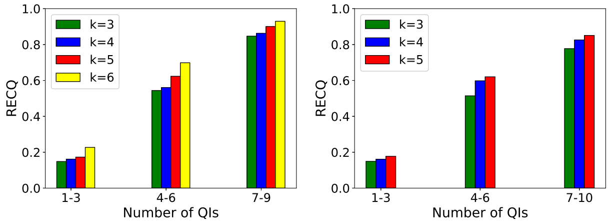
(a) Adult (b) Census-Income

(c) Adult (d) Census-Income

(e) Adult (f) Census-Income

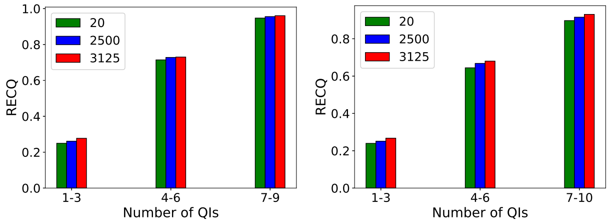
(g) Adult (h) Census-Income

Fig. 8. Relative Error of Different Parameters



(a) Adult (b) Census-Income

Fig. 9. Relative Error of Different Subtables



(a) Adult (b) Census-Income

Fig. 10. Relative Error of Different Group size

in different tables and groups. Although **Slicing** has a better utility than **SLPPA**, the difference is very small. In addition, our **SLPPA** can resist the most attack types among the three algorithms.

7 Conclusion

We design and implement an anonymity technique named SLPPA to protect the sensitive attribute during the publication of social data. To resist attacks resulted from record linkage, table linkage and attribute linkage as well as probabilistic attacks, we propose a $(\alpha, \beta, \gamma, \delta)$ privacy model. To reduce the time consumption and improve the performance of our SLPPA, we design two algorithms respectively for efficient table division and the group division. Besides better preserving the data privacy, our performance studies based on two comprehensive sets of real-world data demonstrate that SLPPA can also provide good data usability. Our privacy analysis show that SLPPA can also resist the background knowledge attack.

In the future work, we will consider how to protect the privacy of published data with multiple sensitive attributes and extend our algorithms to protect the privacy of graph data in social networks.

Acknowledgments

This research is sponsored in part by the National Natural Science Foundation of China (contract/grant numbers: 61872053), and US National Science Foundation (CNS 1526843).

References

[1] Y. Wang, L. Xie, B. Zheng, and K. C. Lee, "High utility k-anonymization for social network publishing," *Knowledge and Information Systems*, vol. 41, no. 3, pp. 697–725, 2014.

[2] Y. Xu, T. Ma, M. Tang, and W. Tian, "A survey of privacy preserving data publishing using generalization and suppression," *Applied Mathematics and Information Sciences*, vol. 8, no. 3, p. 1103, 2014.

[3] N. Victor, D. Lopez, and J. H. Abawajy, "Privacy models for big data: a survey," *International Journal of Big Data Intelligence*, vol. 3, no. 1, pp. 61–75, 2016.

[4] B. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Computing Surveys (CSUR)*, vol. 42, no. 4, p. 14, 2010.

[5] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[6] M. Rajaei, M. S. Haghjoo, and E. K. Miyaneh, "Ambiguity in social network data for presence, sensitive-attribute, degree and relationship privacy protection," *PloS one*, vol. 10, no. 6, p. e0130693, 2015.

[7] X. Xiao and Y. Tao, "Anatomy: Simple and effective privacy preservation," in *Proceedings of the 32nd international conference on Very large data bases*. VLDB Endowment, 2006, pp. 139–150.

[8] T. Li, N. Li, J. Zhang, and I. Molloy, "Slicing: A new approach for privacy preserving data publishing," *IEEE transactions on knowledge and data engineering*, vol. 24, no. 3, pp. 561–574, 2012.

[9] K. Tadisetti, P. Madhuri, and J. B. Shastri, "Implementation of slicing technique for privacy preserving data publishing," *Journal of Neurophysiology*, vol. 114, no. 3, pp. 1538–44, 2015.

[10] M. Wang, Z. Jiang, Y. Zhang, and H. Yang, "T-closeness slicing: A new privacy-preserving approach for transactional data publishing," *Informs Journal on Computing*, vol. 30, no. 3, pp. 438–453, 2018.

[11] V. S. Susan and T. Christopher, "Anatomisation with slicing: a new privacy preservation approach for multiple sensitive attributes," *SpringerPlus*, vol. 5, no. 1, pp. 1–21, 2016.

