Privacy-Preserving Distributed k-Anonymity*

Wei Jiang and Chris Clifton

Department of Computer Science, Purdue University,
West Lafayette, IN 47907
{wjiang, clifton}@cs.purdue.edu
http://www.cs.purdue.edu/people/wjiang
http://www.cs.purdue.edu/people/clifton

Abstract. k-anonymity provides a measure of privacy protection by preventing re-identification of data to fewer than a group of k data items. While algorithms exist for producing k-anonymous data, the model has been that of a single source wanting to publish data. This paper presents a k-anonymity protocol when the data is vertically partitioned between sites. A key contribution is a proof that the protocol preserves k-anonymity between the sites: While one site may have individually identifiable data, it learns nothing that violates k-anonymity with respect to the data at the other site. This is a fundamentally different distributed privacy definition than that of Secure Multiparty Computation, and it provides a better match with both ethical and legal views of privacy.

Keywords: k-anonymity, privacy, security.

1 Introduction

Privacy is an important concept in our society, and has become very vulnerable in these technologically advanced times. Legislation has been proposed to protect individual privacy; a key component is the protection of *individually identifiable data*. Many techniques have been proposed to protect privacy, such as data perturbation [1], data swapping [2], query restriction [3], secure multiparty computation (SMC) [4,5,6], etc. One challenge is relating such techniques to a privacy definition that meets legal and societal norms. Anonymous data are generally considered to be exempt from privacy rules – but what does it mean for data to be anonymous? Census agencies, which have long dealt with private data, have generally found that as long as data are aggregated over a group of individuals, release does not violate privacy. k-anonymity provides a formal way of generalizing this concept. As stated in [7,8], a data record is k-anonymous if and only if it is indistinguishable in its identifying information from at least k specific records or entities. The key step in making data anonymous is to generalize a specific value. For example, the ages 18 and 21 could be generalized to

* This material is based upon work supported by the National Science Foundation under Grant No. 0428168.

S. Jajodia and D. Wijesekera (Eds.): Data and Applications Security 2005, LNCS 3654, pp. 166–177, 2005.

an interval [16..25]. Details of the concept of k-anonymity and ways to generate k-anonymous data are provided in Section 2.

Generalized data can be beneficial in many situations. For instance, a car insurance company may want to build a model to estimate claims for use in pricing policies for new customers. To build this model, the company may wish to use state-wide driver's license records. Such records, even with name and ID numbers removed, are likely to contain sufficient information to link to an individual. However, by generalizing data (e.g., replacing a birth date with an age range [26..30]), it is possible to prevent linking a record to an individual. The generalized age range is likely to be sufficient for building the claim estimation model. Similar applications exist in many areas: medical research, education studies, targeted marketing, etc.

Due to vast improvements in networking and rapid increase of storage capacity, the full data about an individual are typically partitioned into several sub-data sets (credit history, medical records, earnings, ...), each stored at an independent site.[1] The distributed setting is likely to remain, partially because of performance and accessibility, but more importantly because of autonomy of the independent sites. This autonomy provides a measure of protection for the individual data. For instance, if two attributes in combination reveal private information (e.g., airline and train travel records indicating likely attendance at political rallies), but the attributes are stored at different sites, a lack of cooperation between the sites ensures that neither is able to violate privacy.

In this paper, data are assumed to be vertically partitioned and stored at two sites, and the original data could be reconstructed by a one-to-one join on a common key. The goal is to build a k-anonymous join of the datasets, so that the join key and any other candidate keys in the joined dataset are k-anonymized to prevent re-identification.

1.1 What Is a Privacy-Preserving Distributed Protocol?

A key question in this problem is the definition of privacy preservation. Simply stating that the result is k-anonymous is not enough, as this does not ensure that the participating sites do not violate privacy. However, since the sites already have individually identifiable information, we cannot fully extend the k-anonymity measure to them. We now give an informal definition for privacy preservation; the paper will then present an algorithm and show formally that it does not violate k-anonymity in the sense of the following definition.

Definition 1. *Let T_i be the input of party i, $\prod_i(f)$ be the party i's execution image of the protocol f, r be the result computed by f, and P be a set of privacy constraints. f is privacy-preserving if every inference induced from $< T_i, \prod_i(f), r >$ that violates any privacy constraint in P could also be induced from $< T_i >$.*

[1] In the context of this paper, assume data are represented by a relational table, where each row indicates an individual data record and each column represents an attribute of data records.

This definition has much in common with that of Secure Multiparty Computation (SMC) [9]. Both talk about a party's view during execution of a protocol, and what can be inferred from that view. The key distinction is the concept of privacy (and privacy constraints) versus security. An SMC protocol must reveal nothing except the final result, and what can be inferred from one's own input and the result. Definition 1 is weaker (giving greater flexibility): It allows inferences from the protocol that go beyond what can be inferred from the result, provided that such inferences do not violate the privacy constraints.

