

Algorithm-Safe Privacy-Preserving Data Publishing

Xin Jin
George Washington University
USA
xjin@gwu.edu

Nan Zhang^{*}
George Washington University
USA
nzhang10@gwu.edu

Gautam Das[†]
University of Texas at Arlington
USA
gdas@uta.edu

ABSTRACT

This paper develops toolsets for eliminating algorithm-based disclosure from existing privacy-preserving data publishing algorithms. We first show that the space of algorithm-based disclosure is larger than previously believed and thus more prevalent and dangerous. Then, we formally define Algorithm-Safe Publishing (ASP) to model the threats from algorithm-based disclosure. To eliminate algorithm-based disclosure from existing data publishing algorithms, we propose two generic tools for revising their design: *worst-case eligibility test* and *stratified pick-up*. We demonstrate the effectiveness of our tools by using them to transform two popular existing ℓ -diversity algorithms, Mondrian¹ and Hilb, to SP-Mondrian and SP-Hilb which are algorithm-safe. We conduct extensive experiments to demonstrate the effectiveness of SP-Mondrian and SP-Hilb in terms of data utility and efficiency.

1. INTRODUCTION

1.1 Privacy-Preserving Data Publishing

Many organizations, such as hospitals, require publishing microdata with personal information, such as medical records, for facilitating research and serving public interests. Nonetheless, such publication may incur privacy concerns for the individual owners of tuples being published (e.g., patients). To address this challenge, privacy-preserving data publishing (i.e., PPDP) was proposed to generate the published table in a way that enables analytical tasks (e.g., aggregate query answering, data mining) over the published data, while protecting the privacy of individual data owners.

In general, a *microdata* table (denoted by \mathbf{T}) can contain three types of attributes: 1) *personal identifiable* attributes (e.g., *SSN*), each of which is an explicitly unique identifier of an individual, 2) *quasi-identifier* (QI) attributes (e.g., *Age*, *Sex*, *Country*), which are

^{*}Partly supported by NSF grants 0852673, 0852674, 0845644 and 0915834.

[†]Partly supported by NSF grants 0845644, 0812601, 0915834 and grants from Microsoft Research and Nokia Research.

¹Throughout our paper, we use the term Mondrian and Mondrian ℓ -diversity, Hilb and Hilb ℓ -diversity interchangeably.

not explicit identifiers but, when combined together, can be empirically unique for each individual, and 3) *sensitive attributes* (SA) (e.g., *Disease*), each of which contains a sensitive value (set) that must be protected. In privacy-preserving data publishing, personal identifiable attributes are usually removed prior to publishing. QI and/or SA attributes are perturbed to achieve a pre-defined privacy model while maximizing the utility of published data.

Samarati and Sweeney [32] first proposed a privacy model, k -anonymity, for PPDP. It requires each tuple in the published table (denoted by \mathbf{T}^*) to have at least $k - 1$ other tuples with the same QI attribute values. To protect individual SA information, Machanavajjhala et al [26] introduced another privacy model, ℓ -diversity, which further requires each group of QI-indistinguishable tuples (i.e., QI-group) to have diverse SA values. Variations of the ℓ -diversity model include (α, k) -anonymity [35], t -closeness [20], (k, e) -anonymity [42], m -invariance [38], etc. To satisfy these privacy models, numerous PPDP algorithms have been proposed [9, 16–18, 36, 42].

1.2 Algorithm-Based Disclosure

It was traditionally believed that, to determine whether a privacy model is properly satisfied, one only needs to look at the published table, i.e., the output of a data publishing algorithm, but does not need to investigate the algorithm itself. Algorithm-based disclosure contradicts this traditional belief as it refers to the privacy disclosure caused by the *design* of a data publishing algorithm. Intuitively, when an adversary learns the details of an algorithm, s/he may utilize this knowledge to reverse-engineer the published table to compromise additional private information. We shall discuss the details in Section 2.

Wong et al. [34] demonstrated the first known case of algorithm-based disclosure by showing that the minimality principle used by many existing algorithms, i.e., to perturb QI with the minimum degree possible for satisfying the privacy model, may lead to the disclosure of private SA information when the adversaries have the original QI as *external knowledge*. An example of this disclosure will be explained in Section 2. To counteract this attack, Wong et al. proposed a new privacy model called m -confidentiality [34], which guarantees that even an adversary with knowledge of QI cannot have confidence of more than $1/m$ on the SA value of an individual tuple. [41] also studied this attack and proposed a new privacy model called p -safety.

Algorithm-based disclosure poses a significant threat to the privacy of published data, because the data publishing algorithm is usually considered public and may be learned by an adversary. One might argue that, given the large number of public algorithms that are available for PPDP, it is difficult for an adversary to precisely identify which algorithm has been used and thereby to launch the algorithm-based attack. This is a typical “security through obscu-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

EDBT 2010, March 22–26, 2010, Lausanne, Switzerland.

