Enhanced P-Sensitive K-Anonymity Models for Privacy Preserving Data Publishing

Xiaoxun Sun[*], **Hua Wang**[*], **Jiuyong Li**[**] **and Traian Marius Truta**[***]

[*]Department of Mathematics & Computing, University of Southern Queensland, Queensland, Australia.

[**]School of Computer and Information Science, University of South Australia, Adelaide, Australia.

[***]Department of Computer Science, Northern Kentucky University, Highland Heights, KY, USA.

E-mail: {sunx, wang}@usq.edu.au; jiuyong.li@unisa.edu.au; trutat1@nku.edu;

Abstract. Publishing data for analysis from a micro data table containing sensitive attributes, while maintaining individual privacy, is a problem of increasing significance today. The k-anonymity model was proposed for privacy preserving data publication. While focusing on identity disclosure, k-anonymity model fails to protect attribute disclosure to some extent. Many efforts are made to enhance the k-anonymity model recently. In this paper, we propose two new privacy protection models called (p, α)-sensitive k-anonymity and (p^+, α)-sensitive k-anonymity, respectively. Different from previous the p-sensitive k-anonymity model, these new introduced models allow us to release a lot more information without compromising privacy. Moreover, we prove that the (p, α)-sensitive and (p^+, α)-sensitive k-anonymity problems are NP-hard. We also include testing and heuristic generating algorithms to generate desired micro data table. Experimental results show that our introduced model could significantly reduce the privacy breach.

1 Introduction

With the rapid growth in database, networking, and computing technologies, a large amount of personal data can be integrated and analyzed digitally, leading to an increased use of data-mining tools to infer trends and patterns. This has raised universal concerns about protecting the privacy of individuals.

Many data holders publish their micro data for different purposes. However, they have difficulties in releasing information which does not compromise privacy. The traditional approach of releasing the data tables without breaching the privacy of individuals in the table is to de-identify records by removing the identifying fields such as name, address, and social security number. However, joining this de-identified table with a publicly available database (like the voters database) on attributes like race, age, and zip code (usually called quasi-identifier) can be used to identify individuals.

In order to protect privacy, Sweeney [17] proposed the k-anonymity model, where some of the quasi-identifier fields are suppressed or generalized so that, for each record in the modified table, there are at least $k-1$ other records in the modified table that are identical to it along the quasi-identifier attributes. For the Table 1, Table 2 shows a 2-anonymous

Age	Country	Zip Code	Health Condition
27	USA	14248	HIV
28	Canada	14207	HIV
26	USA	14246	Cancer
25	Canada	14249	Cancer
41	China	13053	Hepatitis
48	Japan	13074	Phthisis
45	India	13064	Asthma
42	India	13062	Heart Disease
33	USA	14242	Flu
37	Canada	14204	Flu
36	Canada	14205	Flu
35	USA	14248	Indigestion

Table 1: Micro data

Age	Country	Zip Code	Health Condition
<30	America	142∗∗	HIV
<30	America	142∗∗	HIV
<30	America	1424∗	Cancer
<30	America	1424∗	Cancer
>40	Asia	130∗∗	Hepatitis
>40	Asia	130∗∗	Phthisis
>40	Asia	130∗∗	Asthma
>40	Asia	130∗∗	Heart Disease
3∗	America	1424∗	Flu
3∗	America	142∗∗	Flu
3∗	America	142∗∗	Flu
3∗	America	1424∗	Indigestion

Table 2: a 2-anonymous view of Table 1

view corresponding to it. The sensitive attribute (Health Condition) is retained without change in this example.

In the literature of k-anonymity problem, there are two main models. One model is global recoding [5, 8, 16, 13] while the other is local recoding [1, 16]. Here, we assume that each attribute has a corresponding conceptual generalization hierarchy or taxonomy tree. A lower level domain in the hierarchy provides more details than a higher level domain. For example, Zip Code 14248 is a lower level domain and Zip Code 142∗∗ is a higher level domain. We assume such hierarchies for numerical attributes too. In particular, we have a hierarchical structure defined with $\{$value, interval, $*\}$, where value is the raw numerical data, interval is the range of the raw data and $*$ is a symbol representing any values. Generalization replaces lower level domain values with higher level domain values. For example, Age 27, 28 in the lower level can be replaced by the interval (27-28) in the higher level (See Table 2).

In recent years, numerous algorithms have been proposed for implementing k-anonymity via generalization and suppression. Samarati [13] presents an algorithm that exploits a binary search on the domain generalization hierarchy to find minimal k-anonymous table. Sun *et al.* [14] recently improve his algorithm by integrating the hash-based technique. Bayardo and Agrawal [3] presents an optimal algorithm that starts from a fully generalized table and specializes the data set in a minimal k-anonymous table, exploiting ad hoc pruning techniques. LeFevre *et al*. [8] describes an algorithm that uses a bottom-up technique and a priori computation. Fung *et al*. [5] present a top-down heuristic to make a table to be released k-anonymous. As to the theoretical results, Meyerson and Williams [11] and Aggarwal *et al*. [1, 2] proved the optimal k-anonymity is NP-hard (based on the number of cells and number of attributes that are generalized and suppressed) and describe approximation algorithms for optimal k-anonymity. Sun *et al*. [15] proved that k-anonymity problem is also NP-hard even in the restricted cases, which could imply the results in [1, 2, 11] as well.