A more subtle distinction is that Definition 1 is also *stronger* than SMC. The above definition requires that the inferences from the result r and from one's own input combined with the result (and the protocol execution) do not violate the privacy constraints. The SMC definitions do not account for this.

For example, a privacy-preserving classification scheme meeting SMC definitions [10,11,12,13] ensures that nothing is disclosed but the resulting model. Assume that Party A holds input attributes, and B holds the (private) class attribute: B has committed to ensuring that the class is not revealed for the individuals that have given it data. An SMC protocol can generate a classifier without revealing the class of the individuals to A. Moreover, the classifier need not inherently violate privacy: A properly pruned decision tree, for example, will only contain paths corresponding to several data values. A, however, can use its input along with the classifier to learn (with high probability) the class values held by B. This clearly violates the commitment B has made, even if the protocol meets SMC definitions. More discussion of this specific problem can be found in [14].

Generally speaking, if the set of privacy constraints P can be easily incorporated into the functionality computed by a SMC protocol, a SMC protocol also preserves privacy. However, there is no obvious general framework that easily and correctly incorporates privacy constraints into part of the functionality computed by a SMC protocol.

This paper presents a privacy-preserving two-party protocol that generates k-anonymous data from two vertically partitioned sources such that the protocol does not violate k-anonymity of either site's data. While one site may already hold individually identifiable data, we show that the protocol prevents either site from linking its own individually identifiable data to specific values from the other site, except as permitted under k-anonymity. (This privacy constraint will be formally defined in Section 3.) Interestingly, one of distinctive characteristics of the proposed protocol is that it is not secure by SMC definitions; parties may learn more than they can infer from their own data and the final k-anonymous datset. Nevertheless, it preserves the privacy constraint.

The rest of the paper is organized as the following: Section 2 introduces the fundamental concepts of k-anonymity. Section 3 presents a generic two-party protocol, with proof of its correctness and privacy-preservation property. The paper concludes with some insights gained from the protocol and future research directions on achieving k-anonymity in a distributed environment.

2 Background

We now give key background on k-anonymity, including definitions, a single-site algorithm, and a relevant theorem, from [7,15,16]. The following notations are crucial for understanding the rest of the paper:

- Quasi-Identifier (QI): a set of attributes that can be used with certain external information to identify a specific individual.
- T, T[QI]: T is the original dataset represented in a relational form, T[QI] is the projection of T to the set of attributes contained in QI.
- T_k[QI]: k-anonymous data generated from T with respect to the attributes in the Quasi-Identifier QI.

Definition 2. *T_k[QI] satisfies k-anonymity if and only if each record in it appears at least k times.*

Let T be Table 1, T_k be Table 2 and QI = {AREA, POSITION, SALARY}. According to Definition 2, T_k[QI] satisfies 3-anonymity.

Several algorithms have been proposed to generate k-anonymous data [17,8,18]. Datafly [8,18] is a simple and effective algorithm, so for demonstration of our protocol, Datafly is used to make local data k-anonymous. Algorithm 1 presents several key steps in Datafly (detailed explanations regarding this algorithm can be found in [8]). The main step in most k-anonymity protocols

Algorithm 1. Key Steps in Datafly

Require: T, QI$[A_1, \ldots, A_m]$, k, Hierarchies VGHs Assume $k \leq |T|$

```
freq ← a frequency list contains distinct sequences of values of T[QI] along with the number of occurrences of each sequence.
while (sequences ∈ freq occurring less than k times that count for more than k tuples) do
    A_i ∈ QI having the most number of distinct values
    freq ← generalize the values of A_i ∈ freq
end while
freq ← suppress sequences in freq occurring less than k times
freq ← enforce k requirement on suppressed tuples in freq
T_k[QI] ← construct table from freq
return T_k[QI]
```

is to substitute a specific value with a more general value. For instance, Figure 1(a) contains a value generalization hierarchy (VGH) for attribute AREA, in which Database Systems is a more general value than Data Mining. Similarly, Figure 1(b) and Figure 1(c) present VGHs of attributes POSITION and SALARY contained in QI. Continuing from the previous example, T_k[QI] satisfies 3-anonymity. According to the three VGHs and the original data represented by T, it is easily verified that Datafly can generate T_k[QI] by generalizing the data on SALARY, then AREA, then SALARY again. Next, we present a useful theorem about k-anonymity.

