Computer Science and Information Systems 12(4):1193–1216 DOI: 10.2298/CSIS141212052Z

Anonymization on Refining Partition: Same Privacy, More Utility

Hong Zhu[1], Shengli Tian[2], Genyuan Du[2], and Meiyi Xie[1]

[1] School of Computer Science and Technology, Huazhong University of Science and Technology
Wuhan,Hubei, 430074, P.R.China
zhuhong@hust.edu.cn

[2] School of Information Engineering, Xuchang University
Xuchang,Henan, 461000, P.R.China
zelintian@gmail.com

Abstract. In privacy preserving data publishing, to reduce the correlation loss between sensitive attribute (*SA*) and non-sensitive attributes(*NSAs*) caused by anonymization methods (such as generalization, anatomy, slicing and randomization, etc.), the records with same *NSAs* values should be divided into same blocks to meet the anonymizing demands of ℓ-diversity. However, there are often many blocks (of the initial partition), in which there are more than ℓ records with different *SA* values, and the frequencies of different *SA* values are uneven. Therefore, anonymization on the initial partition causes more correlation loss. To reduce the correlation loss as far as possible, in this paper, an optimizing model is first proposed. Then according to the optimizing model, the refining partition of the initial partition is generated, and anonymization is applied on the refining partition. Although anonymization on refining partition can be used on top of any existing partitioning method to reduce the correlation loss, we demonstrate that a new partitioning method tailored for refining partition could further improve data utility. An experimental evaluation shows that our approach could efficiently reduce correlation loss.

Keywords: anonymization, refining partition, correlation loss.

1. Introduction

In this information age, it is easy for many organizations and agencies to collect digital data (containing unaggregated information about individuals), and knowledge extracted by data mining techniques represents a key asset driving innovation, policy-making activities. Driven by the regulations that require certain data to be published, or by mutual benefits, there is a demand for the publication of data among various parties [1] [14] [15]. However, detailed person-specific information often contains sensitive values about individuals (such as salary, credit, etc.), and publishing such data would lead to the disclosure of the sensitive values of individuals [1] [14] [15]. Thus, it is important to transform data table into anonymized data table so that the sensitive values of individuals could not be inferred with high certainty, and the statistical information for large number of individuals, such as the correlations between sensitive attribute (*SA*) and non-sensitive attributes (*NSAs*), should be preserved as much as possible.

The anonymization methods, such as generalization [14] [15] [3] [7] [11] [19] anatomy (also called bucketization) [20] [16] [5] slicing [13] [17] and randomization [12] [18] [2], are usually used for privacy preserving data publishing (*PPDP*). To reduce the correlation loss caused by anonymizing, the records with the same *NSAs* values should first be divided into blocks to meet the anonymizing demands of ℓ-diversity [1] [14] [15] (i.e., for each block B there are at least ℓ records with different *SA* values and at most $\lfloor |B|/\ell \rfloor$ records with the same *SA* value, otherwise the actual *SA* values of some individuals may be disclosed with a certainty higher than $1/\ell$). Then, the records of each block in the partition are anonymized, so that the *SA* values appeared in the anonymized block all may be the actual *SA* values of the individuals whose records are in the block.

1.1. Motivation

The optimal partitioning, which puts the records with same *NSAs* values into the same blocks to meet the anonymizing demands of ℓ-diversity, is *NP*-hard [7] [11]. So in the (initial) partition there are usually many blocks having more than ℓ different *SA* values and uneven frequencies of *SA* values (i.e., there are more than ℓ records with different *SA* values and the numbers of the records with different *SA* values are unequal). Therefore, the average probabilities, which individuals are assigned to their actual *SA* values in the anonymized blocks of these blocks, are smaller. The smaller values mean that there are more correlation loss (less utility), as stated in Section III. If we refine the initial partition, such that the average probabilities of the anonymized blocks of refining partition are maximized (which are certainly higher than that of the initial partition), then the anonymized data of the refining partition would preserve more utility than that of the initial partition. So a problem arises: anonymizing a dataset such that the *SA* values of individuals should not be inferred with a certainty higher than $1/\ell$ while the correlation loss is as less as possible. We believe this is an issue need be addressed.

1.2. Contributions

In this paper, we systematically study anonymization on the refining partition of initial partition in privacy preserving data publishing. Our contributions include the following.

First, we propose an optimizing model for anonymizing a dataset so that the ℓ-diversity is preserved while the correlation loss is as less as possible. According to the model, we propose the approach of anonymizing on the refining partition of initial partition, so that the anonymized data of the refining partition preserves more utility than that of the initial partition.

The second, although the refining partition can be used on top of any existing partitioning approach, to further reduce the correlation loss, we present a partitioning approach based on the lexicographic and *NSAs* sorting by correlation (between *SA* and each of *NSAs*). This approach increases the utility of published data, as it preferentially ensures that a part of the *NSAs* (more interrelated with *SA*) values of the records of blocks are the same, while the records of blocks are anonymized, the correlations are retained.

The third, we show the validity of our approach from several aspects. We conduct extensive workload experiments with real data set. The results confirm that our approach greatly improves the utility of anonymized data.

The rest of the paper is organized as follows. Section II briefly reviews related work. Section III presents our problem definition, optimizing model and our purposes. In section IV, we provide our anonymization on refining partition. Section V experimentally evaluates the effectiveness of our methodology. Section VI gives conclusions and directions for future work.

2. Related Work

In anonymized data publishing, to prevent the actual sensitive values of individuals from being revealed with high certainty, generalization [14] [15] [3] [7] [11] (and non-homogeneous generalization [19]) transforms the *NSAs* values of the records of each block into ”less specific” values. Therefore, the information loss caused by generalization or non-homogeneous generalization includes the loss of the *NSAs* values of records, the correlation loss between *SA* and *NSAs*, and the correlation loss among *NSAs* [13]. Anatomy separates *SA* from *NSAs* by randomly permuting the *SA* values of records in each block [20] [16] [5]. Randomization [12] [18] [2] replaces the *SA* value of each record in each block with a retention probability p by a value randomly selected from the value-set consisted of the *SA* values of the records within the same block. Thus, there is only correlation loss between *SA* and *NSAs* caused by anatomy [20][16][5] or randomization [12] [18], [2], since these methods publish the *NSAs* values of records in their original forms, there is not the loss of the *NSAs* values of records and the correlation loss among *NSAs*. After the records have been partitioned to blocks, slicing [13] and disassociation [17] divide the attributes into columns. Disassociation partitions the attributes into columns based on the items of the values of the attributes. Therefore, the disassociation causes more correlation loss among the attributes of different columns. Slicing partitions the attributes based on the correlation among the attributes, the intersections between the columns and the blocks are buckets, and the tuples in each bucket are randomly permuted. In the buckets with *SA*, the *NSAs* values of the tuples are generalized. Thus, the information loss caused by slicing includes the correlation loss between *NSAs* and *SA*, the loss of a part of *NSAs* values of records, the correlation loss among these *NSAs*, and the correlation loss between these *NSAs* and the other *NSAs*, as the *NSAs* values are generalized.

In general, in the records blocks with the anonymizing demand of ℓ-diversity, the approach of generalization uniformly anonymizes the *NSAs* values of records. Yet, the method of non-homogeneous generalization first arranges the records of the blocks in a cycle, in which the adjacent ℓ records have mutual different *SA* values. Then, the *NSAs* of each adjacent ℓ records respectively are anonymized to an anonymized record (the method is called ring handling). Thus, comparing with the approach of generalization, in the same partition, to generate an anonymized record, the less records are processed in the method of non-generalizaiton. The less records are processed, the *NSAs* of the anonymized records may be more specific, and the information loss may be less. Therefore, the *NSAs* values of records of non-homogeneous generalization may be more specific than that of generalization, non-homogeneous generalization may reduce the information loss caused by the generalization of *NSAs* values of records. Yet, the method (ring handling) of non-homogeneous generalization could not be used for other anonymization techniques (such as anatomy, slicing and randomization, etc), i.e., the method of ring handling could not be used to reduce the information loss caused by other anonymization techniques. However, our method can be used to reduce the information loss caused by all the above anonymization techniques, since our refining partition is the local optimal partition of initial partition, and anonymization on the refining partition only cause less information loss.

In anonymized data publishing, to reduce the information loss, the records with the same *NSAs* values should first be divided into blocks with the anonymizing demands of ℓ-diversity. Then, the records of each block in the partition are anonymized. There are many partition methods [19] [20] [5] [6] [22] [21], but none of them meets the demands of the optimizing model, since the partition methods in [19] [5] [6] [22] [21] could not ensure that in each block of the partition the frequencies of different *SA* values are uniform and the numbers of different *SA* values is minimized, and the method in [20][8] does not take into account the *NSAs* values of the records. In addition, all the above methods do not take into account the correlation between *SA* and each of *NSAs*. Therefore, as stated in Section III, while the records of the blocks in the partition divided by these methods are anonymized by the above anonymizing techniques, there is more correlation loss.

3. Problem Definition

Consider a dataset T, in which there are 3 classes of attributes: (1) identifier attributes (*IDs*), such as name, social security number, *IDs* are removed in the published table to prevent immediate identification of individuals; (2) sensitive attribute (*SA*), the *SA* of individual's record is the sensitive value of individual; (3) non-sensitive attributes (*NSAs*), which contain all attributes that do not fall into the previous two categories and which taken together, can potentially identify an individual. Each record in T represents a distinct individual, and each *SA* value must be distinctively different. We introduce the 'different' in two aspects: (1) if *SA* is a nominal attribute that each value in a *SA* value set of T would be semantic distinguishable, and (2) if *SA* is a numerical attribute the difference of any two values are greater than a certain threshold. Otherwise, T should be preprocessed.