Copyright 2010 ACM 978-1-60558-945-9/10/0003 ...\$10.00

urity” argument which counts on the secrecy of an algorithm to ensure the security of its output. However, such arguments have been repeatedly argued against and aborted in the literature of security and cryptography. As the Kerckhoff’s principle [13] in cryptography states, “The cipher method must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience.” Similarly, we argue that, to design an effective algorithm for privacy-preserving data publishing, one must eliminate algorithm-based disclosure.

To address algorithm-based disclosure, the existing work defined new privacy models such as m -confidentiality [34] and p -safety [41] which by definition require safety against algorithm disclosure. In addition, some recently proposed privacy models such as differential privacy [7] are also by definition immune from algorithm-based disclosure. While defining these new privacy models and developing their corresponding new algorithms provides a clean solution for eliminating algorithm-based disclosure, limiting the investigation of algorithm-based disclosure to this realm has a number of problems.

First, the state-of-the-art PDP calls for a proper understanding of the scope of algorithm-based disclosure for the existing data publishing algorithms. Currently, unless a data publishing algorithm is designed for an inherently algorithm-disclosure-safe privacy model such as differential privacy, it is unclear how to determine whether the algorithm is vulnerable to algorithm-based disclosure. Meanwhile, there are considerable ongoing efforts [15, 25] on developing data publishing algorithms for popular privacy models such as ℓ -diversity which do not provide such definition-inherent guarantee against algorithm-based disclosure. To enable the safe deployment of these algorithms in practice, it is important to understand whether and how algorithm-based disclosure may occur for a given data publishing algorithm.

Furthermore, the wide prevalence of data publishing algorithms call for a generic method that can revise the design of a given existing algorithm for eliminating algorithm-based disclosure. In the literature, for popular privacy models such as ℓ -diversity, there have been not only a myriad of algorithms for publishing tabular data, but also numerous others that publish application-specific data such as location [24], social network [23], and transaction information [40]. Instead of recreating algorithms for all these applications, We argue that a more cost-effective way is to develop a generic method that eliminates algorithm-based disclosure from the existing algorithms.

1.3 Outlines of Our Results

In this paper, we attack the problem of algorithm-based disclosure from a novel algorithmic angle. In particular, we first illustrate the challenge of identifying algorithm-based disclosure by demonstrating that the space of such disclosure is substantially larger than previously recognized. Then, we provide an exploratory tool for testing whether a given data publishing algorithm may lead to algorithm-based disclosure. Finally, we develop two methods, *worst-case eligibility test* and *stratified pick-up*, to revise the design of existing data publishing algorithms such that algorithm-based disclosure can be eliminated while a high level of utility is retained for the published table.

Our detailed results can be stated as follows:

First, we find that the space of algorithm-based disclosure is much broader than previously discovered. While the previous work identifies algorithm-based disclosure when an adversary holds external knowledge about the QI attributes, we find that other forms of external knowledge, such as the distribution of SA values and/or certain negative association rules [21] can also give rise to algorithm-based disclosure. Our further investigation even elim-

inates the dependency of algorithm-based disclosure on external knowledge, that is, we find algorithm-based disclosure can happen at the occasion when the adversary holds no external knowledge about the published data. To this end, we find that MASK [34], originally proposed to eliminate the previously discovered algorithm-based disclosure, actually suffers from another type of algorithm-based disclosure we discover in the paper.

Second, we propose an exploratory tool for checking whether a given data publishing algorithm is vulnerable to algorithm-based disclosure. In order to do so, we first introduce *Algorithm-Safe data Publishing (ASP)*, a model that formally defines algorithm-based disclosure as the difference between two random worlds: a *naive* one where every possible mapping between an original table and the published table is equally likely unless such a mapping violates an adversary’s external knowledge, and a *smart* one where the mapping must also follow the data publishing algorithm. An algorithm satisfies ASP iff it always maintains equivalence between these two worlds.

We derive two necessary conditions of ASP which are then used to identify the vulnerability of several existing data publishing algorithms. We also derive a sufficient condition for ASP which is used to prove the immunity of several other algorithms to algorithm-based disclosure. We refer to this sufficient condition as the *simulatable publishing* design paradigm. Intuitively, simulatable publishing requires the published QI to be conditionally independent of the original SA given the original QI and the published SA as prior knowledge. In other words, no unpublished QI-SA correlation information is used for generating the published table. The combination of these necessary and sufficient conditions forms an exploratory tool for checking whether a given data publishing algorithm is vulnerable to algorithm-based disclosure.

Finally, we develop two tools, worst-case eligibility test and stratified pick-up, for revising the design of existing algorithms to follow the simulatable publishing paradigm. The first tool is designed to amend the most common violation of ASP found in existing algorithms - the QI-grouping strategy - and make it adhere to the simulatable publishing paradigm. To demonstrate its effectiveness, we apply worst-case eligibility test to revise two well-known data publishing algorithms, Mondrian [17] and Hilb [9] ℓ -diversity algorithms, and prove that the revised algorithms satisfy ASP. The second tool is designed to improve the utility of published data without violating the simulatable publishing paradigm. In particular, it uses an anatomy-like [36] technique to minimize the number of tuples in each published QI-group. To demonstrate its effectiveness, we apply stratified pick-up on top of the first-tool output of Mondrian and Hilb to produce SP-Mondrian and SP-Hilb, respectively, and find that they provide almost equal or even better utility than the original Mondrian and Hilb algorithms, respectively.