Table 1. Original Dataset Before Partitioning

ID	AREA	POSITION	SALARY	SSN
1	Data Mining	Associate Professor	$90,000	708-79-1698
2	Intrusion Detection	Assistant Professor	$91,000	606-67-6789
3	Data Warehousing	Associate Professor	$95,000	626-23-1459
4	Intrusion Detection	Assistant Professor	$78,000	373-55-7788
5	Digital Forensics	Professor	$150,000	626-87-6503
6	Distributed Systems	Research Assistant	$15,000	708-66-1552
7	Handhold Systems	Research Assistant	$17,000	810-74-1079
8	Handhold Systems	Research Assistant	$15,500	606-37-7706
9	Query Processing	Associate Professor	$100,000	373-79-1698
10	Digital Forensics	Assistant Professor	$78,000	999-03-7892
11	Digital Forensics	Professor	$135,000	708-90-1976
12	Intrusion Detection	Professor	$145,000	606-17-6512

Table 2. Generalized Data with $k = 3$

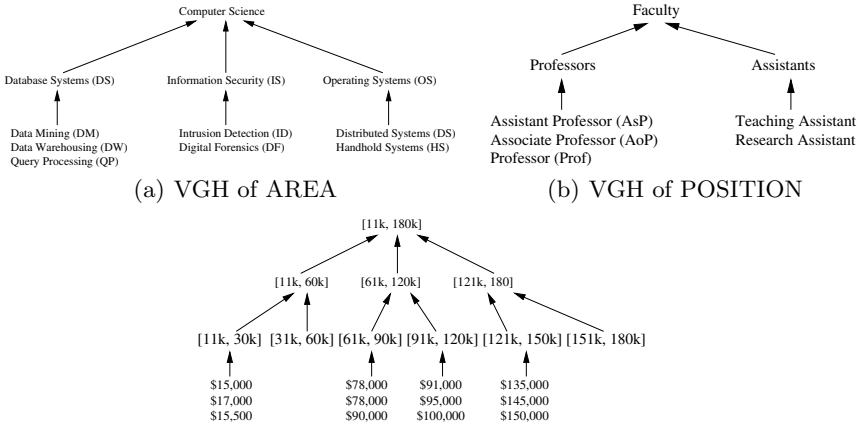
ID	AREA	POSITION	SALARY	SSN
1	Database Systems	Associate Professor	[61k, 120k]	708-79-1698
2	Information Security	Assistant Professor	[61k, 120k]	606-67-6789
3	Database Systems	Associate Professor	[61k, 120k]	626-23-1459
4	Information Security	Assistant Professor	[61k, 120k]	373-55-7788
5	Information Security	Professor	[121k, 180k]	626-87-6503
6	Operating Systems	Research Assistant	[11k, 30k]	708-66-1552
7	Operating Systems	Research Assistant	[11k, 30k]	810-74-1079
8	Operation Systems	Research Assistant	[11k, 30k]	606-37-7706
9	Database Systems	Associate Professor	[61k, 120k]	373-79-1698
10	Information Security	Assistant Professor	[61k, 120k]	999-03-7892
11	Information Security	Professor	[121k, 180k]	708-90-1976
12	Information Security	Professor	[121k, 180k]	606-17-6512

Theorem 1. *If T_k[QI] is k-anonymous, then T_k[QI'] is also k-anonymous, where QI' $\subseteq$ QI [8].*

Proof. Assume T_k[QI] is being k-anonymous and T_k[QI'] does not satisfy k-anonymity. Then there exists a record t(QI') that appears in T_k[QI'] less than k times. It is trivial to observe that t(QI) also appears less than k times in T_k[QI]. That contradicts the assumption. Therefore, if T_k[QI] satisfies k-anonymity, so does T_k[QI']. □

3 The Protocol: DPP$_2$GA

Before presenting the protocol, we present an alternative view of k-anonymity. Define T_k to be the k-anonymous data computed from T. Let $x \triangleright y$ denote that x is directly generalized from y. E.g., in Table 2 the Salary for ID 1: [61k, 120k] $\triangleright$ $90,000.



(a) VGH of AREA (b) VGH of POSITION

(c) VGH of SALARY

Fig. 1. Value Generalization Hierarchies

Theorem 2. *T_k achieved through generalization satisfies k-anonymity if and only if $\forall t' \in T_k, Prob[t' \triangleright t \in T] \leq \frac{1}{k}$.*

Proof. $\Rightarrow$: Given generalized values t', if $t' \in T_k$ then there is a set S of identical $t'_i \in T_k$ s.t. $|S| \geq k$ and $t' = t'_i$ (by the definition of k-anonymity). Each $t'_i \in S \triangleright t \in T$. Since we cannot distinguish between the t'_is, the probability that we have a particular $t'_i = \frac{1}{S} \leq \frac{1}{k}$. Thus the probability that t' is generalized from a particular t_i is $Prob[t' \triangleright t_i] = Prob[t' = t'_i] \leq \frac{1}{k}$.