3.1. Attack Assumption and ℓ-diversity

We assume an attacker may obtain the *NSAs* and the *IDs* of any record in T by sources other than T^* (e.g. a public voters table). Let H be the attacker's knowledge containing *NSAs* values and *IDs* values of all known individuals. In the worst case, the attacker may access to the *NSAs* of every individual, thus by joining H and T^* on *NSAs*, record t^* in T^* may be linked to an individual (i.e., t^* is the anonymized record of the individual's record). ℓ-diversity principle aims at preventing the attacker from finding an individual's actual *SA* value with a probability higher than $1/\ell$.

Definition 1 (ℓ-diversity principle *[14], [15]*) *Let T^* be an anonymized data table of T. A records block B^* of T^* is ℓ-diversity, if there are at least ℓ different SA values, which may be the actual SA values of the records in B^*, and the numbers of the different SA values are all not more than $\lfloor |B^*|/\ell \rfloor$. T^* is ℓ-diversity, if all the blocks of T^* are ℓ-diversity.*

Even if the attacker has known that the record of victim is included in B^* by joining H and T^* on *NSAs*, the attacker could not infer victim's actual *SA* value with a probability higher than $1/\ell$, since at least ℓ different *SA* values all may be victim's actual *SA* value, and the numbers of the *SA* values are not more than $\lfloor |B^*|/\ell \rfloor$.

3.2. Our Goal

While the original block B is anonymized (to generate B^*), the anonymization methods all destroy the correlations. In addition, as stated in Section II, some methods also lose other information, since the *NSAs* values of records have been dealt. As H may be obtained from public resources, data accepter may prefer to care the correlations between *SA* and *NSAs*.

Therefore, our goal is to acquire a local optimal partition of T such that the anonymized table T^*, which is consisted of the anonymized blocks (of the blocks of the partition), meets the following two conditions:

(1) ℓ-diversity principle is satisfied.

(2) the correlation loss between *SA* and *NSAs* is as less as possible.

3.3. Problem Definition and Optimizing Model

Consider any anonymized block B^* in T^* (with ℓ-diversity). Let B be the original block of B^*, $t_1, t_2, \ldots, t_n$ be the records of $B(n = |B|)$ and $t_1^*, t_2^*, \ldots, t_n^*$ be the corresponding anonymized records in B^*. Assume the *SA* value-set (multiset) composed of the *SA* values appeared in B are $S = \{s_1 : c_1, s_2 : c_2, \ldots, s_m : c_m\}$; m is the number of different *SA* values, $(m \geq \ell)$; c_i is the number of the records with the *SA* value s_i (in B). For each $c_i(1 \leq i \leq m)$, $c_i \leq \lfloor |B|/\ell \rfloor$, (i.e., $c_i/|B| \leq 1/\ell$), since T^* is ℓ-diversity.

Property 1 *the mean value of the probabilities, which the records (t^*) in B^* are assigned to their actual SA values ($t\,[SA]$), is* $\overline{p(t^*, t[SA])} = \sum_{i=1}^{m} (\frac{c_i}{n})^2 \leq \frac{1}{\ell}$.

Proof. $\overline{p(t^*, t[SA])} = \frac{1}{n} \sum_{i=1}^{n} p(t_i^*, t_i[SA]) = \frac{1}{n}(c_1 \times \frac{c_1}{n} + c_2 \times \frac{c_2}{n}$

$+ \ldots + c_m \times \frac{c_m}{n}) = \frac{1}{n} \sum_{i=1}^{m} \frac{c_i \times c_i}{n} = \sum_{i=1}^{m} (\frac{c_i}{n})^2$.

$\forall t_i^* \in B^*, 1 \leq i \leq n, p(t_i^*, t_i[SA]) \leq \frac{1}{\ell}$, as B^* is ℓ-diversity.

Thus, $\frac{1}{n} \sum_{i=1}^{n} p(t_i^*, t_i[SA]) \leq \frac{1}{\ell}$.

Therefore, $\overline{p(t^*, t[SA])} = \sum_{i=1}^{m} (\frac{c_i}{n})^2 \leq \frac{1}{\ell}$.

The precise correspondences (such as an actual *SA* value is assigned to an individual in *B*) are all converted to imprecise correspondences (such as m different *SA* values are assigned to the individual in B^*) by anonymization methods. Thus, the smaller the value of m is, the more precise correspondence is (i.e., the less correlation loss is), and the higher the probability value is, the more precise correspondence is (i.e., the less correlation loss is). However, ℓ is the minimal value of m and $1/\ell$ is the maximal value of the probability, due to ℓ-diversity.

Therefore, for each block B^* of T^*, the bigger the value of the average probability $\overline{p(t^*, t[SA])}$ is, the less correlation loss is (i.e., the more precise correspondence is). In addition, as $\sum_{i=1}^{m} c_i = n$,

$$\overline{p(t^*, t[SA])} = \sum_{i=1}^{m} (\frac{c_i}{n})^2 = \frac{c_1^2 + c_2^2 + \ldots + c_m^2}{(c_1 + c_2 + \ldots + c_m)^2}.$$

Having computed the partial derivatives of $\overline{p(t^*, t[SA])}$ for $c_1, c_2, \ldots, c_m$, we acquire the conditional extreme value, i.e., while the values of $c_1, c_2, \ldots, c_m$ are the same and m is ℓ, the value of $\overline{p(t^*, t[SA])}$ is maximized, the maximal value is $1/\ell$.

However, the optimal partition that put the records with the same *NSAs* values into the same blocks (with the demands of ℓ-diversity) is *NP*-hard [7] [11]. Thus, there are often many blocks having more *SA* values and uneven frequencies of *SA* values, such that the $\overline{p(t^*, t[SA])}$ values of the anonymized blocks of these blocks are lower. To generate ℓ-diverse T^* (of T) so that the correlation loss is as less as possible, we propose an optimizing model.

Definition 2 (Optimizing model) *Let T^* be an anonymized data table of T. For each block B^* of T^* (let B^* be the anonymized block of B), if the following conditions (1), (2), (3) and (4) are satisfied, and the $\overline{p(t^*, t[SA])}$ is maximized, then T^* is an optimizing ℓ-diverse anonymized data table.*

(1) A part of NSAs (having higher correlation with SA) of the records (of B) are the same values;
(2) $c_i/n \leq 1/\ell, 1 \leq i \leq m$;
(3) $\sum_{i=1}^{m} c_i = n$;
(4) $\ell \leq m \leq n$.

In the best case, the *NSAs* of the records of *B* should have the same values. While the records with different *NSAs* values must be merged to the block for meeting the demands of ℓ-diversity, we should ensure that a part of *NSAs* (having higher correlation with *SA*) of the records of B are the same values, as stated in the condition (1), i.e., we should as less as possible break the correlations (between *NSAs* and *SA* in T) in B^*. In addition, the conditions (2), (3) and (4) ensure that B^* is ℓ-diversity.

4. Anonymization on Refining Partition

We follow the following framework. (1) The records are divided into blocks with the anonymizing demands of ℓ-diversity. (2) Based on the optimizing model, the initial partition is refined. (3) The records of each block of the refining partition are anonymized by the methods such as anatomy, generalization or slicing, etc.

4.1. Initial Partitioning

To preferentially retain the correlations between a part of *NSAs* (having more interrelated with *SA*) and *SA*, which further increases the data utility of anonymized table, we also propose a partitioning approach based on the lexicographic and *NSAs* sorting by the correlation between *NSA* and *SA*.

1) Computing correlation

Mean-square contingency coefficient [4] is a chi-square measure of correlation between two categorical attributes. For continuous attributes, we first apply discretization to partition the range of a continuous attribute into intervals and then treat the collection of interval values as a discrete value-set. Given a *NSA* (A_1) with value-set of data table

$v_{1_1}, v_{1_2}, \ldots, v_{1_{d_1}}$, and *SA* with value-set of data table $s_1, s_2, \ldots, s_d$. Their sizes of value-sets are thus d_1 and d, respectively. The mean-square contingency coefficient between A_1 and *SA* is defined as:

$$\phi^2(A_1, SA) = \frac{1}{min\{d_1, d\} - 1} \times \sum_{i=1}^{d_1} \sum_{j=1}^{d} \frac{(f_{ij} - f_{i.}f_{.j})^2}{f_{i.}f_{.j}}.$$

Here, $f_{i.}$ and $f_{.j}$ are the fraction of occurrences of v_{1_i} and s_j in the data, respectively. The f_{ij} is the fraction of co-occurrences of v_{1_i} and s_j in the data. The $f_{i.}$ and $f_{.j}$ are the marginal totals of f_{ij} : $f_{i.} = \sum_{j=1}^{d} f_{ij}$ and $f_{.j} = \sum_{i=1}^{d_1} f_{ij}$. Obviously, $0 \leq \phi^2(A_1, SA) \leq 1$. The higher the value of $\phi^2(A_1, SA)$ is, the more related between A_1 and *SA* are.