Another method to achieve anonymity is through micro-aggregation [4]. Micro-aggregation is an Statistical Disclosure Control (SDC) technique consisting in the aggregation of individual data. It can be considered as an SDC sub-discipline devoted to the protection of individual data, also called micro-data. Micro-aggregation can be seen as a clustering problem with constraints on the size of the clusters. It is somehow related to other clustering problems (e.g. dimension reduction or minimum squares design of clusters). However, the main difference of the micro-aggregation problem is that it does not consider the number of clusters to generate or the number of dimensions to reduce, but only the minimum number of elements that are grouped in the same cluster. In this paper, we focus on the using generalization and suppression.

While focusing on identity disclosure, k-anonymity model fails to protect attribute disclosure [7]. Several models such as p-sensitive k-anonymity [19], l-diversity [10], (α, k)-anonymity [22] and t-closeness [9] were proposed in the literature in order to deal with the problem of k-anonymity. The work presented in this paper is highly inspired by [19]. The main contribution of [19] is to introduce the p-sensitive k-anonymity property, which requires, in addition to k-anonymity, that for each group of tuples with identical combination of quasi-identifier values, the number of distinct sensitive attributes values must be at least p. However, depending on the nature of the sensitive attributes, even p-sensitive property still permits the information to be disclosed. We identify in this paper situations when p-sensitive property is not enough for privacy protection and we propose two solutions to overcome this identified problem: (p, α)-sensitive and (p^+, α)-sensitive k-anonymity models and the heuristic algorithms to enforce these properties.

The rest of paper is organized as follows. We introduce some basic concepts and p-sensitive k-anonymity model in Section 2. Enhanced p-sensitive k-anonymity models are discussed in Section 3. Hardness results with respect to the enhanced models are discussed in Section 4. Algorithmic issues are presented in Section 5 and the experimental results are analyzed in Section 6. Finally, we conclude the paper in Section 7.

2 Problem Definitions

Let T be the initial micro data table and T' be the released micro data table. T' consists of a set of tuples over an attribute set. The attributes characterizing micro data are classified into the following three categories.

- *Identifier attributes* that can be used to identify a record such as Name and Medicare card.

- *Quasi-identifier (QI) attributes* that may be known by an intruder, such as Zip code and Age. QI attributes are presented in the released micro data table T' as well as in the initial micro data table T.

- *Sensitive attributes* that are assumed to be unknown to an intruder and need to be protected, such as Health Condition or ICD9Code [1]. Sensitive attributes are presented both in T and T'.

In what follows we assume that the identifier attributes have been removed and the quasi-identifier and sensitive attributes are usually kept in the released and initial micro data table. Another assumption is that the value for the sensitive attributes are not available from any external source. This assumption guarantees that an intruder can not use the sensitive attributes to increase the chances of disclosure. Unfortunately, an intruder may use record linkage techniques [21] between quasi-identifier attributes and external available information to glean the identity of individuals from the modified micro data. To avoid this possibility of privacy disclosure, one frequently used solution is to modify the initial micro data, more specifically the quasi-identifier attributes values, in order to enforce the k-anonymity property.

Definition 1. (Quasi-identifier) *A quasi-identifier (QI) is a minimal set Q of attributes in micro data table T that can be joined with external information to re-identify individual records (with sufficiently high probability).*

[1] available at http://www.icd9code.com/

Name	Age	Country	Zip Code
Rick	26	USA	14246
Hassen	45	India	13064
Rudy	25	Canada	14249
Yamazaki	48	Japan	13074

Table 3: External available information

Category ID	Sensitive attribute values	Sensitivity
One	HIV, Cancer	Top Secret
Two	Phthisis, Hepatitis	Secret
Three	Heart Disease, Asthma	Less Secret
Four	Flu, Indigestion	Non Secret

Table 4: Categories of Health Condition

Age	Country	Zip Code	Health Condition
<30	America	142**	HIV
<30	America	142**	HIV
<30	America	142**	Cancer
<30	America	142**	Cancer
>40	Asia	130**	Hepatitis
>40	Asia	130**	Phthisis
>40	Asia	130**	Asthma
>40	Asia	130**	Heart Disease
3*	America	142**	Flu
3*	America	142**	Flu
3*	America	142**	Flu
3*	America	142**	Indigestion

Table 5: 2-sensitive 4-anonymous Micro data

Definition 2. (k-anonymity) *The modified Micro data table T' is said to satisfy k-anonymity if and only if each combination of quasi-identifier attributes in T' occurs at least k times.*

A QI-group in the modified micro data T' is the set of all records in the table containing identical values for the QI attributes. There is no consensus in the literature over the term used to denote a QI-group. This term was not defined when k-anonymity was introduced [13, 17]. More recent papers use different terminologies such as equivalence class [22] and QI-cluster [18].

For example, let the set {Age, Country, Zip Code} be the quasi-identifier of Table 1. Table 2 is one 2-anonymous view of Table 1 since there are five QI-groups and the size of each QI-group is at least 2. So k-anonymity can ensure that even though an intruder knows a particular individual is in the k-anonymous micro data table T, s/he can not infer which record in T corresponds to the individual with a probability greater than $1/k$.