$\Leftarrow$: Let $Prob[t' \triangleright t \in T] \leq \frac{1}{k}$, and t' be the record with the highest such probability for a generalization from t. Since the generalization is done according to a hierarchy, t must generalize to a (uniquely determined) single node in each hierarchy. This defines the only allowed values for t'. Thus all $t'_i \in T_k$ have $Prob[t'_i \triangleright t] = 0$ or $Prob[t'_i \triangleright t] = Prob[t' \triangleright t] \leq \frac{1}{k}$. Since t must uniquely generalize to one of the t'_i, the sum of probabilities must be 1. Thus there must be at least k $t'_i \in T_k$ that are identical to t', so k-anonymity holds for t'. □

From Theorem 2, the privacy constraint P in our application domain can be formally defined as: inferences from $< T_i, \prod_i(f), T_k >$ do not enable party i to conclude $\exists t' \in T_k$ (or a t' seen in $\prod_i(f)$) such that $Prob[t' \triangleright t \in T] > \frac{1}{k}$. Informally, $< T_i, \prod_i(f), T_k >$ does not make T_k less k-anonymous. We will revisit this privacy constraint when proving that the proposed protocol is privacy-preserving.

Since the protocol can utilize any k-anonymity algorithm to compute locally anonymous data, we call the proposed approach Distributed Privacy-Preserving two-Party Generic Anonymizer (DPP$_2$GA). The protocol is presented in Section 3.1, Section 3.2 proves the correctness of the protocol and Section 3.3 proves the protocol satisfies the k-anonymity privacy constraint.

3.1 DPP$_2$GA

The protocol is executed between two parties: P1 and P2. Let T refer to Table 1 and QI = {AREA, POSITION, SALARY}. T is vertically partitioned into $T1 \equiv T$[ID, AREA, POSITION] and $T2 \equiv T$[ID, SALARY, SSN] stored at P1 and P2 respectively. Also, assume P1 and P2 are semi-honest in that they follow the execution of the protocol but may later use the information seen to try to violate privacy. (Discussion of the privacy properties under stronger adversarial models omitted due to space constraints.)

The key idea of the protocol is based on Theorem 1. Initially, each party Pi (i = 1 or 2) makes his data k-anonymous locally (for simplicity, Datafly is used for illustration). Based on this locally k-anonymous data, a set γ^i is produced containing IDs partitioned into subsets. Let $\gamma^i[p]$ indicates the p^{th} subset in γ^i, then all records Pi whose keys are contained in $\gamma^i[p]$ have the same value with respect to QI. For any γ^i, the following properties hold:

- $\gamma^i[p] \cap \gamma^i[q] = \emptyset$, for any $1 \leq p, q \leq |\gamma^i|$ and $p \neq q$
- $\bigcup_p \gamma^i[p]$ is the same across all γ^is

Note that although each element $\gamma^i[p]$ in γ^i contains record keys, it does make sense to say that $\gamma^i[p]$ contains a subset of records or data tuples because each key is related to a single tuple. Define Ti_{γ^i} be the generalized data at Pi based on which γ^i is computed. For example, refer to Table 3, the columns [AREA^p, POSITION^q] indicate the generalized data of $T1$[AREA, POSITION], where p+q indicates the number of times $T1$[AREA, POSITION] has been generalized (by Datafly). Also, the last generalization of $T1$[AREA, POSITION] was performed on the attribute whose superscript was incremented comparing to its previous value. $T2$[SALARY] can be interpreted similarly. According to Table 3, we have:

$$\gamma_1^1 = \{\{1, 3, 9\}, \{2, 4, 10\}, \{5, 11, 12\}, \{6, 7, 8\}\}$$
$$\gamma_1^2 = \{\{1, 4, 10\}, \{2, 3, 9\}, \{5, 11, 12\}, \{6, 7, 8\}\}$$

Table 3. P1 and P2 's Generalized Data (left and right respectively)

ID	AREA^1	POSITION^0	AREA^1	POSITION^1
1	DB	AoP	DB	Professors
2	IS	AsP	IS	Professors
3	DB	AoP	DB	Professors
4	IS	AsP	IS	Professors
5	IS	Prof	IS	Professors
6	OS	RA	OS	Assistant
7	OS	RA	OS	Assistant
8	OS	RA	OS	Assistant
9	DB	AoP	DB	Professors
10	IS	AsP	IS	Professors
11	IS	Prof	IS	Professors
12	IS	Prof	IS	Professors