2) Partitioning

Having computed the correlations for *SA* and all *NSAs* of data table T, the *NSAs* of T are sorted by their correlations in descending. Then, the records of T are sorted by lexicographic *NSAs* order. The records are partitioned in a top-down fashion. First, we consider attribute A_1, which the value of $\phi^2(A_1, SA)$ is the biggest among all *NSAs* $(A_1, A_2, \ldots, A_{|NSAs|})$, and put records with the same A_1 value in the same block. This results in a set of blocks $B_1, B_2, \ldots, B_m$, (assume $d_1 = m$). However, some blocks may not satisfy the anonymizing demands of ℓ-diversity. For each such block B_j, we random find a neighboring block B_x (either B_{j-1} or B_{j+1}), and merge B_x with B_j. After we have done with A_1, we recursively partition the resulting blocks using the next attribute in order (i.e., A_2). In some blocks, the records may have different A_1 values (due to merging). For such blocks, we do not attempt to further decompose them recursively using another attribute. The partitioning strategy is repeated until all blocks are finalized or there are no more attributes that can be used for recursive partitioning. The detailed process is shown in Algorithm 1.

Each block of the initial partition generated by Algorithm 1 satisfies the anonymizing demands of ℓ-diversity. As shown in line 1 of Algorithm 1, B is divided into B_1, B_2,. . .,B_m using the current NSA (i.e., A_i). For each B_j which do not meet the anonymizing demands of ℓ-diversity, a neighboring block of B_j (i.e., B_x) is merged to B_j. At worst, $B_1, B_2, \ldots, B_m$ are all merged to the block (i.e., B), and B satisfies the anonymizing demands of ℓ-diversity (as stated by the Precondition of Algorithm 1). Thus, at the end of the iterations (line 2 $\sim$ 5), the sub-blocks of B must meet the anonymizing demands of ℓ-diversity. Some sub-blocks of B, which have different A_i values (due to merging), are output. In some sub-blocks of B, the records have the same A_i values. If $i = |NSAs|$, output such sub-blocks. Otherwise, the partition attempts to further decompose them until all blocks are finalized or there are no more attributes that can be used for recursive partitioning. Therefore, as stated above, each block of the initial partition generated by Algorithm 1 satisfies the anonymizing demands of ℓ-diversity.

4.2. Refining Partitioning and Anonymizing

To maximize the mean probabilities which individuals are assigned to their actual *SA* values, according to the optimizing model, in each block of partition there should exactly be ℓ different *SA* values, and the frequencies of different *SA* values should be higher

Algorithm 1 Partitioning (B, ℓ, A_i)

```
Precondition: $B$ satisfies the anonymizing demands of $\ell$-diversity.
Input: a set of records $B$, parameter in $\ell$-diversity, attribute $A_i$ is used for partitioning.
Output: sub-blocks of $B$.
Partition $B$ into $B_1, B_2, \ldots, B_m$ using $A_i$
//Assume $|A_i| = m$.
for each $B_j$ which do not meet the anonymizing demands of $\ell$-diversity do
    Random find $B_x$ as a neighboring block of $B_j$
    // $B_x$ may be $B_{j-1}$ or $B_{j+1}$
    Merge $B_j$ and $B_x$
end for
for each $B_j$ do
    if $(\exists t_a, t_b \in B_j \wedge t_a[A_i] \neq t_b[A_i]) \vee (A_i = A_{|NSAs|})$ then
        Output($B_j$) //Output the block $B_j$
    else
        Partitioning $(B_j, \ell, A_{i+1})$ // recursive call
    end if
end for
```

and uniform. Therefore, we refine the initial partition such that any block of the refining partition is composed of ℓ records with mutual different *SA* values. The residual records of the neighbor block of initial partition are merged to form a sub-block (composed of ℓ records with different *SA* values) of refining partition. In this case, the $\overline{p(t^*, s)}$ value of each anonymized block of the block in the refining partition is the maximum value (i.e., 1/ℓ). Finally, the last residual records (which the number of records with different *SA* values is less than ℓ) are respectively inserted into the corresponding sub-blocks of the refining partition. The detailed process is shown in Algorithm 2.

Property 2 *In Algorithm 2, for each B_i ($1 \leq i \leq m$), at the end of the iterations (line 4 $\sim$ 8), the number of nonempty buckets is $|B_i|$ mod ℓ, and there is only one record in each nonempty bucket.*

Proof. For each iteration, ℓ records with the mutual different *SA* values are removed from the ℓ largest nonempty buckets (i.e., in each bucket, only one record is random removed). So the iterations all are executed $\lfloor |B_i|/\ell \rfloor$ times, denoted as I_1, I_2, …, $I_{\lfloor |B_i|/\ell \rfloor}$, respectively.

Otherwise, we assume the iterations all are executed $\lfloor |B_i|/\ell \rfloor - 1$ times. At the end of the iterations, the number of the nonempty buckets is at most ℓ-1, (otherwise, the iterations could not have terminated). Then there is a set of nonempty buckets with at least 2 records (as the number of the residual records (termed x) is $\ell \leq x < 2\ell$ and the number of the nonempty buckets is at most $\ell - 1$).

Let a residual buckets (*rb*) have at least 2 records. Before iteration $I_{\lfloor |B_i|/\ell \rfloor - 1}$ starts, at most ℓ-1 buckets (including *rb*) have at least 3 records (otherwise, there would be ℓ nonempty buckets after $I_{\lfloor |B_i|/\ell \rfloor - 1}$, contradicting the fact that $I_{\lfloor |B_i|/\ell \rfloor - 1}$ is the last iteration). Thus, *rb* loses a record for $I_{\lfloor |B_i|/\ell \rfloor - 1}$, meaning that, before $I_{\lfloor |B_i|/\ell \rfloor - 1}$, the *rb* has at least 3 records.

Similarly, before $I_{\lfloor |B_i|/\ell \rfloor - 2}$, at most $\ell - 1$ buckets (including *rb*) have at least 4 records (otherwise, there would be ℓ buckets with at least 3 records after $I_{\lfloor |B_i|/\ell \rfloor - 2}$, contradicting our earlier analysis). Thus, *rb* loses a record for $I_{\lfloor |B_i|/\ell \rfloor -2}$, meaning that, before $I_{\lfloor |B_i|/\ell \rfloor -2}$, the *rb* has at least 4 records.

Carrying out the same discussion to the other iterations, we arrive at a fact that the *rb* has at least $\lfloor |B_i|/\ell \rfloor + 2$ (i.e., $\lfloor |B_i|/\ell \rfloor + 1$) records before I_1. This fact violates that there are at most $|Bi|/\ell$ records with the same *SA* values in B_i.

For the similarly reason, we could get the fact, which there are at least $\lfloor |B_i|/\ell \rfloor + 1$ records in a bucket before I_1, and this fact violates that there are at most $\lfloor |Bi|/\ell \rfloor$ records with the same *SA* values in B_i, if the iterations are executed less than $\lfloor |B_i|/\ell \rfloor$ times. Thus, when the iterations terminate, it must have been executed $\lfloor |B_i|/\ell \rfloor$ times. So the number of the residual records is $|B_i|$ mod ℓ. The residual records must have mutual different *SA* values (i.e., each non residual bucket has only one residual record.), otherwise, assume there are two residual records having the same *SA* value s, using the above similarly analysis, before I_1, there must be $\lfloor |B_i|/\ell \rfloor + 1$ records have the same *SA* value s. This fact violates that there are at most $|B_i|/\ell$ records with the same *SA* values in B_i.

Property 3 *In Algorithm 2, for each B_i ($1 \leq i \leq m$), at the end of the iterations (line 4 $\sim$ 8), $\lfloor |B_i|/\ell \rfloor$ blocks (i.e., Sub_B[i_1], Sub_B[i_2], . . ., Sub_B[$i_{\lfloor |B_i|/\ell \rfloor}$]) are generated. If $|B_i|$ mod ℓ is not zero, then for each residual record (t), there is a Sub_B[i_c], ($1 \leq c \leq \lfloor |Bi|/\ell \rfloor$), in which the SA values of the records are different with t[SA].*

Proof. Assume, on the contrary, there is a residual record (*t*), for $\forall$ Sub_B[i_c], ($1 \leq c \leq \lfloor |B_i|/\ell \rfloor$), $\exists t' \in$ Sub_B[i_c] $\wedge$ *t*[*SA*]= *t*′[*SA*]. Then, the number of the records (having the same *SA* value with *t*) is $\lfloor |B_i|/\ell \rfloor + 1$. This fact violates that there are at most $\lfloor |B_i|/\ell \rfloor$ records with the same *SA* values in B_i. Thus, for each residual record (*t*), there is a Sub_B[i_c], ($1 \leq c \leq \lfloor |B_i|/\ell \rfloor$), in which the *SA* values of the records and *t* are mutual different.

Property 4 *In Algorithm 2, for each record t in R_B (line 16), there is a block Sub_B[j] ($1 \leq j \leq count$), which t[SA] and the SA values of the records in Sub_B[j] are mutual different.*

Proof. For $\forall t \in$ R_B, $\exists B_i (1 \leq i \leq m) \wedge t \in B_i$, as *t* is a residual record of B_i (at the end of the iterations (line 4 $\sim$ 8)). By Property 3, we know that there is a Sub_B[i_c], ($1 \leq c \leq \lfloor |B_i|/\ell \rfloor$), in which the *SA* values of the records are different with *t*. Thus, the Sub_B[i_c] would be as Sub_B[j] ($1 \leq j \leq$ *count*).

Here we propose a hypothesis. Let T_3^* be an anonymized table, which is generated by the following two steps.