[12] Z. H. Zou, Y. Yi, and J. N. Sun, "Entropy method for determination of weight of evaluating indicators in fuzzy synthetic evaluation for water quality assessment," *Journal of Environmental Sciences*, vol. 18, no. 5, pp. 1020–1023, 2006.

[13] A. Gkoulalas-Divanis, G. Loukides, and J. Sun, "Publishing data from electronic health records while preserving privacy: A survey of algorithms," *Journal of Biomedical Informatics*, vol. 50, no. 8, pp. 4–19, 2014.

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

[15] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 571–588, 2002.

[16] S. Kiyomoto and T. Tanaka, "A user-oriented anonymization mechanism for public data," in *International Conference on International Workshop on Data Privacy Management*. ACM, 2010, pp. 22–35.

[17] V. S. Iyengar, "Transforming data to satisfy privacy constraints," *Kdd*, pp. 279–288, 2002.

[18] X. Zhang, C. Liu, S. Nepal, and J. Chen, "An efficient quasi-identifier index based approach for privacy preservation over incremental data sets on cloud," *Journal of Computer and System Sciences*, vol. 79, no. 5, pp. 542–555, 2013.

[19] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Incognito: Efficient full-domain k-anonymity," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. ACM, 2005, pp. 49–60.

[20] R. Chen, B. C. M. Fung, N. Mohammed, B. C. Desai, and K. Wang, "Privacy-preserving trajectory data publishing by local suppression," *Information Sciences An International Journal*, vol. 231, no. 1, pp. 83–97, 2013.

[21] M. Serpell, J. Smith, A. Clark, and A. Staggemeier, "A preprocessing optimization applied to the cell suppression problem in statistical disclosure control," *Information Sciences*, vol. 238, no. 7, pp. 22–32, 2013.

[22] B. C. M. Fung, K. Wang, and P. S. Yu, "Anonymizing classification data for privacy preservation," *IEEE Transactions on Knowledge & Data Engineering*, vol. 19, no. 5, pp. 711–725, 2015.

[23] B. C. Fung, K. Wang, and P. S. Yu, "Top-down specialization for information and privacy preservation," in *21st International Conference on Data Engineering (ICDE'05)*. IEEE, 2005, pp. 205–216.

[24] X. Sun, L. Sun, and H. Wang, "Extended k -anonymity models against sensitive attribute disclosure," *Computer Communications*, vol. 34, no. 4, pp. 526–535, 2011.

[25] M. Ye, X. Wu, X. Hu, and D. Hu, "Anonymizing classification data using rough set theory," *Knowledge-Based Systems*, vol. 43, no. 2, pp. 82–94, 2013.

[26] R. Mahesh and T. Meyyappan, "Anonymization technique through record elimination to preserve privacy of published data," in *International Conference on Pattern Recognition, Informatics and Mobile Engineering*, 2013, pp. 328–332.

[27] Q. Liu, H. Shen, and Y. Sang, "Privacy-preserving data publishing for multiple numerical sensitive attributes," *Tsinghua Science and Technology*, vol. 20, no. 3, pp. 246–254, 2015.

[28] A. Thakkar, A. A. Bhatti, and J. Vasa, *Correlation Based Anonymization Using Generalization and Suppression for Disclosure Problems*. Springer International Publishing, 2015.

[29] E. G. Komishani, M. Abadi, and F. Deldar, "Pptd: Preserving personalized privacy in trajectory data publishing by sensitive attribute generalization and trajectory local suppression," *Knowledge-Based Systems*, vol. 94, pp. 43–59, 2016.

[30] Y. Tao, H. Chen, X. Xiao, S. Zhou, and D. Zhang, "Angel: Enhancing the utility of generalization for privacy preserving publication," *Knowledge and Data Engineering IEEE Transactions on*, vol. 21, no. 7, pp. 1073–1087, 2009.

[31] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu, "Aggregate query answering on anonymized tables," in *2007 IEEE 23rd International Conference on Data Engineering*. IEEE, 2007, pp. 116–125.

[32] M. Rajaei, M. S. Haghjoo, and E. K. Miyaneh, "An anonymization algorithm for (α, β, γ, δ)-social network privacy considering data utility." *J. UCS*, vol. 21, no. 2, pp. 268–305, 2015.