The k-anonymity property ensures protection against identity disclosure, i.e. the identification of an entity (person, institution). However, as we will show next, it does not protect the data against attribute disclosure, which occurs when the intruder finds something new about a target entity.

Still consider the modified 2-anonymous table (Table 2), where the set of quasi-identifier is composed of {Age, Country, Zip Code} and Health Condition is the sensitive attribute. As we discussed above, identity disclosure does not happen in this modified micro data. However, assuming that external information in Table 3 is available, attribute disclosure can take place. If the intruder knows that in the modified table (Table 2) the Age attribute was modified to '<30', he can deduce that both Rick and Rudy have Cancer, even he does not know which record, 3 or 4, corresponds to which person. This example shows that even if k-anonymity can well protect identity disclosure, sometimes it fails to protect against sensitive attribute disclosure.

To deal with this problem in privacy breach, the p-sensitive k-anonymity model was introduced in [19]. A similar privacy model, called l-diversity, is described in [10].

Definition 3. (p-sensitive k-anonymity) *The modified micro data table T' satisfies p-sensitive k-anonymity property if it satisfies k-anonymity, and for each QI-group in T', the number of distinct values for each sensitive attribute occurs at least p times within the same QI-group.*

Although the p-sensitive k-anonymity represents an important step beyond k-anonymity

ID	Age	Country	Zip Code	Health Condition	Category ID
1	<40	America	1424∗	HIV	One
4	<40	America	1424∗	Cancer	One
9	<40	America	1424∗	Flu	Four
12	<40	America	1424∗	Indigestion	Four
5	>40	Asia	130∗∗	Hepatitis	Two
6	>40	Asia	130∗∗	Phthisis	Two
7	>40	Asia	130∗∗	Asthma	Three
8	>40	Asia	130∗∗	Heart Disease	Three
2	<40	America	1420∗	HIV	One
3	<40	America	1420∗	Cancer	One
10	<40	America	1420∗	Flu	Four
11	<40	America	1420∗	Flu	Four

Table 6: $(2^+, 2)$-sensitive 4-anonymous Micro data

in protecting against attribute disclosure, it still has some shortcomings. Following through, we show that p-sensitive k-anonymity is insufficient to prevent *Similarity Attack*.

Similarity Attack: When the sensitive attribute values in a QI-group are distinct but similar sensitivity, an adversary can learn important information.

Sometimes, the domain of the sensitive attributes, especially the categorical ones, can be partitioned into categories according to the sensitivity of attributes. For example, in medical data sets Table 1, the Health Condition attribute can be classified into four categories (see Table 4). The different types of diseases are organized in a category domain. The attribute values are very specific, for example they can represent HIV or Cancer, which are both Top Secret information of the individuals. In the case that the initial micro data contains specific sensitive attributes like Health Condition, the data owner can be interested in protecting not only these most specific values, but also the category that the sensitive values belong to. For example, the information of a person affected with Top Secret needs to be protected, no matter whether it is HIV or Cancer. If we modify the micro data to satisfy p-sensitive k-anonymity property, it is possible that in a QI-group with p distinct sensitive attribute values, all of them belong to the same pre-defined category. For instance, the values {HIV, HIV, Cancer, Cancer} in one QI-group in Table 5 all belong to Top Secret category. To avoid such situations, we introduce our new enhanced p-sensitive k-anonymity models, which are aware of not only protecting specific sensitive values.

3 Enhanced P-Sensitive K-Anonymity Models

Let S be a categorical sensitive attribute we want to protect against attribute disclosure. First, we sort out the values of S according to their sensitivity, forming an ordered value domain D, and then partition the attribute domain into m-categories $(S_1, S_2, \cdots, S_m)$, such that $S = \cup_{i=1}^{m} S_i$, $S_i \cap S_j = \emptyset$ (for $i \neq j$) and S_l is more sensitive than the S_k (for $1 \leq l \leq k$). For example, Consider the Health Condition S={HIV, Cancer, Phthisis, Hepatitis, Heart Disease, Asthma, Flu, Indigestion} in Table 1, it has been partitioned into four categories according to the sensitivity of the diseases (Table 4), where S_1 (Top Secret) is the most sensitive and S_4 (Non Secret) is the least one.

Furthermore, in order to measure the distance between two categories (attributes) and the degree that sensitive attribute values contribute to one QI-group, we introduce the following ordinal metric system.

Let $D(S)$ denote a categorical domain of an attribute S and $|D(S)|$ be the total number of categories in domain $D(S)$. The normalized distance between two categories S_i and S_j of the attribute S with $S_i \leq S_j$ is:

$$d(S_i, S_j) = \frac{|S_l|S_i \leq S_l < S_j|}{|D(S)| - 1}$$

The distance between two sensitive attribute values is equal to the distance between the categories that they fall into. Moreover, we put an ordinal weight to each category to represent the degree that each specific sensitive attribute value in S contributes to S.

Let $D(S) = \{S_1, S_2, \cdots, S_k\}$ denote a partition of categorical domain of an attribute S and let $weight(S_i)$ denote the weight of category S_i. Then,

$$\begin{cases} weight(S_1) = 0, \\ weight(S_i) = \frac{i-1}{k-1}; \ 1 < i < k \\ weight(S_k) = 1, \end{cases} \tag{1}$$

Note that the weight of the specific sensitive value is equal to the weight of the category that the specific value belongs to. The weight of the QI-group is the total weight of each specific sensitive values that the QI-group contains.