We validate the theoretical results and evaluate the effectiveness of our algorithms by a comprehensive set of experiments on the real-world dataset. To demonstrate the effectiveness of our simulatable publishing paradigm, we evaluate the utility of SP-Mondrian and SP-Hilb and compare them against various existing ℓ -diversity algorithms. Experimental results show that while eliminating algorithm-based disclosure, our algorithms remain efficient and achieve almost equal (SP-Hilb) or significantly better (SP-Mondrian) data utility than the existing algorithms.

The rest of the paper is organized as follows. Section 2 shows motivating examples for algorithm-based disclosure. Section 3 introduces background and notations used in the paper. Section 4 formally defines ASP. Section 5 derives two necessary conditions and one sufficient condition for ASP, and verifies the vulnerability of existing algorithms. Section 6 gives two generic tools, and adapts two existing algorithms via these tools to eliminate algorithm-based

Table 1: An example of algorithm-based disclosure in ℓ -diversity algorithm

(a) microdata			(b) 2-diversity table		(c) external knowledge		(d) (1M,1AIDS)		(e) (3M,2AIDS)		(f) (2M,0AIDS)		(g) (2M,1AIDS)	
row#	Sex	Disease	Sex	Disease	Name	Sex	Sex	Disease	Sex	Disease	Sex	Disease	Sex	Disease
1	F	gastritis	F	gastritis	Amy	F	F	gastritis	F	gastritis	F	gastritis	F	gastritis
2	F	heart disease	F	heart disease	Eva	F	F	heart disease	F	heart disease	F	heart disease	F	heart disease
3	F	cancer	*	cancer	Grace	F	*	cancer	*	cancer	F	AIDS	F	cancer
4	F	diabetes	*	diabetes	Helen	F	*	AIDS	*	AIDS	F	AIDS	F	AIDS
5	M	AIDS	*	AIDS	Jack	M	M	diabetes	M	diabetes	M	cancer	M	diabetes
6	M	AIDS	*	AIDS	Tom	M	M	AIDS	M	AIDS	M	diabetes	M	AIDS

Table 2: An example of algorithm-based disclosure in MASK algorithm

(a) microdata					(b) k -anonymity table ($k = 4$)					(c) m -confidentiality ($m = 2$)				
row #	Age	Sex	Country	Disease	Age	Sex	Country	Disease	Age	Sex	Country	Disease		
1	46	F	Mexico	cancer	[32 - 49]	F	Mexico	cancer	[32 - 49]	F	Mexico	cancer		
2	49	F	Mexico	heart disease	[32 - 49]	F	Mexico	heart disease	[32 - 49]	F	Mexico	heart disease		
3	32	F	Mexico	heart disease	[32 - 49]	F	Mexico	heart disease	[32 - 49]	F	Mexico	heart disease		
4	35	F	Mexico	AIDS	[32 - 49]	F	Mexico	AIDS	[32 - 49]	F	Mexico	AIDS		
5	24	F	Japan	AIDS	[24 - 38]	*	Japan	AIDS	[24 - 38]	*	Japan	cancer		
6	38	F	Japan	AIDS	[24 - 38]	*	Japan	AIDS	[24 - 38]	*	Japan	cancer		
7	25	M	Japan	AIDS	[24 - 38]	*	Japan	AIDS	[24 - 38]	*	Japan	heart disease		
8 (Tom)	37	M	Japan	AIDS	[24 - 38]	*	Japan	AIDS	[24 - 38]	*	Japan	AIDS		

disclosure. We conduct experiments in Section 7, review the related works in Section 8, and conclude in Section 9.

2. MOTIVATING EXAMPLES

This section describes two motivating examples of algorithm-based disclosure. We consider two adversaries: “naive” Nash and “smart” Sam throughout the paper. Both of them hold the same external knowledge and observe the same published table. The only difference is that “naive” Nash does not know the data publishing algorithm, whereas “smart” Sam does. Both Nash and Sam want to compromise whether their friend Tom, a 37-year-old male from Japan, has AIDS or not.

For the ease of discussion, we follow the same SA settings as [29, 34]: infectious disease {AIDS} is sensitive, while noninfectious diseases {cancer, diabetes, gastritis, heart disease} are non-sensitive.

2.1 Example 1: Disclosure of ℓ -diversity Algorithms Based on QI Generalization

Consider a generalization algorithm (e.g., [26]) that achieves ℓ -diversity. Table 1a depicts a microdata table with one QI attribute *Sex* and one SA *Disease*. Table 1b is 2-diversity version of Table 1a, such that the proportion of any sensitive SA value in one QI-group is at most $\frac{1}{\ell} = \frac{1}{2}$.