ID	SALARY^1	SALARY^2
1	[61k, 90k]	[61k, 120k]
2	[91k, 120k]	[61k, 120k]
3	[91k, 120k]	[61k, 120k]
4	[61k, 90k]	[61k, 120k]
5	[121k, 150k]	[121k, 180k]
6	[11k, 30k]	[11k, 30k]
7	[11k, 30k]	[11k, 30k]
8	[11k, 30k]	[11k, 30k]
9	[91k, 120k]	[61k, 120k]
10	[61k, 90k]	[61k, 120k]
11	[121k, 150k]	[121k, 180k]
12	[121k, 150k]	[121k, 180k]

The two parties then compare γ_1^1 and γ_1^2. If they are *equal* (this notion of equality will be defined shortly), joining data $T1_{\gamma_1^1}$ and $T2_{\gamma_1^2}$ creates globally k-anonymous data. If γ_1^1 and γ_1^2 are not equal, each party generalizes his local data one step further and creates a new γ^i. Repeat the above steps until the two parties find a pair of equal γ^is. Let's define the notion of equality between any two γ^is.

Definition 3. *If $\gamma_\alpha^i \equiv \gamma_\beta^j$, then there are no p, q such that $0 < |\gamma_\alpha^i[p] \cap \gamma_\beta^j[q]| < k$.*

According to the above definition, $\gamma_1^1 \neq \gamma_1^2$ because $|\{1,3,9\} \in \gamma_1^1 \cap \{2,3,9\} \in \gamma_1^2| = 2 < k$ (where $k = 3$). Thus, P1 and P2 generalize their data one step further and compute two new γ^is:

$$\gamma_2^1 = \{\{1,3,9\},\{2,4,5,10,11,12\},\{6,7,8\}\}$$
$$\gamma_2^2 = \{\{1,2,3,4,9,10\},\{5,11,12\},\{6,7,8\}\}$$

Since $\gamma_2^1 \equiv \gamma_2^2$, the join of $T1_{\gamma_2^1}$ (columns [AREA^1, POSITION^1] in Table 3) and $T2_{\gamma_2^2}$ (column [SALARY^2] in Table 3) satisfies 3-anonymity.

Due to privacy issues, the comparison between γ^is are not performed directly. Instead, P1 encrypts γ^1 and sends $E_{K_{P1}}(\gamma^1)$ to P2. P2 then encrypts $E_{K_{P1}}(\gamma^1)$ and sends a copy of $E_{K_{P2}}(E_{K_{P1}}(\gamma^1))$ back to P1. γ^2 is treated similarly. After this exchange, both parties have copies of $\left[E_{K_{P2}}(E_{K_{P1}}(\gamma^1)), E_{K_{P1}}(E_{K_{P2}}(\gamma^2))\right]$. Note that the encryption is applied to individual value, and we also adopt the commutative encryption scheme described in [19], but any other commutative encryption scheme can also be used. The key property of this scheme is that $E_{K_{P2}}(E_{K_{P1}}(v)) = E_{K_{P1}}(E_{K_{P2}}(v))$: encryption order does not matter.

Algorithm 2. DPP$_2$GA

Require: Private Data $T1$, $QI = (A_1, \ldots, A_n)$, Constraint k, Hierarchies VGH_{A_i}, where $i = 1, \ldots, n$, assume $k \leq |T1|$

1: P1 generalizes his data to be locally k-anonymous;
2: int $c \leftarrow 0$;
3: **repeat**
4: $c = c + 1$;
5: P1 computes γ_c^1;
6: P1 computes $E_{K_{P1}}(\gamma_c^1)$ and sends it to P2;
7: P1 receives $E_{K_{P2}}(\gamma_c^2)$ and computes $\Gamma_{P2} = E_{K_{P1}}(E_{K_{P2}}(\gamma_c^2))$;
8: P1 receives $\Gamma_{P1} = E_{K_{P2}}(E_{K_{P1}}(\gamma_c^1))$;
9: **until** $\Gamma_{P1} \equiv \Gamma_{P2}$
10: *return* $T_k[\text{QI}] \leftarrow T1_{\gamma_c^1} \bowtie T2_{\gamma_c^2}$;

Key steps in our approach are highlighted in Algorithm 2. The algorithm is written as executed by P1. Note that synchronization is needed for the counter c, and the encryption keys are different for each round. When the *loop* is executed more than once, the algorithm requires local data to be generalized one step further before computing the next γ_c^1 at Step 5. At step 10, the symbol $\bowtie$ represents the one-to-one join operator on the ID attribute to create globally k-anonymous dataset from the two locally k-anonymous datasets.

3.2 Proof of Correctness

In this section, we prove Algorithm 2 achieves global k-anonymity. Refer to notations adopted in Section 3.1, let γ_c^1, γ_c^2 synchronously computed from P1 and P2's locally k-anonymous data and use the equality operator $\equiv$ defined in Definition 3. Define $T1_{\gamma_c^1}$ and $T2_{\gamma_c^2}$ as the locally k-anonymous data related to γ_c^1 and γ_c^2 respectively.