We illustrate these concepts by taking Table 6 as an example. Given the partition of sensitive attribute values as shown in Table 4 and four corresponding values set A={Cancer, Phthisis, Asthma, Flu}. The distance between Cancer (S_1) and Flu (S_4) is 3/3=1, while the distance between Phthisis (S_2) and Asthma (S_3) is 1/3. According to (1), $weight(S_1) = 0$, $weight(S_2) = 1/3$ and $weight(Asthma) = 2/3$, $weight(Flu) = 1$, the total weight of A is 0+1/3+2/3+1=2.

Definition 4. ((p, α)-sensitive k-anonymity) *The modified micro data table T' satisfies (p, α)-sensitive k-anonymity property if it satisfies k-anonymity, and each QI-group has at least p **distinct sensitive attribute values** with its total weight at least α.*

Table 7 is a $(3, 1)$-sensitive 4-anonymous view of Table 1, since there are at least three different values in each QI-group and the least total weight of the QI-group is 1. We can easily see that the (p, α)-sensitive k-anonymity model can well protect sensitive information disclosure as well when compared with previous p-sensitive k-anonymity model.

Definition 5. ((p^+, α)-sensitive k-anonymity): T*he modified micro data table T' satisfies (p^+, α)-sensitive k-anonymity property if it satisfies k-anonymity, and each QI-group has at least p **distinct categories** of the sensitive attribute and its total weight is at least α.*

Table 6 is a $(2^+, 2)$-sensitive 4-anonymous view of Table 1. As you can see, for example, the records 1,4,9 and 12 belong to one QI-group in which the Health Condition is not that easy to be referred since they belong to two different categories with its total weight 2. Compared with previous 2-sensitive 4-anonymity model (See Table 5), our new model could overcome the shortcomings of previous models and significantly reduce the possibility of leaking privacy. Different from previous p-sensitive k-anonymity model which publishes original specific sensitive attributes, we publish the categories that the sensitive values belong to.

ID	Age	Country	Zip Code	Disease	weight	total
1	<40	America	142**	HIV	0	1
2	<40	America	142**	HIV	0	
3	<40	America	142**	Cancer	0	
9	<40	America	142**	Flu	1	
5	>40	Asia	130**	Hepatitis	1/3	2
6	>40	Asia	130**	Phthisis	1/3	
7	>40	Asia	130**	Asthma	2/3	
8	>40	Asia	130**	Obesity	2/3	
4	<40	America	14***	Cancer	0	3
10	<40	America	14***	Flu	1	
11	<40	America	14***	Flu	1	
12	<40	America	14***	Indigestion	1	

Table 7: (3, 1)-sensitive 4-anonymous Micro data

4 Hardness Results

Optimal p-sensitive k-anonymity problem is NP-hard as discussed in [19]. Now, we show that optimal (p, α) and (p^+, α)-sensitive k-anonymity problems are also NP-hard.

Theorem 1: *(p, α)-sensitive k-anonymity problem is NP-hard for a binary alphabet ($\sum = \{0, 1\}$).*

Proof: The proof is by transforming the problem of EDGE PARTITION INTO 4-CLIQUES [6] to the (p, α)-sensitive k-anonymity problem.

EDGE PARTITION INTO 4-CLIQUES: Given a simple graph $G = (V, E)$, with $|E| = 6m$ for some integer m, can the edges of G be partitioned into m edge-disjoint 4-cliques?

Given an instance of EDGE PARTITION INTO 4-CLIQUES. Set $p = 2$, $\alpha = 6$ and $k = 12$. For each vertex $v \in V$, construct a non-sensitive attribute. For each edge $e \in E$, where $e = (v_1, v_2)$, create a pair of records $r_{v1,v2}$ and $\tilde{r}_{v1,v2}$, where the two records have the attribute values of both v_1 and v_2 equal to 1 and all other non-sensitive attribute values equal to 0, but one record $r_{v1,v2}$ has the sensitive attribute equal to 1 and the other record $\tilde{r}_{v1,v2}$ has the sensitive attribute equal to 0.

We define the cost of the (2, 6)-sensitive 12-anonymity to be the number of suppressions applied in the data set. We show that the cost of the (2, 6)-sensitive 12-anonymity is at most $48m$ if and only if E can be partitioned into a collection of m edge-disjoint 4-cliques.

Suppose E can be partitioned into a collection of m disjoint 4-cliques. Consider a 4-clique C with vertices v_1, v_2, v_3 and v_4. If we suppress the attributes v_1, v_2, v_3 and v_4 in the 12 records corresponding to the edges in C, then a cluster of these 12 records are formed where each modified record has four $*$s. Note that the (p, α)-sensitive requirement can be satisfied as the frequency of the sensitive attribute value 1 is equal to 6. The cost of the (2, 6)-sensitive 12-anonymity is equal to $12 \times 4 \times m = 48m$.