First, let us review the case discussed in [34], where both “naive” Nash and “smart” Sam know original QI (Table 1c) as external knowledge. What Nash can do is to join Table 1c with the published Table 1b to infer that Tom belongs to the “*”-group. Thus, from Nash’s view, the probability of Tom having AIDS is $\frac{1}{2}$, which does not violate 2-diversity. We now consider “smart” Sam who knows that the generalization algorithm will not generalize any group unless it violates 2-diversity. Based on this, Sam can infer that no generalization would have been conducted if the 2 males had 0 or 1 AIDS. Therefore, both males, including Tom, must have AIDS. Hence, by leveraging the algorithm-based knowledge, “smart” Sam acquires a different view from “naive” Nash, and Sam’s view violates 2-diversity. This is an example of algorithm-based disclosure.

Now, we show the limitation of [34] by demonstrating that algorithm-based disclosure may occur without involving any external knowl-

edge. Note that when “naive” Nash holds no external knowledge, his view of Tom’s SA is the same as what the published table discloses, which does not violate 2-diversity.

Consider the view of “smart” Sam. He can reason as follows: 1) the number of males in the table should be less than 4 but greater than 0, because otherwise no generalization would be needed; 2) if there were only 1 male, Table 1b would not be published because the algorithm would prefer an alternative FFFF** (i.e., Table 1d) to attain better data utility; 3) if there were 3 males, Table 1b would again not be published because of another alternative FF**MM (i.e., Table 1e) with better utility. Apparently, there is only one option left, that is, 2 males in the table. If none or only one of them had AIDS, no generalization would be needed (i.e., Table 1f and 1g). Thus, both males, including Tom, must have AIDS. Thereby, the knowledge of algorithm renders “smart” Sam’s view in violation of 2-diversity. As we can see, algorithm-based disclosure can exist without external knowledge.

2.2 Example 2: Disclosure of MASK Algorithm Based on SA Perturbation

MASK [34] is intended for eliminating algorithm-based disclosure by achieving m -confidentiality. m -confidentiality is essentially the same as ℓ -diversity (let $\ell = m$), except that m -confidentiality tries to protect privacy when an adversary has the original QI as external knowledge.

Consider a microdata table in Table 2a. Tables 2b and 2c depict an example of using MASK to achieve 2-confidentiality. MASK first applies k -anonymization ($k \geq m$) to the microdata table (e.g., 4-anonymity in Table 2b). Then, for each group violating ℓ -diversity (e.g., the “Japan” group), MASK randomly perturbs the sensitive SA values (e.g., AIDS) to non-sensitive values (e.g., cancer, heart disease), until the proportion of sensitive SA values is decreased to p , where p is the proportion of sensitive SA values from a randomly selected ℓ -diversity group (e.g., $p = \frac{1}{4}$ in the “Mexico” group).

We now show the existence of algorithm-based disclosure in Table 2c when an adversary knows a negative association rule from common-sense, say, “Japanese have an extremely low incidence of heart disease [26, 34]”. Consider the view of “naive” Nash. He can conclude from Table 2c that Tom is in the “Japan” group, and heart

disease must be a perturbed value because the heart disease rate in that group (i.e., 25%) conflicts with the negative association rule. But without knowing the MASK algorithm, “naive” Nash can only randomly guess the original value of heart disease to be AIDS or cancer. Thus, the probability of Tom having AIDS in his view is: $50\% \times \frac{1}{2} + 50\% \times \frac{1}{4} = \frac{3}{8}$. This does not violate 2-confidentiality.

Now consider the view of “smart” Sam, who is clear that MASK would not perturb any SA values in the “Japan” group unless the group violates 2-confidentiality after k -anonymization (i.e., Table 2b). Thus, Sam concludes that the “Japan” group should have at least 3 AIDS (out of 4 tuples). As such, in “smart” Sam’s view, the probability of Tom having AIDS is at least $\frac{3}{4}$, which violates 2-confidentiality. Again, knowing the algorithm empowers “smart” Sam to gain a different view from “naive” Nash, where Sam’s view violates m -confidentiality.

Consider another algorithm-based disclosure situation in MASK if an adversary has access to some original SA distribution. This is common in reality because data publishers may report statistics for public use. For example, in order to ease the fear of increasing cancer incidence in the community, a local hospital may announce that “only 1 out of 8 hospitalized patients has cancer”.

Now consider what “naive” Nash can compromise from the published Table 2c. He can confirm that MASK should have perturbed 2 SA values to cancer. However, he cannot further tell which 2 out of the 3 cancer are the perturbed values, and whether AIDS or heart disease is the original value. As such, the probability of Tom having AIDS in the view of “naive” Nash is $33.3\% \times \frac{3}{8} + 33.3\% \times \frac{3}{8} + 33.3\% \times \frac{1}{2} = \frac{5}{12}$. Likewise, this does not violate 2-confidentiality.