Theorem 3. *If $\gamma_c^1 \equiv \gamma_c^2$, then $T_k[QI] \leftarrow T1_{\gamma_c^1} \bowtie T2_{\gamma_c^2}$ satisfies global k-anonymity.*

Proof. Let's prove the above theorem by contrapositive. In other words, prove the following statement: If $T_k[\text{QI}]$ does not satisfy global k-anonymity, then $\gamma_c^1 \neq \gamma_c^2$. Suppose $T_k[\text{QI}]$ is not k-anonymous, then there exists a subset of records $S = \{t_1, \ldots, t_j\} \subset T_k[\text{QI}]$ such that $|S| < k$ or $j < k$. Let $t_j[\gamma_c^1]$ denote the portion of the record t_j related to γ_c^1 stored at P1 and $t_j[\gamma_c^2]$ denote the portion of the record related to γ_c^2 stored at P2. Then $\{t_1[\gamma_c^1], \ldots, t_j[\gamma_c^1]\}$ must be contained in some subset $\gamma_c^1[p]$, and $\{t_1[\gamma_c^2], \ldots, t_j[\gamma_c^2]\}$ must be contained in some subset $\gamma_c^2[q]$; as a result, $|\gamma_c^1[p] \cap \gamma_c^2[q]| < k$. According to Definition 3, the equality between γ_c^1 and γ_c^2 does not hold. Thus, the contrapositive statement is true, so Theorem 3 holds. □

3.3 Proof of Privacy Preservation

Referring to Step 9 in Algorithm 2, although equality is tested on the encrypted version of γ_c^1 and γ_c^2, inference problems do exist.

For simplicity and consistency, let's use γ_c^1 and γ_c^2 instead of Γ_{P1} and Γ_{P2} for the following analysis. The inference problem exists only when $\gamma_c^1 \neq \gamma_c^2$. More specifically, we analyze the inference problem when $0 < |\gamma_c^1[p] \cap \gamma_c^2[q]| < k$ (for some p and q) because this inference seemingly violates global k-anonymity.

We classify inference problems into two types: final inference problem (FIP) and intermediate inference problem (IIP). FIP refers to the implication when the inequality occurs at Step 9 of Algorithm 2 only once. IIP refers to the implication when the inequality occurs multiple times. Let $T_k[\text{QI}]$ be the k-anonymous data computed by Algorithm 2.

Theorem 4. *FIP does not violate the privacy constraint P (previously stated in this section); in other words, FIP does not make $T_k[QI]$ less k-anonymous.*

Proof. If $\gamma_c^1 \neq \gamma_c^2$, then according to Definition 3, there must exist an intersection set $I_c = \gamma_c^1[p] \cap \gamma_c^2[q]$ such that $0 < |I_c| < k$. Since the equality test at Step 9 of Algorithm 2 is performed on the encrypted versions of γ_c^1 and γ_c^2, we are not able to know the exact records in I_c. Because of the definition of FIP, $\gamma_{c+1}^1 \equiv \gamma_{c+1}^2$ holds. Since γ_{c+1}^i computed from more generalized data than γ_c^i, the following conditions hold:

- $\gamma_c^1[p] \subseteq \gamma_{c+1}^1[p']$, where $1 \leq p' \leq |\gamma_{c+1}^1|$
- $\gamma_c^2[q] \subseteq \gamma_{c+1}^2[q']$, where $1 \leq q' \leq |\gamma_{c+1}^2|$

When the final generalized data released, for the worst case scenario, we may be able to identify unencrypted records related to $\gamma^1_{c+1}[p']$ and $\gamma^2_{c+1}[q']$. Define $I_{c+1} = \gamma^1_{c+1}[p'] \cap \gamma^2_{c+1}[q']$. According to the above conditions and $\gamma^1_{c+1} \equiv \gamma^2_{c+1}$, $I_c \subset I_{c+1}$ and $|I_{c+1}| \geq k$.

Since the equality test was performed on encrypted data, $Prob[x \triangleright y] = \frac{|I_c|}{|I_{c+1}|}$, where $x \in I_{c+1}$ and $y \in I_c$. If x is not directly generalized from y of any I_c, then $Prob[x \triangleright t \in T] \leq \frac{1}{k}$ because x is k-anonymous. If $x \triangleright y$, then $Prob[x \triangleright t \in T] = Prob[x \triangleright y] \cdot Prob[y \triangleright t]$. y is $|I_c|$-anonymous, so $Prob[y \triangleright t] = \frac{1}{|I_c|}$. Then we have $Prob[x \triangleright t \in T] = \frac{|I_c|}{|I_{c+1}|} \cdot \frac{1}{|I_c|} \leq \frac{1}{k}$. □