First, for each B_i,($1 \leq i \leq m$), if $|B_i|$ mod ℓ is not zero, then all the residual records are inserted into their corresponding sub-blocks (since as stated in Property 3, for each residual record (*t*), there is a Sub_B[i_c], ($1 \leq c \leq \lfloor |B_i|/\ell \rfloor$), in which the *SA* values of the records are different with *t*[*SA*]). Let $\overline{(p(t^*, s))'}$ be the mean probability, which individuals are assigned to their actual *SA* values in the anonymized blocks of the sub-blocks of B_i. As stated in Property 1, $\overline{p(t^*, s)}$ is the mean probability that individuals are assigned to their actual *SA* values in the anonymized blocks of B_i. Then, $\overline{p(t^*, s)} \leq \overline{p(t^*, s)'}$, as shown in Property 5.

Second, T_3^* is composed of the anonymized sub-blocks of these sub-blocks of B_1, B_2, . . ., B_m. Assume T_2^* is composed of the anonymized blocks of $B_1, B_2, \ldots, B_m$, and T_1^*

Algorithm 2 Refining-partition-anonymizing (T, ℓ)

Precondition: (1) T satisfies the anonymizing demands of ℓ-diversity; (2) the records in T have been sorted by lexicographical and *NSAs* attributes $(A_1, A_2, \ldots, A_{|NSAs|})$ ordering.

Input: a data table T, the parameter in ℓ-diversity.

Output: anonymized table T^*.

```
Partitioning $(T, \ell, A_1)$ // also by others partition approach
// Let $B_1, B_2, \ldots, B_m$ be the blocks of initial Partition
// Let Sub_B be an array consisted of sub-blocks
// Let count be counter, which initial value is 0
// R-B is used to retain residual records, which
// initial value is $\Phi$.
for each $B_i$ do
    Hash the records in $B_i$ by their SA values to buckets
    //each bucket per SA value; at least $\ell$ non-empty hash
    //buckets due to at least $\ell$ records with the different SA values in $B_i$
    while (there are at least $\ell$ non-empty hash buckets) do
        Take $\ell$ largest non-empty buckets $b_1, b_2, \ldots, b_\ell$
        Set count= count+1
        Set Sub_B[count]=$\Phi$ // i.e., Sub_B[count]is empty
        Random remove a record of each $b_j$ to Sub_B[count] // $1 \leq j \leq \ell$
    end while
    Remove the record of each nonempty bucket to R_B
    //only $|B_i|$ mod $\ell$ nonempty buckets, each nonempty
    //bucket per record, as shown in Property 2
    while there are at least $\ell$ records with different SA values in R_B do
        Set count= count+1
        Set Sub_B[count]= $\Phi$ // i.e., Sub_B[count]is empty
        Remove these $\ell$ records of neighbor blocks to Sub_B[count]
    end while
end for
if R_B is non empty then
    for For each $t$ in R_B do
        Find a Sub_B[j] such that
        $\exists B_i(t \in B_i) \wedge \exists Sub\_B[j](Sub\_B[j] \subseteq B_i) \wedge \forall t' \in Sub\_B[j](t'[SA] \neq t[SA])$,
        $1 \leq i \leq m, 1 \leq j \leq count$
        // as shown in Property 4
        Remove $t$ to Sub_B[$j$]
    end for
end if
Anonymize each Sub_B[$i$], $(1 \leq i \leq count)$
// by such as slicing, anatomy, generalization, etc.
Output the anonymized block of Sub_B[$i$]
```

is generated by anonymizing on refining partition. Then, T_1^*, T_2^* and T_3^* satisfy Property 6.

Property 5 $\overline{p(t^*, s)} \le \overline{p(t^*, s)'}$

Proof. As stated in Property 1, $S = \{s_1 : c_1, s_2 : c_2, \ldots, s_d : c_d\}$, d is the number of different *SA* values in B_i, c_j is the number of the records having s_j as *SA* value, and $\overline{p(t^*, s)} = \sum_{j=1}^{d}(\frac{c_j}{|B_i|})^2$. There are following two cases.

Case 1 (the frequencies of the *SA* values in B_i are uneven): assume $c_b(1 \le b \le d)$ is biggest among $c_1, c_2, \ldots, c_d$. Then $c_b \le \lfloor |B_i|/\ell \rfloor$ (as there are at most $\lfloor |B_i|/\ell \rfloor$ records with the same *SA* values in B_i, and c_d must be integer). Thus,

$$\sum_{j=1}^{d}(\frac{c_j}{|B_i|})^2 < \frac{1}{|B_i|} \times \sum^{|B_i|} \frac{c_b}{|B_i|} = \frac{c_b}{|B_i|} \le \frac{\lfloor |B_i|/\ell \rfloor}{|B_i|}.$$

Therefore, the average probability of B_i is $\overline{p(t^*, s)} < \frac{\lfloor |B_i|/\ell \rfloor}{|B_i|}$. But the average probability, which these individuals are assigned to their actual *SA* values in the anonymized blocks of Sub_B[i_1], Sub_B[i_2], ... and Sub_B[$i_{\lfloor |B_i|/\ell \rfloor}$] is

$$\overline{p(t^*, s)'} = \frac{1}{|B_i|} \sum_{x=1}^{\lfloor |B_i|/\ell \rfloor} \sum_{1}^{|Sub\_B[i_x]|} \frac{1}{|Sub\_B[i_x]|} = \frac{\lfloor |B_i|/\ell \rfloor}{|B_i|}.$$

Thus, $\overline{p(t^*, s)} < \overline{p(t^*, s)'}$.

Case 2 (the frequencies of the *SA* values in B_i are uniform): then the $c_1, c_2, \ldots, c_d$ are equal (denoted by c), and $|B_i| = c \times d$ and $d \ge \ell$ and $c/|B_i| \le 1/\ell$.

The mean probability, which the individuals are assigned to their actual *SA* values in the anonymized block of B_i, is

$$\overline{p(t^*, s)} = \sum_{j=1}^{d}(\frac{c_j}{|B_i|})^2 = \sum_{j=1}^{d}(\frac{c}{|B_i|})^2 = \frac{d \times c}{|B_i|} \times \frac{c}{|B_i|} = \frac{c}{|B_i|} = \frac{1}{d}.$$

The mean probability which these individuals are assigned to their actual *SA* values in the anonymized blocks of Sub_B[i_1], Sub_B[i_2], ... and Sub_B[$i_{\lfloor |B_i|/\ell \rfloor}$] is

$$\overline{p(t^*, s)'} = \frac{1}{|B_i|} \sum_{x=1}^{\lfloor |B_i|/\ell \rfloor} \sum_{1}^{|Sub\_B[i_x]|} \frac{1}{|Sub\_B[i_x]|} = \frac{\lfloor |B_i|/\ell \rfloor}{|B_i|} = \frac{\lfloor (d \times c)/\ell \rfloor}{d \times c} \ge \frac{\lfloor (d \times c)/d \rfloor}{d \times c} = \frac{c}{d \times c} = \frac{c}{d \times c} = \frac{1}{d}.$$

Thus, $\overline{p(t^*, s)} \le \overline{p(t^*, s)'}$.

According to Case 1 and Case 2, $\overline{p(t^*, s)} \le \overline{p(t^*, s)'}$.

Property 6 *The average probabilities, which individuals are assigned to their actual SA value in T_1^*, is more than that of T_3^* and T_2^*.*

Proof. According to Property 5, for each B_i,$(1 \le i \le m)$, $\overline{p(t^*, s)} \le \overline{p(t^*, s)'}$. Therefore, the average probability, which individuals are assigned to their actual *SA* value in T_3^*, is more than that of T_2^*.

In the following, we prove that the average probability of T_1^* is more than that of T_3^*.

Assume at the end of line 14 of Algorithm 2, there are $t_1, t_2, \ldots, t_p$ in R_B. Among them, assume $t_{a_1}, t_{a_2}, \ldots, t_{a_z}(z \le p)$ be a part of residual records of B_i, $(1 \le i \le m)$. At the end of the iterations (line $4 \sim 8$), let there be y residual records of B_i, then

$z \leq y \leq \ell - 1$, as some residual records of B_i and some residual records of the neighbor blocks of B_i may be merged to a new sub-block of refining partition (line 10 $\sim$ 13).

At the end of the iterations (line 4 $\sim$ 8), let the sub blocks of B_i be Sub_B[i_1], Sub_B[i_2], ..., Sub_B[$i_{\lfloor |B_i|/\ell \rfloor}$]. At the end of the iterations (line 16 $\sim$ 18), i.e., t_{a_1}, t_{a_2}, ..., t_{a_z} have been inserted into some sub blocks (by Property 4) of B_i. Then the average probability, which the individuals (whose records are in B_i) are assigned to their actual *SA* values in T_1^*, is:

$$\overline{p(t^*, s)''} = \frac{1}{|B_i|}\left(\frac{(y-z)}{\ell} + \sum_{x=1}^{\lfloor |B_i|/\ell \rfloor} \sum_{j=1}^{|Sub\_B[i_x]|} \frac{1}{|Sub\_B[i_x]|}\right) = \frac{\lfloor |B_i|/\ell \rfloor}{|B_i|} + \frac{(y-z)}{|B_i|}.$$

But the average probability, which the individuals (whose records are in B_i) are assigned to their actual *SA* values in T_3^*, is

$$\overline{p(t^*, s)'} = \frac{1}{|B_i|} \sum_{x=1}^{\lfloor |B_i|/\ell \rfloor} \sum_{1}^{|Sub\_B[i_x]|} \frac{1}{|Sub\_B[i_x]|} = \frac{\lfloor |B_i|/\ell \rfloor}{|B_i|}.$$

As $z \leq y$, $\frac{y-z}{|B_i|} \geq 0$.