Whereas, “smart” Sam, who knows the MASK algorithm, can infer that the extra 2 cancer must be from the “Japan” group, and AIDS is their original value. The reason is: otherwise, MASK would not conduct any perturbation because both groups in the table after k -anonymization are already 2-confidentiality. Thereby, there exists algorithm-based disclosure because the probability of Tom having AIDS in Sam’s view is at least $\frac{3}{4}$, which violates 2-confidentiality.

As we can see, MASK is still subject to algorithm-based disclosure. And algorithm-based disclosure can exist along with various types of external knowledge, or even without external knowledge. We re-emphasize that this paper is aiming to limit the algorithm-based disclosure (i.e., the view of “smart” Sam), that is, the private information beyond what can be gained by external knowledge.

3. FORMAL FRAMEWORK

3.1 Privacy-Preserving Data Publishing

Consider $\mathbf{T} = \{t_1, \dots, t_n\}$, a microdata table of n tuples. Each t_i consists of d QI attributes (Q_1, Q_2, \dots, Q_d), denoted by Q and one SA attribute, denoted by S . For example, Table 2a is a table of $n = 8$ tuples. Each tuple has $d = 3$ QI attributes (i.e., *Age*, *Sex*, *Country*) and 1 SA (i.e., *Disease*). Let $t_i[Q] = (t_i[Q_1], t_i[Q_2], \dots, t_i[Q_d])$ be a conjunction value of QI attributes in tuple t_i ; let $t_i[S]$ be the attribute value of SA; let $\mathcal{D}_Q = \mathcal{D}_{Q_1} \times \mathcal{D}_{Q_2} \times \dots \times \mathcal{D}_{Q_d}$ and \mathcal{D}_S be the finite domain of Q and S , respectively. For $q \in \mathcal{D}_Q$ and $s \in \mathcal{D}_S$, we say that $(q, s) \in \mathbf{T}$ iff there exists $i \in [1, n]$ such that $t_i[Q] = q$ and $t_i[S] = s$.

Before the data release, a data publisher normally uses a data publishing algorithm \mathbf{A} to perturb the microdata \mathbf{T} . Let \mathbf{T}^* be the published table of \mathbf{T} ; let Q^* be the perturbed QI attributes in \mathbf{T}^* ; let \mathcal{D}_{Q^*} be the domain of Q^* . We require \mathcal{D}_S to be the domain of SA attribute in either \mathbf{T} or \mathbf{T}^* .

The objective of privacy-preserving data publishing is to prevent an adversary from learning the individual SA in the published table

Table 3: Q^* and S^* of TABLE 1b

(a) Q^*	(b) S^*				
Sex	AIDS	cancer	diabetes	gastritis	heart disease
F	0	0	0	1/2	1/2
*	1/2	1/4	1/4	0	0

Table 4: Q^* and S^* of TABLE 2c

(a) Q^*			(b) S^*		
Age	Sex	Country	AIDS	cancer	heart disease
[32 – 49]	F	Mexico	1/4	1/4	1/2
[24 – 38]	*	Japan	1/4	1/2	1/4

based on knowledge of QI. Thus, the *correlation* between QI and SA attributes in the published \mathbf{T}^* is the private information to be protected. We represent such private information by *QI-SA correlation* $S^*(\cdot)$, which is a function that maps the QI attributes of an individual tuple, $q \in \mathcal{D}_Q$, to the posterior distribution of the SA for that tuple. Table 3 and Table 4 are examples of Q^* and $S^*(\cdot)$ for Table 1b and Table 2c, respectively.

Formally, for each $(q, s) \in T$, $S^*(q)$ is a $|\mathcal{D}_S|$ -dimensional vector ($S^*(q)[s_1], S^*(q)[s_2], \dots, S^*(q)[s_{|\mathcal{D}_S|}]$) where $S^*(q)[s_j] = \Pr\{s = s_j | q, \mathbf{T}^*\}$.

We are now ready to state the ℓ -diversity privacy model [26] in terms of $S^*(\cdot)$. In particular, we adopt a simple variation of ℓ -diversity [9, 34, 36] which requires that no individual SA value can be compromised with probability over $\frac{1}{\ell}$:

DEFINITION 1. (ℓ -diversity [26]) A published table \mathbf{T}^* fulfills ℓ -diversity iff $\forall t_i = (q, s) \in \mathbf{T}$ and $\forall s_j \in \mathcal{D}_S$,

$$\max_{j \in [1, |\mathcal{D}_S|]} S^*(q)[s_j] \leq \frac{1}{\ell}.$$

3.2 Expression of External Knowledge

As shown in Section 2, algorithm-based disclosure does not mandate the adversary’s possession of external knowledge. Nonetheless, it is also clear from earlier discussion that certain type of external knowledge may facilitate algorithm-based disclosure. Thus, we now formulate an expression of external knowledge.