Next, we show a concrete example that illustrates why FIP does not violate k-anonymity. Refer to $\gamma^1_1, \gamma^2_1, \gamma^1_2, \gamma^2_2$ in Section 3.1. Let $\gamma^i_c = \gamma^i_1$ and $\gamma^i_{c+1} = \gamma^i_2$ where $i \in \{1, 2\}$. As stated previously, we have $\gamma^1_1 \neq \gamma^2_1$, so let $\gamma^1_c[p] = \{1, 3, 9\}$ and $\gamma^2_c[q] = \{2, 3, 9\}$. Then we have $I_c = \gamma^1_c[p] \cap \gamma^2_c[q] = \{3, 9\}$, $\gamma^1_{c+1}[p'] = \{1, 3, 9\}$, $\gamma^2_{c+1}[q'] = \{1, 2, 3, 4, 9, 10\}$ and $I_{c+1} = \gamma^1_{c+1}[p'] \cap \gamma^2_{c+1}[q'] = \{1, 3, 9\}$. Note that in this example, we can directly observe record IDs. However, in the real execution of the protocol, each party can only see the encrypted ID values. Now let's see if the data records contained in I_c violate the property stated in Theorem 2. Let $x \triangleright y \in I_c$, then $Prob[x \triangleright t \in T] = Prob[x \triangleright y] \cdot Prob[y \triangleright t] = \frac{|I_c|}{|I_{c+1}|} \cdot \frac{1}{|I|} = \frac{1}{3} = \frac{1}{k}$.

Theorem 5. *IIP does not violate the privacy constraint P; in other words, IIP does not make $T_k[QI]$ less k-anonymous.*

Proof. Use the notations defined in the proof of Theorem 4. According to the definition of IIP, $\gamma^1_c \neq \gamma^2_c$ and $\gamma^1_{c+1} \neq \gamma^2_{c+1}$. Define $I_c = \gamma^1_c[p] \cap \gamma^2_c[q]$ such that $0 < |I| < k$. Similar to the previous analysis, the following two conditions hold:

- $\gamma^1_c[p] \subseteq \gamma^1_{c+1}[p']$, where $1 \leq p' \leq |\gamma^1_{c+1}|$
- $\gamma^2_c[q] \subseteq \gamma^2_{c+1}[q']$, where $1 \leq q' \leq |\gamma^2_{c+1}|$

Define $I_{c+1} = \gamma^1_{c+1}[p'] \cap \gamma^2_{c+1}[q']$. If I_{c+1} is k-anonymous or $|I_{c+1}| \geq k$, then this inference problem caused by I_c is the same as FIP.

Now consider the case where $|I_{c+1}| < k$. Because γ^i_{c+1} computed from more generalized data than γ^i_c, $I_c \subseteq I_{c+1}$. If $|I_c| = |I_{c+1}|$, the inference effect caused by I_c does not propagate to the equality test between γ^1_{c+1} and γ^2_{c+1}. If $|I_c| < |I_{c+1}|$, define $x \in I_{c+1}$ and $y \in I_c$. If x is not directly generalized from y, then $Prob[x \triangleright t \in T] = \frac{1}{|I_{c+1}|}$ because x is $|I_{c+1}|$-anonymous. Nevertheless, if $x \triangleright y$, then $Prob[x \triangleright t \in T] = Prob[x \triangleright y] \cdot Prob[y \triangleright t] = \frac{|I_c|}{|I_{c+1}|} \cdot \frac{1}{|I_c|} = \frac{1}{|I_{c+1}|}$. As a result, $Prob[x \triangleright t \in T]$ is the same for all records in I_{c+1}. The inference effect caused by I_c is independent from one equality test to the next one. Consequently, the effect of IIP is the same as that of FIP. □

The equality test between γ^1_c and γ^2_c is not the focal point of this paper. It is fairly simple to derive, so we do not provide any specifics about how to perform the equality test. In addition, we note that if $|I_c| \geq k$, the records in the I_c do not violate the privacy constraint due to the definition of k-anonymity.

4 Conclusion / Future Work

Privacy of information in databases is an increasingly visible issue. Partitioning data is effective at preventing misuse of data, but it also makes beneficial use more difficult. One way to preserve privacy while enabling beneficial use of data is to utilize k-anonymity for publishing data. Maintaining the benefits of partitioning while generating integrated k-anonymous data requires a protocol that does not violate the k-anonymity privacy constraint. In this paper, we have laid out this problem and presented a two-party protocol DPP$_2$GA that is proven to preserve the constraint. It is a generic protocol in a sense that any k-anonymity protocol can be used to compute locally k-anonymous data.