Thus, $\overline{p(t^*, s)'} \leq \overline{p(t^*, s)''}$.

In addition, as stated above, the mean probability of T_3^* is more than that of T_2^*. Thus, the mean probability of T_1^* is more than that of T_2^*.

4.3. The Analysis of Anonymization on Refining Partition

1) Security Analysis

Assume that the T^* is generated by the Algorithm 2 based on slicing (anatomy or generalization, etc.). Since in each block of T^* there are at least ℓ different *SA* values and the numbers of the different *SA* values are the same value 1, the probabilities, which the individuals (linked to the anonymized block by their *NSAs* values) are assigned to their actual *SA* values, all are not more than $1/\ell$. Therefore, T^* is ℓ-diversity.

2) Utility Analysis

(1) T^* retains more correlations

In this section, we illustrate that the T^* generated by our approach has more data utility from the following two aspects.

Property 7 *The correlations between NSA (having more interrelated with SA) and SA are retained in* T^*.

Proof. Assume that the *NSAs* sorting by the correlations (between *NSAs* and *SA* in T) in ascending are $A_1, \ldots, A_y, , A_{|NSAs|}$. Let $y(1 \leq y \leq |NSAs|)$ be the maximum value, such that in each blocks of T^* the records have the same A_y value. Let $v_{x_1}, v_{x_2}, \ldots, v_{x_{d_x}}$ be the set which is consisted of the A_x values in T, and d_x be the size of such set. $s_1, s_2, \ldots, s_d$ is consisted of the *SA* values in T, and d be the size of such set.

For each x, $(1 \leq x \leq y)$, in each block of T^*, the records have the same A_x value, due to our partition approach. Thus, each record of T^* has the same A_x value with that of the homologous record of T, no matter what methods (such as generalization, anatomy (or bucketization) and slicing, etc.) have been used to generate T^*, i.e., the A_x values of the records in T^* retain their original forms in T. Therefore, for each v_{x_i} $(1 \leq i \leq d_x)$, the frequency of v_{x_i} value in T^* (i.e., $f^*_{x_i}.$) is equal to $f_{x_i}.$ (i.e., the frequency of v_{x_i} value in T).

The frequency of *SA* value in T^* is equal to that of the homologous *SA* values in T, as the above anonymizing methods do not destroy the frequency of *SA* value of T. Thus, for each $s_j(1 \leq j \leq d)$, the frequency of s_j in T^* is equal to that of in T, i.e., $f^*_{.j}$ is equal to $f_{.j}$.

In addition, let $f_{x_{i.j}}$ be the co-occurrences of v_{x_i} and s_j in T, i.e., there are $f_{x_{i.j}}$ records, which A_x values are v_{x_i} and *SA* values are s_j. Based on our partition approach we know that the records are divided into $f_{x_{i.j}}$ sub-blocks of the refining partition. In each sub-block, there only is one record has the *SA* value s_j, as the *SA* values of the records in each sub-block are mutual different. Since the A_x values of the records in each sub-block are the same value, and the A_x values of the records of $f_{x_{i.j}}$ sub-blocks are the same value, in the anonymized sub-blocks of the $f_{x_{i.j}}$ sub-blocks (in T^*), the A_x values of the records remain v_{x_i}. Therefore, while the co-occurrences of v_{x_i} and s_j are calculated in T^*, there only are the $f_{x_{i.j}}$ blocks having the co-occurrences, and there only is one co-occurrence in each block. Therefore, the co-occurrences of v_{x_i} and s_j in T^* (i.e., $f^*_{xi.j}$) remain $f_{x_{i.j}}$, which is the co-occurrences of v_{x_i} and s_j in T.

In conclusion, each parameter of $\phi^2(A_x, SA)$ calculated in T^* is the same as that is calculated in T. Thus, the $\phi^2(A_x, SA)$ value in the T^* is the same as that is calculated in T, since all the values of the parameters of the formula $\phi^2(A_x, SA)$ computed in T are the same as that are calculated in T^*. As $1 \leq x \leq y$, the *NSA* (having more interrelated with *SA*) and *SA* are retained in T^*.

Our approach increases the utility of T^*, as the correlations between a part of the *NSAs* (having more interrelated with *SA*) and *SA* are retained in T^*.

(2) T^* has lower bound of *RE*

The reconstruction error [20] [13] [12] [18] [2] (denoted by *RE*) often is used to measure the information loss (between *NSAs* and *SA*) of published table caused by selected anonymized technique, such as slicing, anatomy and generalization, etc.

As stated in [1], the actual *NSAs* values of the records of published table are easy to be acquired, in this section while the *RE* of the anonymized table is computed, the information loss of *NSAs* values of records may not be taken into account, even if the anonymized table is generated by generalization. For illustration purposes, and to show the *RE* of the anonymized table generated by our method (i.e., refining partitioning and anonymizing), assume that T^* is generated by anatomy, as all the *NSAs* values of the records in T^* retain original forms. We demonstrate that T^* has lower bound of *RE*, as shown in Property 8.

For each block B^* of T^* (B is the original block of B^*), let S_B be the *SA* value-set composed of all the *SA* values appeared in B. For each individual (t) whose record is linked to B^* by the *NSAs* values of t, the *RE* of t is the probability that all the values in S_B (which are not the actual *SA* value of t) are assigned to t.

Property 8 *T^* has lower bound of RE.*

Proof. Let $|T| = n$, and at the end of line 14 of Algorithm 2, there are r records in R_B. There are following two cases.

Case 1 ($r = 0$): by Algorithm 2 we know that each block B in T^* has ℓ individual records, and their *SA* values are mutual different. Therefore, the probability that each individual (t) is assigned to the actual *SA* value of t is $1/\ell$. So the *RE* of t is $1 - 1/\ell$. Thus the *RE* of all individual records in T^* are

$$\sum_1^n (1 - \frac{1}{\ell}) = n \times (1 - \frac{1}{\ell}).$$

Case 2 ($r \neq 0$): at the end of line 14 of Algorithm 2, there are $n - r$ individual records which have been divided into blocks, in which there are exactly ℓ individual records having mutual different *SA* values. According to the analysis of Case 1, the total *RE* of these records is

$$(n - r) \times (1 - 1/\ell).$$

Next, we show after inserting a residual record t to a block, in which the *SA* values of the records and the *SA* value of t are mutual different (as shown in line $16 \sim 19$ of Algorithm 2), the overall *RE* increases by 1. Without loss of generality, assume that t is inserted to a block (with d records). Before the insertion, following the derivation of Case 1, the total *RE* of the records in the block is $d \times (1 - \frac{1}{d})$. After the insertion, the total *RE* becomes $(d + 1) \times (1 - \frac{1}{d+1})$, so that the total *RE* of the records in the block increases by:

$$(d + 1) \times (1 - \frac{1}{d+1}) - d \times (1 - \frac{1}{d}) = 1.$$

As mentioned earlier, before the insertion step starts, the total *RE* equals $(n - r) \times (1 - \frac{1}{\ell})$. Therefore, after insertion all r residual records, the total *RE* becomes:
$(n - r) \times (1 - \frac{1}{\ell}) + r = n \times (1 - \frac{1}{\ell}) \times (1 + \frac{r}{n \times (\ell - 1)})$.
That is greater than the lower bound $n \times (1 - 1/\ell)$ by a factor of $(1 + \frac{r}{(n \times (l - \ell))})$.

Generally T usually is a large data table, the number of the residual records is far less than n, so $r \ll n \times (\ell - 1)$, i.e.,$1 + \frac{r}{n \times (\ell - 1)} \approx 1$.

Therefore, in this case, the total *RE* of the individual records in T^* is extremely close to the lower bound.

Actually the higher the probabilities that the individuals are assigned to their actual *SA* values are, the lower bound of *RE* is. In our approach, the probabilities that individuals are assigned to their actual *SA* values are increased, as stated in Property 5. Thus, T^* has lower bound of *RE*. In the same time, the lower bound of reconstruction error of T^* illustrates that our approach is valid to increase the data utility of the published data.

(3) Cost Analysis

Before we apply Algorithm 2, we need compute the correlations between *NSAs* and *SA*, then rearrange the order of *NSAs* and sort the records for Algorithm 1. Suppose the number of attributes in *NSAs* is b, each partition defined by the first attribute of *NSAs* recursively is repartitioned up to b times. In the worst case, the cost of the partition is $|T| \times (b + log(|T|))$. In addition, for each block B of initial partition, the iterations (line $4 \sim 8$) are executed $\lfloor |B|/\ell \rfloor$ times. In the worst case, the $|B|$ mod ℓ residual records need be inserted in the sub blocks of B (lines $16 \sim 18$). In generally, $|B|$, ℓ and b are negligible comparing with $|T|$. Therefore, the overall cost of anonymization on refining partition is $O(|T| \times log(|T|))$.