Suppose the cost of the (2, 6)-sensitive 12-anonymity is at most $48m$. As G is a simple graph, any twelve records should have at least four attributes different. So, each record should have at least four $*$s in the solution of the (2, 6)-sensitive 12-anonymity. Then, the cost of the (2, 6)-sensitive 12-anonymity is at least $12 \times 4 \times m = 48m$. Combining with the proposition that the cost is at most $48m$, we obtain the cost is exactly equal to $48m$ and thus each record should have exactly four $*$s in the solution. Each cluster should have exactly 12 records (where six have sensitive value 1 and the other six have sensitive value

Algorithm: Local-recoding Algorithm

```
fully generalize all tuples such that all tuples are equal.
let $P$ be a set containing all these generalized tuples
$S \leftarrow \{P\}; O \leftarrow \emptyset$.
repeat
  $S' \leftarrow \emptyset$
  for all $P \in S$ so
    specialize all tuples in $P$ one level down in the generalization hierarchy
      such that a number of specialized child nodes are formed.
    unspecialize the nodes which do not satisfy $(p, \alpha)$-sensitive $k$-anonymity by
      moving the tuples back to the parent node.
    if the parent $P$ does not satisfy $(p, \alpha)$-sensitive $k$-anonymity then.
      unspecialize some tuples in the remaining child nodes so that
        the parent $P$ satisfies $(p, \alpha)$-sensitive $k$-anonymity
    for all non-empty branches $B$ of $P$, do $S' \leftarrow S' \cup \{B\}$
    $S \leftarrow S'$
    if $P$ is non-empty then $O \leftarrow O \cup \{P\}$
until $S = \emptyset$
return $O$.
```

0). Suppose the twelve modified records contain four $*$s in attributes v_1, v_2, v_3 and v_4, the records contain 0s in all other nonsensitive attributes. This corresponds to a 4-clique with vertices v_1, v_2, v_3 and v_4. Thus, we conclude that the solution corresponds to a partition into a collection of m edge-disjoint 4-cliques. ■

Corollary 1: *(p^+, α)-sensitive k-anonymity problem is NP-hard for a binary alphabet ($\sum = \{0, 1\}$).*

Distortion Ratio: The cost of recoding is given by the *Distortion Ratio* of the resulting data set and is defined as follows. Suppose the value of the attribute of a tuple (record) has not been generalized, there will be no distortion. However, if the value of the attribute of a tuple is generalized to a more general value in the taxonomy tree, there is a distortion of the attribute of the tuple. If the value is generalized more (i.e. the original value is updated to a value at the node of the taxonomy near to the root), the distortion will be greater. Thus, the distortion of this value is defined in terms of the height of the value generalized. For example, if the value has not been generalized, the height of the value generalized is equal to 0. If the value has been generalized one level up in the taxonomy, the height of the value generalized is equal to 1. Let $h_{i,j}$ be the height of the value generalized of attribute S_i of the tuple t_j. The distortion of the whole data set is equal to the sum of the distortions of all values in the generalized data set. That is, distortion=$\sum_{i,j} h_{i,j}$. *Distortion Ratio* is equal to the distortion of the generalized data set divided by the distortion of the fully generalized data set, where the fully generalized data set is the one with all values of the attributes are generalized to the root of the taxonomy tree.

These two new introduced models focus on different perspectives in protecting sensitive attributes disclosures. Although (p, α)-sensitive k-anonymity model still put the point on the specific values, it includes an ordinal metric system to measure how much the specific sensitive attribute values contribute to each QI-group. Furthermore, instead of focusing on the specific values of sensitive attributes, (p, α)-sensitive k-anonymity model cares more about the categories that the values belong to. Note that both models are effective in avoiding privacy breach, we could compare the *Distortion Ratio* of these two models in the experimental study section.

5 The Algorithms

5.1 Global Recoding

We extend an existing global-recoding based algorithm called Incognito [8] for both (p, α)-sensitive k-anonymity and (p^+, α)-sensitive k-anonymity models. Incognito algorithm is an optimal global-recoding algorithm for the k-anonymity problem. It has also been used in [19] for the p-sensitive k-anonymity problem. [8] and [19] make use of *monotonicity property* in searching the solution space. The searches can be made efficient if a stopping condition is satisfied. The stopping condition is that, if table T' is satisfied with the privacy requirements, then every generalization of T' is also satisfied with the privacy requirement.

The algorithms for generating (p^+, α)-and (p, α)-sensitive k-anonymous tables tables, are similar to [8, 19]. The difference is in the testing criteria of each candidate in the solution space. [8] tests for the k-anonymity property and [19] tests the p-sensitive k-anonymity. Here, we check the (p, α)-sensitive k-anonymity and (p^+, α)-sensitive k-anonymity properties.

5.2 Local Recoding

The extended Incognito algorithm is an exhaustive global recoding algorithm which is not scalable and may generate excessive distortions to the data set. Here we propose a scalable local-recoding algorithm. In this section, we present a top-down approach to tackle the problem. The idea of the algorithm is to first generalize all tuples completely so that, initially, all tuples are generalized into one QI-group. Then, tuples are specialized in iterations. During the specialization, we must maintain (p, α)-sensitive k-anonymity and (p^+, α)-sensitive k-anonymity. The process continues until we cannot specialize the tuples anymore. The code for (p, α)-sensitive k-anonymity is described in Algorithm 1. The algorithm for (p^+, α)-sensitive k-anonymity is similar to Algorithm 1, the only difference is to test (p^+, α)-sensitive k-anonymity property in Line 9 and 11. For ease of illustration, we present how the algorithm works for (p, α)-sensitive k-anonymity for a quasi-identifier of size 1.