The only purpose of formalizing external knowledge is for the ease of understanding our ASP model to be discussed later. Therefore, we take a simple expression by conjunctive COUNT query. Given the microdata \mathbf{T} , consider a COUNT query $\text{CQ}(\mathbf{T})$ in the form:

```
SELECT COUNT(*) FROM  $\mathbf{T}$ 
WHERE ( $Q_1 = q_1$ )  $\wedge$   $\dots$   $\wedge$  ( $Q_d = q_d$ )  $\wedge$  ( $S = s$ )
```

Note that search condition (i.e., WHERE clause) does not need to include every $Q_j (j \in [1, d])$ or S in \mathbf{T} . We describe external knowledge \mathbf{K}_e as arithmetic equations (or inequalities) between a pair of COUNT query answers, or between one COUNT query answer and one constant.

Consider Table 2a as the microdata. An example of external knowledge about Tom, who is a 37-year-old male from Japan, is “Tom does not have cancer”. We can express such \mathbf{K}_e as $\text{CQ}(\mathbf{T}) = 0$ where $\text{CQ}(\mathbf{T}) = \text{SELECT COUNT(*) FROM } \mathbf{T} \text{ WHERE } \text{Age} = 37 \wedge \text{Sex} = \text{M} \wedge \text{Country} = \text{Japan} \wedge \text{Disease} = \text{cancer}$.

Another simple example of \mathbf{K}_e is “Japanese have an extremely low incidence of heart disease”, which can be described by $\text{CQ}(\mathbf{T})/|\mathbf{T}|$

$< 0.05^2$ where $CQ(\mathbf{T}) = \text{SELECT COUNT}(\ast) \text{ FROM } \mathbf{T} \text{ WHERE } \text{Country} = \text{Japan} \wedge \text{Disease} = \text{heart disease}$.

4. ALGORITHM-SAFE DATA PUBLISHING

This section formalizes algorithm-based disclosure by introducing a new model called *Algorithm-Safe data Publishing (ASP)*. A data publishing algorithm is vulnerable to algorithm-based disclosure when it violates ASP. We will first define two key concepts relating to ASP: a *naive random world* which models the view *without* knowledge of the algorithm, and a *smart random world* which models the view *with* such knowledge of the algorithm. Then, we will define ASP based on the equivalence between these two worlds.

4.1 Naive vs. Smart Random World

Recall that \mathcal{D}_Q and \mathcal{D}_S are the domains of QI and SA, respectively. Let Ω be a finite set of all possible values in the microdata that can be calculated from $\mathcal{D}_Q \times \mathcal{D}_S$. When an adversary with external knowledge \mathbf{K}_e observes a published table \mathbf{T}^* , his/her view on the microdata table \mathbf{T} can be modeled as a (posterior) probability distribution over Ω , that is, a *mapping* from any $T' \subseteq \Omega$ to a real value $\Pr(\mathbf{T} = T' | \mathbf{T}^*, \mathbf{K}_e) \in [0, 1]$, such that $\sum_{T' \subseteq \Omega} \Pr(\mathbf{T} = T' | \mathbf{T}^*, \mathbf{K}_e) = 1$.

Return to consider the different views from “naive” Nash and “smart” Sam, as illustrated in Section 2. Given a published table \mathbf{T}^* , for each $T' \subseteq \Omega$, Nash can check whether \mathbf{T}^* can possibly be perturbed from T' by certain data publishing algorithm \mathbf{A} , that is, to check whether T' is “consistent” with the published \mathbf{T}^* and satisfies the integrity conditions imposed by his external knowledge \mathbf{K}_e . Let $\mathcal{D}_{Q'}$ be the domain of QI attributes in T' . We say that T' is “consistent” with \mathbf{T}^* iff $\mathcal{D}_{Q'} \subseteq \mathcal{D}_{Q^*}$. We denote such “consistent” relationship by a partial order $T' \prec \mathbf{T}^*$.

Without learning the data publishing algorithm \mathbf{A} , “naive” Nash cannot distinguish any $T' \subseteq \Omega$. According to the standard random world assumption [26, 27], Nash has to assign the same probability to all tables that pass the above consistency check. Thus, we define the view of Nash as a *naive random world* $\mathcal{NW}(\cdot)$ as follows:

DEFINITION 2. (naive random world) A *naive random world* $\mathcal{NW}(\cdot)$ is a probability distribution, $\forall T' \subseteq \Omega$,

$$\mathcal{NW}(T') = \begin{cases} 1/c, & \text{if } T' \prec \mathbf{T}^* \text{ and } T' \text{ satisfies } \mathbf{K}_e. \\ 0, & \text{otherwise} \end{cases}$$

where $c = |\{T' | T' \prec \mathbf{T}^* \wedge T' \text{ satisfies } \mathbf{K}_e\}|$.

Consider the previous example of Table 1 in a simple way: AIDS is the sensitive SA value (shadow color), other SA values (blank color) are indistinguishable. Suppose adversaries have external knowledge \mathbf{K}_e in forms of “Amy and Grace are unlikely to have AIDS” and “at least 1 male has AIDS”. Table 5a shows an example of the naive random world $\mathcal{NW}(\cdot)$. In the view of “naive” Nash, there are totally 6 T' as a result of linking AIDS to the original QI attributes, such that \mathbf{K}_e is satisfied. Nash can not distinguish any T' because $\mathcal{NW}(T') = \frac{1}{6}$ holds true for every single T' in the naive random world. As we see, the probability distribution on these 6 T' constitutes the naive random world.