5. Experiments

All of the experiments are conducted in *Delphi7.0* and are run on an *Intel Core 2.8 GHz* machine and 2 *GB RAM* with *Windows XP*. We use the *Adult* dataset from the *UC Irvine* machine learning repository [10], which is comprised of data collected from *US census*. The dataset has been used in several literatures [14] [15] [11] [19] [13] for privacy preserving data publication. Records with missing values are eliminated, and there are 30718 valid records used in our experiments. The *Adult* dataset contains 15 attributes in total. We randomly project 8 attributes (*Age*, *Sex*, *Education-Level*, *Marital-Status*, *Race*, *Worklass*, *Country*, *Occupation*) from the original table as the experimental dataset, and the attributes for *Adult* dataset in our experiments are described in TABLE I. The *Occupation* is taken as *SA* and the other attributes are taken as *NSAs*.

Table 1. Description of the dataset

ID	Attribute	Type	Cardinality
1	*Age*	Continuous	72
2	*Sex*	Categorical	2
3	*Education-Level*	Continuous	16
4	*Marital-Status*	Categorical	7
5	*Race*	Categorical	5
6	*Work-Class*	Categorical	7
7	*Relationship*	Categorical	6
8	*Occupation*	Categorical	14

In this section, we illustrate that the T^* generated by our method has less correspondences (between *NSAs* values and *SA* values of records) loss. If the correspondences (between *NSAs* values and *SA* values of records) in T^* are not damaged, then for each *NSA* (A_x) of *NSAs*, there is not the correlation (between A_x and *SA*) loss, since the value of $\phi^2(A_x, SA)$ in T is equal to the value of $\phi^2(A_x, SA)$ in T^*. Therefore, the less the correspondence loss is, the less correlation loss is (the more data utility of T^* is).

The count queries [20] [12] [18] [2] usually are used to measure the correspondences loss of anonymized table. However, the methods only select a small part of records of T as query predicates. Thus, the count queries do not accurately measure the loss. To accurately measure the correspondences loss, the approach of [9] is used in this paper, all the records of T are taken as query predicates to query in anonymized table T^*. Assume S is the *SA* value-set composed of the *SA* values of the records in T. For each record (t) in T, let S_t be the *SA* value-set composed of *SA* values of the records having the same *NSAs* values with t. Obviously, for each s in $S - S_t$, in T there is not the record, which has the *NSAs* values t[*NSAs*] and has the *SA* value s. But in T^* the values in $S - S_t$ may be the *SA* values of t^* (the anonymized record of t), as the anonymization methods (such as anatomy, slicing, generalization, etc.) all disturb the correspondence between *NSAs* and

SA of the records of T. Thus, the probability, which the values in $S - S_t$ may be the *SA* value of t^* (in T^*), is taken as the normalized correspondences loss penalty (*NLP*) of the record t (as shown in Definition 3). The minimum value of *NLP*(t^*) is zero, and the correspondences loss is zero when record t^* is generated from t. The *GLP* is a normalized version of *NLP*. The smaller the value of *NLP*(t^*) is, the better data utility is achieved, so do as *GLP*(T^*). In this paper, the *GLP* is used to measure the correspondences loss of anonymized table generated by different anonymization approaches.

Definition 3 (*GLP [8],[9]*) *Let T^* be an anonymized data table of T. Assume that S is the SA value-set of T. Let t[SA] be the SA value of t, and S_t be a SA value-set consisted of the SA values of the records having the same NSAs values with t, i.e., $\forall t \in T, \exists S_t(t[SA] \in S_t)\forall t' \in T(t[NSAs] = t'[NSAs] \rightarrow t'[SA] \in S_t)$. The normalized correspondences loss penalty that generate record t^* from t (t and t^* belong to the same individual) is NLP. NLP(t^*)=$\sum_{s\in(S-S_t)} P_r(t^*[SA] = s)$. The GLP is a normalized version of the NLP.*

In the following discussion, our method anonymization on refining partition is denoted by *ARP*, the anonymization on the initial partition is denoted by *AIP*. We select bucketization as the anonymization approach of *ARP* and *AIP*, since we only to compare the correlation loss of anonymized data, as stated in our goal (of Section III). Since the partition of anatomy [20] (denoted by *AT*) often is used in other methods [19] [13] [18], we compare the *GLP* of *AT* with *ARP*. In addition, since non-homogeneous generalization (denoted by *NG*) also is used to reduce the information loss caused by generalization, as stated in literature [19], we compare the *GLP* of *NG* with *ARP*. Without loss generality, we also generate the partition of *NG* by Algorithm 1.

Our experiments demonstrate that the execution-time and correspondences loss of *ARP* comparing with *AIP*, *NG* and *AT*, when privacy level (ℓ), the size (n) of the dataset and the size (d) of *NSAs* of dataset are varied.

5.1. Varying Privacy Level

1) Experimental results

In this experiments, we set the size of the dataset n=20 thousands. With privacy level ℓ increase, the execution time and the *GLP* of four methods are shown in Fig. 1 and Fig. 2.

(1) With ℓ increase, the execution times of four approaches are increasing, as shown in Fig. 1. Among them, the execution time of *NG* is always maximal. The execution time of *ARP* is more than that of *AT* and *AIP*, and there are fluctuations in the execution time of *AT*.

(2) With ℓ increase, the *GLP* of four approaches are increasing, as shown in Fig. 2. Among them, the *GLP* of *AT* is always maximal. The *GLP* of *ARP* and *NG* are almost the same in statistical sense, and their values are always minimal.

2) The analysis of the experimental results

(1) For the experimental result (1), comparing with *ARP* and *NG*, the anonymization of *AIP* is directly applied on the initial partition, so *AIP* spends less time than that of *ARP* and *NG*. Comparing with *ARP*, *NG* separately anonymizes the *NSAs* of each record of each block of initial partition, but *ARP* entirety anonymizes the *NSAs* of the records of each sub-blocks. Thus, *ARP* spends less time than that of *NG*. Comparing with *AT*,

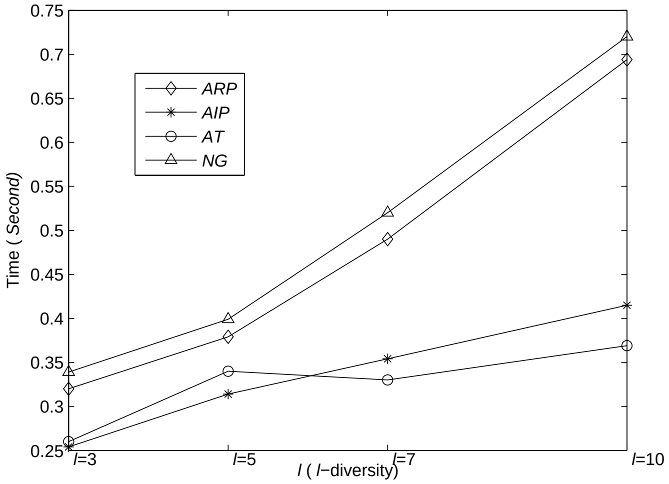

Fig. 1. Execution time; varying ℓ

the partition approach of *AIP* need spend more time on computing the information of *SA* values of blocks (for judging the demands of ℓ-diversity), the approach of *AT* only counts the numbers of records of blocks (for judging the demands of ℓ-diversity), as the *SA* values of the records of each block of *AT* are mutual different. The fluctuations in the execution time of *AT* is because of the randomization of the partition, and with ℓ increase, the execution time of *AT* is increasing, since it need spend more time on partitioning. With ℓ increase, the numbers of the records in the blocks of initial partition are increasing, so *AIP* spends more time on computing the information of *SA* values of blocks (for judging the demands of ℓ-diversity), and comparing with *AIP*, *ARP* and *NG* need spend more time to anonymize records. Therefore, with ℓ increase, the execution time of *AIP*, *ARP* and *NG* are increasing.

(2) For the experimental result (2), in each block of *AT*, the *SA* values of records are mutual different, and the *NSAs* values of records may be different since *AT* is not care about the *NSAs* values of records. Thus, the *GLP* of *AT* is always maximal. Although the records with same *NSAs* values are divided into same blocks of *AIP*, the numbers of different *SA* values in the blocks of *AIP* are always more than ℓ and the records with different *NSAs* values would be in the same block due to neighbor block merging (to meet the demands of ℓ-diversity). Therefore, although the *GLP* of *AIP* is less than that of *AT*, the *GLP* of *AIP* is still always more than that of *NG* and *ARP*, since comparing with that of *AIP*, each individual is assigned to less *SA* values in the anonymized table of *NG* and *ARP*. For the same reason the *GLP* values of *NG* and *ARP* are almost the same in statistical sense. With ℓ increase, each individual would be assigned to more *SA* values in the anonymized tables of four methods, but these values are not always the *SA* values of the records with same *NSAs* values as that of the individual, so the *GLP* values of four methods are increasing.

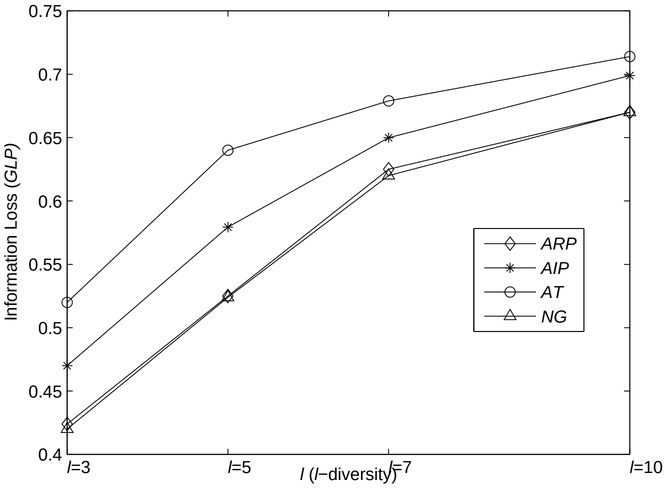

Fig. 2. The correspondences loss (*GLP*); varying ℓ

5.2. Varying the Size of Dataset

1) Experimental results

In this experiments, we set privacy level ℓ=5. With the size of the dataset n increase, the execution time and the *GLP* values of four methods are shown in Fig. 3 and Fig. 4.