One disadvantage of DPP$_2$GA is that it may not produce as precise data (with respect to the precision metric defined in [8]) as other k-anonymity algorithms do when data are not partitioned. For instance, DPP$_2$GA could be modified to simulate Datafly. At Step 9 of Algorithm 2, when the equality does not hold, only the party with the attribute that has most distinct values globally should generalize the data. Then the equality test would be performed on the newly computed Γ^1_{c+1} with previously used Γ^2_c. The data generated this way are the same as those computed by Datafly.

Even though this approach may produce more precise data, it does introduce additional inference problems because some Γ^i_{c+j} may be compared more than once. It is not obvious that this additional inference must (or can) violate k-anonymity with respect to individual parties, but proving this formally is not an easy task. One key design philosophy of DPP$_2$GA is to provably eliminate such inference problems, so DPP$_2$GA sacrifices a certain degree of precision. More precise protocols with fewer or no inference problems are a worthwhile challenge for future research. Another observation we have during the design of DPP$_2$GA is that more precise data can also be generated by removing already k-anonymous data at the end of each round (resulting in different data being generalized to different levels). Again, providing a formal method to analyze the inference problem might be very difficult, but this provides a valuable future research direction.

DPP$_2$GA is not a SMC protocol because it introduces certain inference problems, such as FIP and IIP. However, based on our analyses, both FIP and IIP do not violate the k-anonymity privacy constraint. Formally defining and understanding the differences between privacy-preserving and Secure Multiparty Computation may open up many new opportunities for designing protocols that preserve privacy.

Acknowledgements

We wish to thank Professor Elisa Bertino for comments and discussions that lead to this work.

References

1. Agrawal, R., Srikant, R.: Privacy-preserving data mining. In: Proceedings of the 2000 ACM SIGMOD Conference on Management of Data, Dallas, TX, ACM (2000) 439–450

2. Moore, Jr., R.A.: Controlled data-swapping techniques for masking public use microdata sets. Statistical Research Division Report Series RR 96-04, U.S. Bureau of the Census, Washington, DC. (1996)
3. Dobkin, D., Jones, A.K., Lipton, R.J.: Secure databases: Protection against user influence. ACM Transactions on Database Systems **4** (1979) 97–106
4. Yao, A.C.: Protocols for secure computation. In: Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science, IEEE (1982) 160–164
5. Yao, A.C.: How to generate and exchange secrets. In: Proceedings of the 27th IEEE Symposium on Foundations of Computer Science, IEEE (1986) 162–167
6. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game - a completeness theorem for protocols with honest majority. In: 19th ACM Symposium on the Theory of Computing. (1987) 218–229
7. Sweeney, L.: k-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems **10** (2002) 557–570
8. Sweeney, L.: Achieving k-anonymity privacy protection using generalization and suppression. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems **10** (2002) 571–588
9. Goldreich, O.: General Cryptographic Protocols. In: The Foundations of Cryptography. Volume 2. Cambridge University Press (2004)
10. Lindell, Y., Pinkas, B.: Privacy preserving data mining. Journal of Cryptology **15** (2002) 177–206
11. Du, W., Zhan, Z.: Building decision tree classifier on private data. In Clifton, C., Estivill-Castro, V., eds.: IEEE International Conference on Data Mining Workshop on Privacy, Security, and Data Mining. Volume 14., Maebashi City, Japan, Australian Computer Society (2002) 1–8
12. Vaidya, J., Clifton, C.: Privacy preserving naïve bayes classifier for vertically partitioned data. In: 2004 SIAM International Conference on Data Mining, Lake Buena Vista, Florida (2004) 522–526
13. Kantarcıoğlu, M., Clifton, C.: Privately computing a distributed k-nn classifier. In Boulicaut, J.F., Esposito, F., Giannotti, F., Pedreschi, D., eds.: PKDD2004: 8th European Conference on Principles and Practice of Knowledge Discovery in Databases, Pisa, Italy (2004) 279–290
14. Kantarcıoğlu, M., Jin, J., Clifton, C.: When do data mining results violate privacy? In: Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA (2004) 599–604
15. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. In: Proceedings of the IEEE Symposium on Research in Security and Privacy, Oakland, CA (1998)
16. Sweeney, L.: Computational Disclosure Control: A Primer on Data Privacy Protection. PhD thesis, Massachusetts Institute of Technology (2001)
17. Hundepool, A., Willenborg, L.: μ- and τ-argus: software for statistical disclosure control. Third International Seminar on Statistical Confidentiality (1996)
18. Sweeney, L.: Guaranteeing anonymity when sharing medical data, the datafly system. Proceedings, Journal of the American Medical Informatics Association (1997)
19. Pohlig, S.C., Hellman, M.E.: An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. IEEE Transactions on Information Theory **IT-24** (1978) 106–110