[33] Q. Liu, G. Wang, F. Li, S. Yang, and J. Wu, "Preserving privacy with probabilistic indistinguishability in weighted social networks,"

IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 5, pp. 1417–1429, 2017.

[34] L. Dong, X. He, L. B. Cao, and H. Chen, "Permutation anonymization," *Journal of Intelligent Information Systems*, vol. 47, no. 3, pp. 427–445, 2016.

[35] M. Bahrami and M. Singhal, "A light-weight permutation based method for data privacy in mobile cloud computing," in *IEEE International Conference on Mobile Cloud Computing*. IEEE, 2015, pp. 190–196.

[36] M. Bahrami, L. Dong, M. Singhal, and A. Kundu, "An efficient parallel implementation of a light-weight data privacy method for mobile cloud users," in *Seventh International Workshop on Data-intensive Computing in the Clouds*. IEEE, 2017, pp. 51–58.

[37] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 2001, pp. 247–255.

[38] W. Du and Z. Zhan, "Using randomized response techniques for privacy-preserving data mining," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2003, pp. 505–510.

[39] Z. Huang, W. Du, and B. Chen, "Deriving private information from randomized data," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. ACM, 2005, pp. 37–48.

[40] I. Cano and V. Torra, "Edit constraints on microaggregation and additive noise," in *International Workshop on Privacy and Security Issues in Data Mining and Machine Learning*. Springer, 2010, pp. 1–14.

[41] L. Peng, L. E. Wang, and X. Li, "Randomized perturbation for privacy-preserving social network data publishing," in *IEEE International Conference on Big Knowledge*. IEEE, 2017, pp. 208–213.

[42] C. Dwork, "Differential privacy," in *International Colloquium on Automata, Languages, and Programming*, 2006, pp. 1–12.

[43] F. Ahmed, A. X. Liu, and J. Rong, "Social graph publishing with privacy guarantees," in *IEEE International Conference on Distributed Computing Systems*. IEEE, 2016, pp. 447–456.

[44] A. N. K. Zaman, C. Obimbo, and R. A. Dara, "An improved data sanitization algorithm for privacy preserving medical data publishing," in *Canadian Conference on Artificial Intelligence*, 2017, pp. 64–70.

[45] J. Domingo-Ferrer and V. Torra, "A quantitative comparison of disclosure control methods for microdata," *Confidentiality, disclosure and data access: theory and practical applications for statistical agencies*, pp. 111–134, 2001.

[46] M. Xue, P. Karras, R. Chedy, P. Kalnis, and H. K. Pung, "Delineating social network data anonymization via random edge perturbation," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 475–484.

[47] N. Mattani, J. S. Kumar, A. Prabakaran, and N. Maheswari, "Privacy preservation in social network analysis using edge weight perturbation," *Indian Journal of Science and Technology*, vol. 9, no. 37, 2016.

[48] C. C. Aggarwal and S. Y. Philip, "A framework for condensation - based anonymization of string data," *Data Mining and Knowledge Discovery*, vol. 16, no. 3, pp. 251–275, 2008.

[49] C. C. Aggarwal and P. S. Yu, "On static and dynamic methods for condensation-based privacy-preserving data mining," *ACM Transactions on Database Systems (TODS)*, vol. 33, no. 1, p. 2, 2008.

[50] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, p. 3, 2007.

[51] Y. M. Cheung and H. Jia, "Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number," *Pattern Recognition*, vol. 46, no. 8, pp. 2228–2238, 2013.

[52] H. Cramér, *Mathematical Methods of Statistics (PMS-9)*. Princeton university press, 2016, vol. 9.

[53] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

Lin Yao is an associate professor in School of Software, Dalian University of Technology (DUT), China. Her research interests include security and privacy of VANET, CCN, and social network.

Zhenyu Chen is an M.E. candidate in School of Software, Dalian University of Technology (DUT), China. His research interests include security and privacy in social network.

Xin Wang is currently an associate professor of the department of Electrical and Computer Engineering, Stony Brook University, New York State, U. S. A. Her research interests include mobile and ubiquitous computing, wireless communications and network systems, networked sensing and fusion, detection and estimation.

Dong Liu is an M.E. candidate in School of Software, Dalian University of Technology (DUT), China. His research interests include security and privacy in social network.

Guowei Wu received his Ph.D. degree from Harbin Engineering University, PR China. He is now a professor at the School of Software, Dalian University of Technology (DUT), China. His research interests include embedded real-time system, cyber-physical systems (CPS), and wireless sensor networks.