(1) With n increase, the execution times of the four methods are increasing, as shown in Fig.3. Among them, the execution time of *NG* is always highest while the execution times of *AT* and *AIP* are less than that of *NG* and *ARP*.

(2) With n increase, the *GLP* of four methods are decreasing, as shown in Fig.4. Obviously, the *GLP* of *AT* is always the maximal, while the *GLP* of *NG* and *ARP* are always less than that of *AIP* and *AT*.

2) The analysis of the experimental results

(1) For the experimental result (1), with n increase, there are more records need be anonymized, so the execution times of four methods are increasing. Comparing with *NG* and *ARP*, the anonymization of *AIP* is directly applied on the initial partition, so the execution time of *AIP* is less than that of *ARP* and *NG*. The execution times of *ARP* is less than that of *NG*, as *NG* separately anonymizes the *NSAs* of each record of each block of initial partition, but *ARP* entirety anonymizes the *NSAs* of the records of each sub-blocks.

(2) For the experimental result (2), with n increase, there are more records with the same *NSAs* values in original data table, and the size of the *SA* value-set consisted of the *SA* values of the records with the same *NSAs* values is increasing. Thus, the probability that each record t takes *SA* values in value-set S_t (consisted of the *SA* values of the records having the same *NSAs* values with t) is increasing ($S - S_t$ is decreasing), so the *GLP* of four methods are decreasing.

As *AT* does not take into account that put the records with same *NSAs* values into same blocks, in the anonymized data of *AT*, the probability that each record t is assigned to the *SA* in value-set (S_t) is small. Thus, in the same n values, the *GLP* of *AT* is always the maximal. Comparing with *AT*, *AIP* divides the records with same *NSAs* values into same

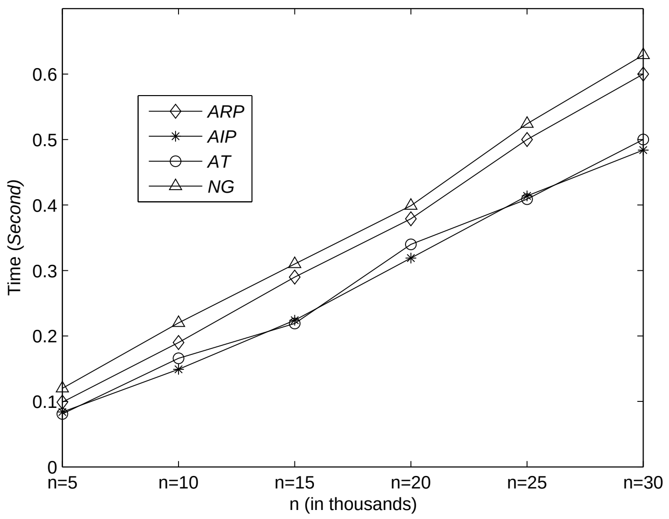

Fig. 3. Execution time; varying n

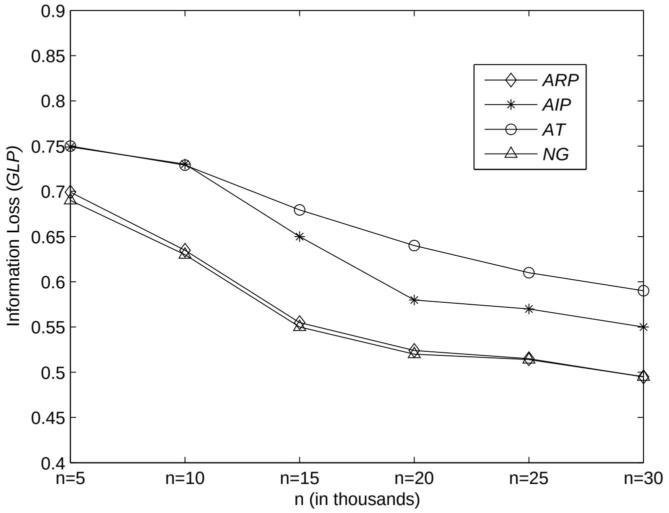

Fig. 4. The correspondences loss (*GLP*); varying n

blocks, so, in the same n values, the *GLP* of *AIP* is smaller than that of *AT*. However, in the blocks of *AIP*, there always are more than ℓ records, and the records with different *NSAs* values would be divided into the same block due to neighbor block merging (to meet the anonymizing demands of ℓ-diversity). Therefore, in the same n values, the *GLP* value of *AIP* is always more than that of *NG* and *ARP*. As *NG* and *ARP* are based on the same initial partition (generated by our method), and in the anonymized data of the two methods, the probabilities that individuals are assigned to their actual *SA* values are local maximal, the *GLP* values of *NG* and *ARP* are almost the same in statistical sense.

5.3. Varying the Size of *NSAs*

1) Experimental results

In this experiments, we set the privacy level ℓ=5 and set the size of a dataset $n = 20$ thousands. With the size (d) of the *NSAs* of the dataset increase, the execution times and the *GLP* values of four methods are shown in Fig. 5 and Fig. 6.

As stated above, there are 8 attributes in the above experiments dataset T, and the *Occuption* is taken as *SA*, the other attributes are taken as *NSAs*. The correlations are computed by the mean-square contingency coefficient (as mentioned in Section IV). Assume that the *NSAs* sorting by the correlations (between *NSAs* and *SA* in T) in ascending are $A_1, A_2, \ldots, A_7$. We respectively project attributes, $A_1SA, A_1A_2SA, A_1A_2A_3SA, \ldots$, $A_1A_2A_3A_4A_5A_6A_7SA$, from T as experimental datasets $T_1, T_2, \ldots, T_7$. Thus, the size (d) of the *NSAs* in T_1 is 1 (i.e., $d = 1$), the size (d) of the *NSAs* in T_2 is 2 (i.e., $d = 2$), ..., the size (d) of the *NSAs* in T_7 is 1 (i.e., $d = 7$).

The anonymized datasets of the 7 datasets ($T_1, T_2, \ldots, T_7$) respectively are generated by the four methods, the execution time and the *GLP* values of the four methods are shown in Fig. 5 and Fig. 6. The *abscissa* 1 is the anonymized dataset of T_1, *abscissa* 2 is the anonymized dataset of $T_2, \ldots$, the *abscissa* 7 is the anonymized dataset of T_7.

(1) With the size (d) of *NSAs* increase, the execution time of *AIP* increase, and the execution time of *AT* is invariable. Yet, the execution times of *ARP* and *NG* decrease at first, and then increase, as shown in Fig.5. Among them the execution time of *NG* is always highest, the execution time of *AIP* is always lowest, while the execution time of *ARP* is more than that of *AIP* and *AT*.

(2) With the size (d) of *NSAs* increase, the *GLP* of four methods are increasing, as shown in Fig.6. Among them, the *GLP* of *AT* is always highest, while the *GLP* of *NG* and *ARP* are less than that of *AIP*. The *GLP* of *NG* and *ARP* are almost the same in statistical sense, and their values are always minimal.

2) The analysis of the experimental results

(1) For the experimental result (1), with the size (d) of *NSAs* increase, more *NSAs* are used in recursively partition, so the initial partition of *AIP*, *ARP* and *NG* consume more times. *AIP* directly anonymize the records on the initial partition, so the execution time of *AIP* increases, and it is lower than that of *ARP* and *NG*. Although the execution time of the initial partition is low when d is small, the record-blocks of the initial partition may be larger, *ARP* and *NG* need to consume more time to anonymize the records of the larger records blocks. Therefore, while d is small, *ARP* and *NG* still consume more time. With d increasing, the records blocks of initial partition are smaller, *ARP* and *NG* only consume less time to anonymize the records of the record-blocks. However, with d increasing, more times need to be consumed on the initial partition. So with d is increasing, the execution

times of *ARP* and *NG* decrease at first, and then increase. As stated above, *ARP* always spends less time than that of *NG*, as *ARP* entirety anonymizes the *NSAs* of the records of each sub-blocks. As *AT* does not consider the *NSAs* of the records, the execution time of *AT* is invariable.