Next, consider the view of “smart” Sam who is clear about the details of the data publishing algorithm \mathbf{A} . Thus, other than consistency check, Sam can further distinguish $T' \subseteq \Omega$ by running the

²The value of 0.05 can be adjusted according to actual needs for reflecting the effect of “extremely low incidence”.

Table 5: An example of naive & smart random world

(a) naive random world						(b) smart random world			
Name	QI	SA	SA	SA	SA	Name	QI	SA	SA
Amy	F					Amy	F		
Eva	F	A	A			Eva	F		
Grace	F					Grace	F		
Helen	F			A	A	Helen	F		
Jack	M	A		A		Jack	M	A	
Tom	M		A		A	Tom	M	A	

algorithm \mathbf{A} on each T' to check whether \mathbf{A} can perturb the table T' to \mathbf{T}^* . Now we define the view of Sam as a smart random world $\mathcal{SW}(\cdot)$ as follows:

DEFINITION 3. (smart random world) A *smart random world* $\mathcal{SW}(\cdot)$ is a probability distribution, $\forall T' \subseteq \Omega$,

$$\mathcal{SW}(T') = \begin{cases} \Pr\{\mathbf{T} = T' | \mathbf{A}\}, & \text{if } T' \prec \mathbf{T}^* \text{ and } T' \text{ satisfies } \mathbf{K}_e. \\ 0, & \text{otherwise} \end{cases}$$

For the ease of illustration, we assume “smart” Sam to have a uniform prior in the above example. However, such assumption would by no means restrict the generality of our definition, which allows other distributions as well. What “smart” Sam can do is to run the 2-diversity algorithm in iterations, by accepting each T' in 5a as input. Table 5b shows the only legitimate T' that can be generalized to \mathbf{T}^* , because the other 5 T' already achieve 2-diversity and it is unnecessary to further generalize them. Therefore, “smart” Sam can conclude that $\mathcal{SW}(T') = 1$ holds for that legitimate T' , whereas $\mathcal{SW}(T') = 0$ for others. Note that such probability distribution constitutes the smart random world.

4.2 Definition of ASP

Now we can see that due to the existence of algorithm-based disclosure, even if given the same external knowledge \mathbf{K}_e and published table \mathbf{T}^* , the amount of disclosure in naive random world $\mathcal{NW}(\cdot)$ and smart random world $\mathcal{SW}(\cdot)$ can still be unequal. Hence, we say that there is *no* algorithm-based disclosure iff the two worlds are always equivalent conditioning on the same external knowledge and published table. Without loss of generality, we now define ASP.

DEFINITION 4. (algorithm-safe data publishing) A *published table* \mathbf{T}^* fulfills algorithm-safe data publishing iff $\forall t_i = (q, s) \in \mathbf{T}$ and $\forall s_j \in \mathcal{D}_S$, there is:

$$\Pr\{t_i[S] = s_j | t_i[Q] = q, \mathcal{NW}\} = \Pr\{t_i[S] = s_j | t_i[Q] = q, \mathcal{SW}\}$$

5. CHECKING ALGORITHM-BASED DISCLOSURE

This section presents two necessary conditions and one sufficient condition of ASP, which in combination serve as an exploratory tool to evaluate the vulnerability of a data publishing algorithm to algorithm-based disclosure. Meanwhile, we use the tool to identify the vulnerabilities of many data publishing algorithms.

5.1 Necessary Condition 1: Q^* -Independence

Recall from the definition of ASP that an algorithm satisfies ASP only if the naive and smart random worlds are equivalent given the same external knowledge. If there exists external knowledge that breaks such equivalence, then the algorithm becomes vulnerable to algorithm-based disclosure. Our first necessary condition, Q^* -Independence, targets specific external knowledge of original QI,

and requires no QI-SA correlation beyond \mathcal{S}^* to be used in the perturbation (e.g., generalization) of QI. Otherwise, algorithm-based disclosure may occur.

THEOREM 5.1. (Q^* -Independence) *If a published table \mathbf{T}^* satisfies ASP, then the published QI attributes Q^* must be conditionally independent with the original SA, given a combination of the original QI attributes Q and the published QI-SA correlation \mathcal{S}^* . In other words, generating Q^* from Q is conditional independent of the original SA given the published \mathcal{S}^* .*

Due to the page limitation, we omit formal proof of all theorems in the rest of our paper. An intuitive explanation can be stated as follows. Due to the definition of mutual information [5], if the mapping from Q to Q^* uses certain QI-SA correlation information that is not ultimately published, then such unpublished QI-SA correlation information can also be derived based on the mapping from Q to Q^* , which can be readily observed by an adversary through the external knowledge of Q . Hence, Q^* -Independence requires the perturbation of QI to be determined solely upon Q and \mathcal{S}^* .