Gender	Zipcode	Disease
Male	4351	HIV
Male	4351	Flu
Female	4351	HIV
Female	4352	Flu

No.	Zipcode	Disease
1	4351	HIV
2	4351	Flu
3	4351	HIV
4	4352	Flu

No.	Zipcode	Disease
1	4351	HIV
2	4351	Flu
3	435*	HIV
4	435*	Flu

Table 8: Left: Sample Data; Middle: Original Projected Table; Right: Generalized Table

Let us illustrate it with an example in Table 8 (Left). Suppose the QI contains Zipcode only. Because there are only two sensitive values, so we assume that $\alpha, p, k = 2$. Initially, we generalize all four tuples completely to a most generalized value Zipcode=**** (Figure 1(a)). Then, we specialize each tuple one level down in the generalization hierarchy. We obtain the branch with Zipcode = 4*** in Figure 1(b). In the next iterations, we obtain the branch with Zipcode = 43** and the branch with Zipcode = 435* in Figure 1(c) and (d),

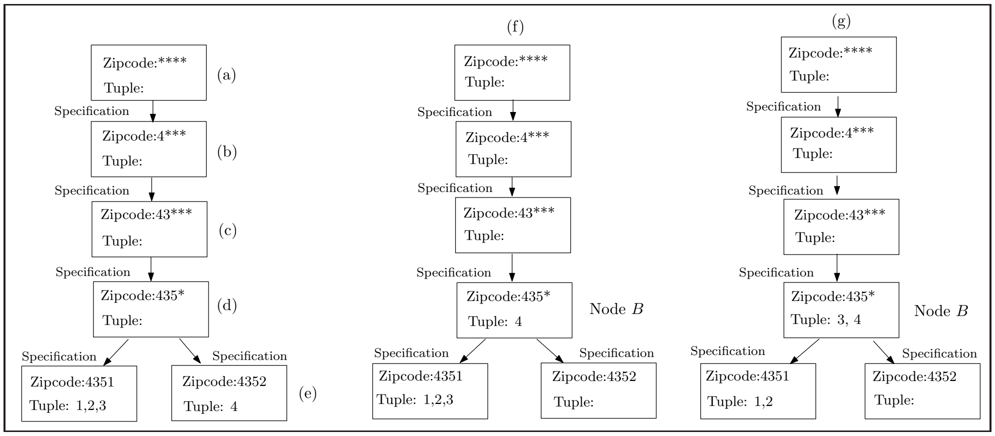

Figure 1: Algorithm for $|QI|$=1

respectively. Next, we can further specialize the tuples into the two branches as shown Figure 1(e). Hence the specialization processing can be seen as the growth of a tree.

If each leaf node satisfies (p, α)-sensitive k-anonymity, then the specialization will be successful. However, we may encounter some problematic leaf nodes that do not satisfy (p, α)-sensitive k-anonymity. Then, all tuples in such leaf nodes will be pushed upwards in the generalization hierarchy. In other words, those tuples cannot be specialized in this process. They should be kept unspecialized in the parent node. For example, in Figure 1(e), the leaf node with Zipcode = 4352 contains only one tuple, which violates (p, α)-sensitive k-anonymity. Thus, we have to move this tuple back to the parent node with Zipcode = 435*. See Figure 1(f).

After the previous step, we move all tuples in problematic leaf nodes to the parent node. However, if the collected tuples in the parent node do not satisfy (p, α)-sensitive k-anonymity, we should further move some tuples from other leaf nodes L to the parent node so that the parent node can satisfy (p, α)-sensitive k-anonymity while L also maintain the (p, α)-sensitive k-anonymity. For instance, in Figure 1(f), the parent node with Zipcode = 435* violates (p, α)-sensitive k-anonymity. Thus, we should move one tuples upwards in the node B with Zipcode = 4351 (which satisfies (p, α)-sensitive k-anonymity). In this example, we move tuple 3 upwards to the parent node so that both the parent node and the node B satisfy the (p, α)-sensitive k-anonymity.

Finally, in Figure 1(g), we obtain a data set where the Zipcode of tuples 3 and 4 are generalized to 435* and the Zipcode of tuples 1 and 2 remains 4351. So the final allocation of tuples in Figure 1(g) is the final distribution of tuples after the specialization. The results can be found at the right in Table 8.

6 Experimental Study

The main goals of the experiments are to study the *Similarity Attack* on real data and to investigate the performance implications of the new introduced (p, α)-sensitive k-anonymity and (p^{+}, α)-sensitive k-anonymity approaches in terms of distortion ratio and execution time.