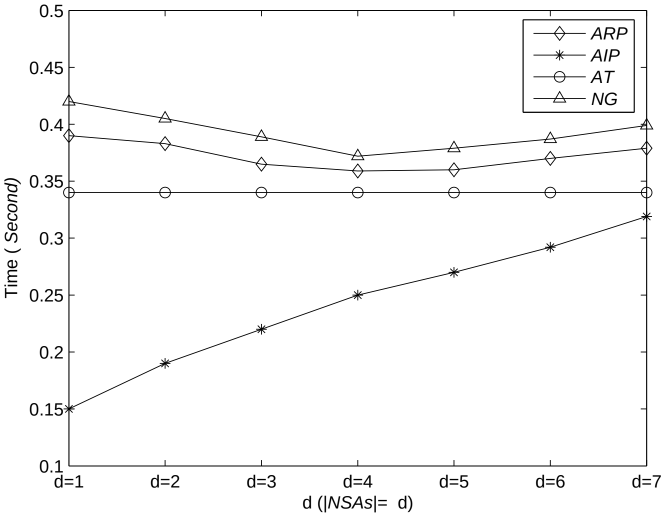

Fig. 5. Execution time; varying d

(2) For the experimental result (2), with the size (d) of *NSAs* increase, there are less records with the same *NSAs* values in original data table, the size of the *SA* value-set consisted of the *SA* values of the records with the same *NSAs* values is decreasing. Thus, the probability that each record t takes *SA* values in value-set S_t (consisted of the *SA* values of the records having the same *NSAs* values with t) is decreasing ($S - S_t$ is increasing), so the *GLP* of four methods are increasing. As stated above, Algorithm 1 ensures that the records with the same *NSA* (having more interrelated with *SA*) values are divided into same blocks, and without loss generality, we also generate the partition of *NG* by Algorithm 1, so anonymization on the records of the blocks causes less correspondence loss, especially while $d = 1$, the records with the same A_1 values are divided into the same blocks, the *GLP* values of *ARP*, *AIP* and *NG* are zero. In addition, as stated above, anonymization on refined partition causes the higher probability that records are assigned to their actual *SA* values in the anonymized tables, so the *GLP* of *NG* and *ARP* are less then that of *AIP*. As *AT* does not consider the *NSAs* of the records, the *GLP* of *AT* always is higher than that of the other methods.

From the above experiments, we can conclude that *ARP* and *NG* could retain more correspondences than that of *AIP* and *AT*, and although the *GLP* of *ARP* and *NG* are almost the same in statistical sense, *ARP* spends less times than that of *NG*.

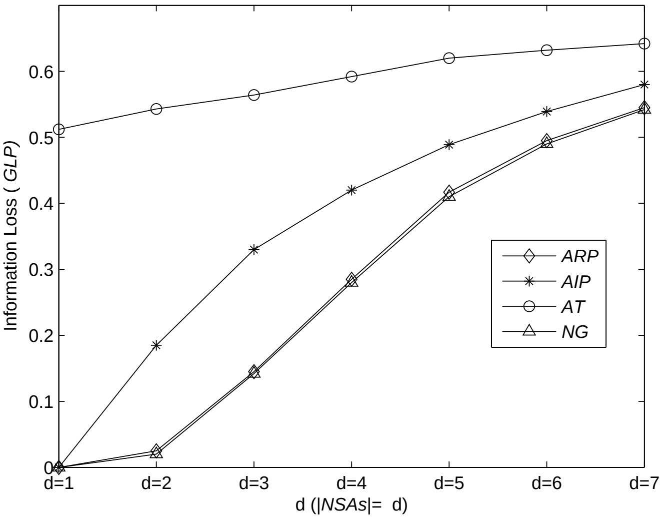

Fig. 6. The correspondences loss (*GLP*); varying d

6. Conclusions and Future work

An approach of anonymizing on the refining partition of initial partition (*ARP*) has been constructed, so that in same privacy level *ARP* preserves more utility than anonymization on the initial partition (*AIP*). In addition, *ARP* has more utility than that of *AT*. A method of initial partition also has been design, although in same privacy level and initial partition *ARP* and *NG* have almost the same information loss in statistical sense, *ARP* spends less times than that of *NG*, and *ARP* can be used for slicing, anatomy, randomization and generalization, etc., but *NG* only is used for generalization.

For future work, we may consider to optimize the partition of the dataset with multiple sensitive attributes, and apply our approach in practical privacy preserving data publishing.

Acknowledgment. We would like to thank the experts of ICSAI 2014 for their insightful comments. This research is supported by National Science and Technology Major Project of China: Research on the Secure database management system and advanced technology. The project number is 2010ZX01042-001-003-03.

References

1. Benjamin C. M., F., Ke, W., Rui, C., Philip S., Y.: Privacy-preserving data publishing: A survey of recent developments. Acm Computing Surveys 42(4), 2623–2627 (2010)
2. Chaytor, R., Ke, W.: Small domain randomization: same privacy, more utility. In: Proceedings of International Conference on Very Large DataBases. pp. 608–618. ACM, Singapore (2010)
3. Chi Wing, W., Jiuyong, L., Wai Chee, F., Ke, W.: (, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 754–759. ACM New York, NY, USA, Philadelphia, USA (2006)

4. Cramer, H.: Mathematical methods of statistics. Princeton, USA (1948)
5. D. J. Martin, D. Kifer, A.M.J.G., Halpern, J.Y.: Worst-case background knowledge for privacy-preserving data publishing. In: Proceedings of IEEE 23rd International Conference on Data Engineering. pp. 126 – 135. IEEE Computer Society, Istanbul, Turkey (2007)
6. Ghinita, G., Kalnis, P.: Fast data anonymization with low information loss. In: Proceedings of International Conference on Very Large DataBases. pp. 758–769. ACM, Austria (2007)
7. Gionis, A., Mazza, A., Tassa, T.: k-anonymization revisited. In: Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE). pp. 744–753. IEEE Computer Society, Cancun, Mexico (2008)
8. Hong, Z., Shengli, T., Kevin, L.: Privacy preserving data publication with features of independent l-diversity. Computer Journal 58(4), 549–571 (2014)
9. Hong, Z., Shengli, T., Meiyi, X.: Anonymization on refining partition: Same privacy, more utility. In: 2014 2nd International Conference on Systems and Informatics (ICSAI). pp. 998 – 1005. IEEE Computer Society, Shanghai China (2014)
10. http://archive.ics.uci.edu/ml/:
11. Junqiang, L., Ke, W.: On optimal anonymization for l+ -diversity. In: Proceedings of the IEEE 26th International Conference on Data Engineering (ICDE). pp. 213–224. IEEE Computer Society, Long Beach, California, USA (2010)
12. Ke, W., Chao, H., Fu, A.W.: Randomization resilient to sensitive reconstruction. arXiv preprint arXiv:1202.3179 (2012)
13. Li, T., Li, N., JianZhang, Ian, M.: Slicing: A new approach for privacy preserving data publishing. IEEE Transactions on Knowledge and Data Engineering 24(3), 561–674 (2010)
14. Machanavajjhala, A., Gehrke, J., Kifer, D.: L-diversity: privacy beyond k-anonymity. In: Proceedings of the 22nd International Conference on Data Engineering. pp. 24–24. IEEE Computer Society, Atlanta, GA, USA (2006)
15. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: L-diversity: privacy beyond k-anonymity. ACM Transactions on Knowledge Discovery from Data 1(1), 1–47 (2007)
16. Qing, Z., Koudas, N., Srivastava, D., Ting, Y.: Aggregate query answering on anonymized tables. In: Proceedings of IEEE 23rd International Conference on Data Engineering. pp. 116–125. IEEE Computer Society, Istanbul, Turkey (2007)
17. Terrovitis, M., Mamoulis, N., Liagouris, J., Skiadopoulos, S.: Privacy preservation by disassociation. Proceedings of the VLDB Endowment 5(10), 944–955 (2012)
18. Wai Chee, F., Jia, W., Ke, W., Chi Wing, W.: Small count privacy and large count utility in data publishing. Eprint Arxiv 50(8), 20–31 (2012)
19. Wai Kit, W., Mamoulis, N., David Wai Lok, C.: Non-homogeneous generalization in privacy preserving data publishing. In: Proceedings of the ACM SIGMOD International Conference on Management of Data. pp. 139–150. ACM, Indianapolis, Indiana, USA (2010)
20. Xiaokui, X., Yufei, T.: Anatomy: Simple and effective privacy preservation. In: Proceedings of International Conference on Very Large Data Bases. pp. 139–150. ACM, Seoul, Korea (2006)
21. Xiaokui, X., Yufei, T., Nick, K.: Transparent anonymization: Thwarting adversaries who know the algorithm. Acm Transactions on Database Systems 35(2), 1–48 (2010)
22. Xin, J., Nan, Z., Gautam, D.: Algorithm-safe privacy-preserving data publishing. In: Proceedings of International Conference on Extending Database Technology. pp. 633–466. ACM, Indianapolis, Indiana, USA (2010)

Hong Zhu is a professor in School of Computer Science and Technology in Huazhong University of Science and Technology. Her research interests include data security and big data processing. Email: zhuhong@hust.edu.cn

Shengli Tian, Ph.D. He is currently an associate Professor in School of Information Engineering at Xuchang University. He received B.S. degree in Computer Science Education from Xinyang Normal University in 2002, the M.Sc. degree from Henan University in 2007, and the Ph.D. degree from Huazhong University of Science and Technology in 2014. He research interests include information security technology and machine learning, etc. Email: zelintian@gmail.com

Genyuan Du, Ph.D. He is currently an associate Professor in the International School of Education at Xuchang University. He received B.S. degree in Computer Science and Technology from Henan Normal University in 1997, and the M.E. degree in Signal and Information Processing from Chengdu University of Technology in 2005, and the Ph.D. degree in the college of Information Engineering at Chengdu University of Technology in 2011. Research area: Parallel computing, Image retrieval, Spatial data mining, remote sensing image processing, remote sensing and computer networks. Mr. Genyuan Du is a senior member of the CCF. Email: xcdgy@163.com

Meiyi Xie is a lecturer in School of Computer Science and Technology in Huazhong University of Science and Technology. Her research interests include data security and big data processing. Email: xiemeiyi@hust.edu.cn

Received: December 12, 2014; Accepted: October 10, 2015.