In our experiment, we adopted the publicly available data set, Adult Database, at the UC

Attribute	Type	Distinct values	Height
Age	Numeric	74	5
Workclass	Categorical	8	3
Education	Categorical	16	4
Country	Categorical	41	3
Marital Status	Categorical	7	3
Race	Categorical	5	3
Gender	Categorical	2	2
Health Condition	Sensitive	8	–

Table 9: Features of Quasi-identifier with Sensitive Attribute

Irvine Machine Learning Repository [12], which has become the benchmark of this field and was adopted by [8, 10, 5]. We used a configuration similar to [8, 10]. We eliminated the records with unknown values. The resulting data set contains 45222 tuples. Seven of the attributes were chosen as the quasi-identifier. We add a column with sensitive values called "Health Condition" consisting of {HIV, Cancer, Phthisis, Hepatitis, Obesity, Asthma, Flu, Indigestion} to the extracted data and randomly assign one sensitive value to each record of the extracted data by. The random technique works in the following way. First, assign a number to each sensitive attribute, i.e., {1:HIV, 2:Cancer, 3:Phthisis, 4:Hepatitis, 5:Obesity, 6:Asthma, 7:Flu, 8:Indigestion}. Second, for each tuple (record), generate a random number from 1-8. Then, assign the corresponding sensitive attribute value to the tuple. For example, for the first tuple in the data set, if the random number is 5, then this record has the sensitive value "Obesity". Table 9 provides a brief description of the data including the attributes we used, the type of each attribute data, the number of distinct values for each attribute, and the height of the generalization hierarchy for each attribute. The implementation of Incognito is available at `http://www.cs.cornell.edu/database/privacy/code/l-diversity/incognito-ldiversity.tgz` and we modified this implementation so that it produces (p, α)- and (p^+, α)-sensitive k-anonymous tables as well. All the experiments are run under Windows XP on a machine with 2.0GHz Pentium 4 processor and 1GB RAM. The algorithms were implemented in Java.

Similarity Attack: We use the first 7 attributes in Table 9 as the quasi-identifier and treat Health Condition as the sensitive attribute. We divide the 8 values of the Health Condition attribute into four pre-defined equal-size categories, based on the confidentiality of the values (See Table 4). Any QI-group that has all values falling in one category is viewed as vulnerable to the similarity attack. We use p-sensitive k-anonymity algorithm [19] to generate all p-sensitive k-anonymous tables. In total, there are 21 minimal tables and 13 of them suffers from the Similarity attack. In one table, a total of 916 records can be inferred about their sensitive value class. We also use the (p, α)- and $(p,^+, \alpha)$-sensitive k-anonymity algorithm to generate all 30 and 28 minimal tables, and found that only 7 and 3 of which are vulnerable to the similarity attack, respectively $(p = 2, k = 4, \alpha = 2)$.

Efficiency: We compare the efficiency of the three privacy measures: (1) p-sensitive k-anonymity; (2) (p, α)-sensitive k-anonymity $(\alpha = 2)$; (3) (p^+, α)-sensitive k-anonymity $(\alpha = 2)$. Results of efficiency experiments are shown in Fig. 2. Fig. 2(a) shows the running times with fixed $p = 4, k = 4, \alpha = 2$ and varied quasi-identifier size s, where $2 \leq s \leq 7$. A quasi-identifier of size s consists of the first s attributes listed in Table 9. Fig. 2(b) shows the running times of the three privacy measures with the same quasi-identifier but with different parameters for p and α. As shown in the figures, p-sensitive k-anonymity runs faster

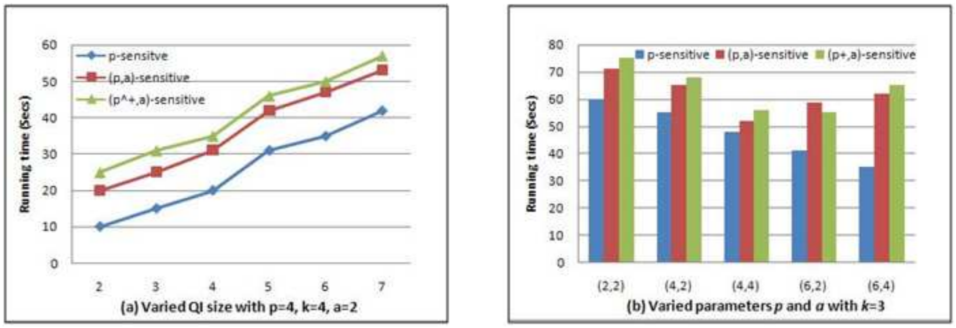

Figure 2: Efficiency of Three Privacy Measures

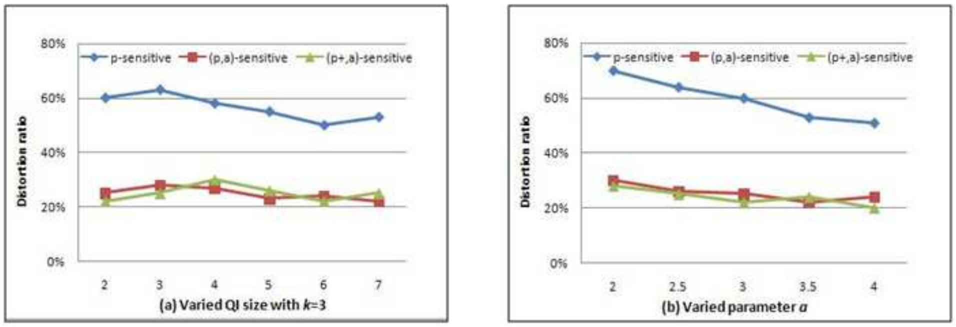

Figure 3: Distortion Ratio of Three Privacy Measures

than (p, α)-and (p^{+}, α)-sensitive k-anonymity; the difference gets larger when α increases.

Distortion ratio: Results of distortion ratio are shown in Fig. 3. From Fig. 3(a), it is easy to see that the distortion ratio increases with the quasi-identifier size. This is because when the quasi-identifier contains more attributes, there is more chance that the quasi-identifier of two tuples are different. In other words, there is more chance that the tuples will be generalized. Thus, the distortion ratio is greater. On average, the distortion ratio of p-sensitive k-anonymity model results in almost three times bigger than that of (p, α)-and (p^{+}, α)-sensitive k-anonymity models. In Fig. 3(b), when α increases, the distortion ratio decreases. Intuitively, if α is greater, there is less requirement of metric α, yielding fewer operations of generalization of the values in the data set. Thus, the distortion ratio is smaller.

7 Conclusion

p-sensitive k-anonymity is a novel property that, when satisfied by micro data sets, can help increase the privacy of the respondents whose data is being used. However, as shown in the paper, to some extent, this property is not enough for protecting sensitive attributes. In this paper, we proposed two enhanced p-sensitive k-anonymity models against *Similarity Attack*, namely (p, α)-and (p^{+}, α)-sensitive k-anonymity models. Our experimental results show that our proposed models could significantly reduce the possibility of *Similarity Attack* and incur less distortion ratio compared with previous p-sensitive k-anonymity model.

Acknowledgements

Thanks for anonymous referees for their valuable suggestion on improving the paper. This research was funded by Australian Research Council (ARC) grant DP0774450 and DP0663414.

References

[1] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas and A. Zhu. Anonymizing tables. *In Proc. of the 10th International Conference on Database Theory (ICDT05)*, pp. 246-258, Edinburgh, Scotland.

[2] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, A. Zhu. Approximation algorithms for k-anonymity. *Journal of Privacy Technology*, paper number 20051120001.

[3] R. Bayardo and R. Agrawal. Data privacy through optimal k-anonymity. *In Proceedings of the 21st International Conference on Data Engineering (ICDE)*, 2005.

[4] J. Domingo-Ferrer and V. Torra, Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Mining and Knowledge Discovery. v11. 195-212.

[5] B. Fung, K. Wang, P. Yu. Top-down specialization for information and privacy preservation. *In Proc. of the 21st International Conference on Data Engineering (ICDE05)*, Tokyo, Japan.

[6] M. R. Garey, D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. San Francisco. Freeman, 1979.

[7] D. Lambert. Measure of disclosure risk and harm. *Journal of Official Statistics*, vol 9, 1993, pp. 313-331.

[8] K. LeFevre, D. DeWitt and R. Ramakrishnan. Incognito: Efficient Full-Domain k-Anonymity. *In ACM SIGMOD International Conference on Management of Data*, June 2005.

[9] N. Li, T. Li, S. Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. *ICDE 2007*: 106-115

[10] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-Diversity: Privacy beyond k-anonymity. In ICDE, 2006.

[11] A. Meyerson and R. Williams. On the complexity of optimal k-anonymity. *In Proc. of the 23rd ACM-SIGMOD-SIGACT-SIGART Symposium on the Principles of Database Systems*, pp. 223-228, Paris, France, 2004.

[12] D. J. Newman, S. Hettich, C. L. Blake and C. J. Merz. UCI Repository of Machine Learning Databases, *available at www.ics.uci.edu/-mlearn/MLRepository.html*, University of Califonia, Irvine, 1998.

[13] P. Samarati. Protecting respondents' identities in micro data release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010-1027. 2001

[14] X. Sun, M. Li, H. Wang and A. Plank. An efficient hash-based algorithm for minimal k-anonymity problem. *in Thirty-First Australasian Computer Science Conference (ACSC2008)*, CRPIT vol 74, pp: 101-106, Wollongong, Australia.

[15] X. Sun, H. Wang and J. Li. On the complexity of restricted k-anonymity problem. *The 10th Asia Pacific Web Conference(APWEB2008)*, LNCS 4976, pp: 287-296, Shenyang, China.

[16] L. Sweeney. Achieving k-Anonymity Privacy Protection Using Generalization and Suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based System*, 10(5) pp. 571-588, 2002.

[17] L. Sweeney. k-Anonymity: A Model for Protecting Privacy. *International Journal on Uncertainty Fuzziness Knowledge-based Systems*, 10(5), pp 557-570, 2002

[18] T. M. Truta, A. Campan and P. Meyer. Generating Micro data with p-sensitive k-anonymity Property. *SDM 2007*: 124-141

[19] T. M. Truta and V. Bindu, Privacy Protection: p-sensitive k-anonymity property *International Workshop of Privacy Data Management (PDM2006), In Conjunction with 22th International Conference of Data Engineering (ICDE)*, Atlanta, 2006.

[20] K. Wang, P. S. Yu, and S. Chakraborty.: Bottom-up Generalization: A Data Mining Solution to Privacy Protection. *The fourth IEEE International Conference on Data Mining (ICDM2004)* 249-256.

[21] W. E. Winkler. Advanced Methods for Record Linkage, *Proceedings of the Section on Survey Research Methods, American Statistical Society*, 467-472

[22] R. Wong, J. Li, A. Fu, K. Wang. (α, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. *KDD 2006*: 754-759.