Examples of existing data publishing algorithms in violation of Theorem 5.1 include most ℓ -diversity algorithms (and its variant) (e.g., [9, 10, 19, 20, 26, 35, 37, 42]) whose grouping strategy aims to produce each group with “similar” QI. We explain the reason as follows. To maximize the utility of published data, these algorithms greedily group QI-similar tuples into the smallest possible groups until a group violates the SA diversity requirement. This objective by itself demands the usage of unpublished QI-SA correlation because, clinging to the published information, it may not always be possible to determine whether pursuing a further grouping will violate the privacy model. By Theorem 5.1, Q^* -Independence is thus violated. Hence, all these algorithms are vulnerable to algorithm-based disclosure.

5.2 Necessary Condition 2: \mathcal{S}^* -Independence

Our second necessary condition, \mathcal{S}^* -Independence, targets another type of external knowledge, the negative association rules (e.g. “Tom is unlikely to have cancer”). In analogy to Q^* -independence, \mathcal{S}^* -independence states no QI-SA correlation beyond \mathcal{S}^* should be used in the perturbation (e.g., generalization) of SA.

Before presenting the theorem, let us first introduce a few notations for formalizing \mathcal{S}^* -Independence. Consider a tuple $t_i = (q, s) \in \mathbf{T}$. Let $t_i^* = (q^*, s^*) \in \mathbf{T}^*$ be the released value of t_i . Let \mathcal{D}_S be the domain of S . We describe a negative association rule about t_i as a set of “impossible” SA values $V_- \subseteq \mathcal{D}_S$, such that s^* cannot take any value in V_- . Let $V_+ = \mathcal{D}_S - V_-$ be the difference between \mathcal{D}_S and V_- . Recall from Section 3 that $\mathcal{S}^*(q)$ returns a probability distribution of all possible SA values from \mathbf{T}^* for t_i , let $\mathcal{S}^*(q)[V_+]$ be $\mathcal{S}^*(q)$ projected on the domain of V_+ .

Continue with the example “Tom is unlikely to have cancer”. Referring to Table 4, we have $V_- = \{\text{cancer}\}$ and $\mathcal{S}^*(q)[V_+] = \{\mathcal{S}^*(q)[\text{AIDS}], \mathcal{S}^*(q)[\text{heart disease}]\} = \{\frac{1}{4}, \frac{1}{4}\}$.

THEOREM 5.2. (\mathcal{S}^* -Independence) *Given a combination of QI attributes Q , “impossible” SA value set V_- and $\mathcal{S}^*(q)[V_+]$, if a published table satisfies ASP, then for any tuple $t_i = (q, s) \in \mathbf{T}$, $\mathcal{S}^*(q)$ must be conditionally independent of the original SA.*

The proof is built on the following observation. Consider the “impossible” SA value set V_- . Due to the definition of mutual information, if whether an “impossible” SA value will be included in \mathcal{S}^* depends on certain unpublished QI-SA correlation information, then such unpublished information can also be derived by observing whether an impossible SA value occurs in \mathcal{S}^* , i.e., whether $\mathcal{S}^*(q)[V_-]$ is empty. Note that $\mathcal{S}^*(q)[V_-]$ can be learned by any

adversary with external knowledge of V_- . Thus, \mathcal{S}^* -Independence requires the perturbation of SA to be determined solely upon Q , V_- and $\mathcal{S}^*[V_+]$.

For example, MASK [34] is vulnerable to algorithm-based disclosure due to violation of Theorem 5.2. Recall that MASK first checks whether a group violates ℓ -diversity, and will only perturb SA of groups that violate it. Again, this demands the usage of unpublished QI-SA correlation during SA perturbation because it may not always be possible to determine whether a group violates ℓ -diversity only based on published information. By Theorem 5.2, \mathcal{S}^* -Independence is violated.

5.3 Sufficient Condition: Simulatable Publishing

DEFINITION 5. (Simulatable Publishing) *A data publishing algorithm is simulatable publishing iff it satisfies both of the following two conditions:*

- Q^* is solely determined by Q and \mathcal{S}^* , and
- $\forall t_i \in \mathbf{T}$, $i \in [1, m]$, its perturbed tuple $t_i^* \in \mathbf{T}^*$ satisfies $t_i^*[S] = t_i[S]$.

In the definition, the first condition states that no unpublished QI-SA correlation should be used in generating Q^* , while the second condition requires that the published SA values remain authentic. With the two conditions, anyone who has access to the published table and the original Q can simulate³ the data publishing process without consulting any additional (unpublished) QI-SA correlation. Intuitively, this also guarantees that no unpublished QI-SA correlation can be inferred from the published table. Formally, we have the following theorem.

THEOREM 5.3. *If a data publishing algorithm is simulatable publishing, then all tables published by the algorithm fulfill ASP.*

Anatomy [36] is a typical example of simulatable publishing. The reasons are: 1) Anatomy uses only SA values to decide the grouping, and 2) it does not perturb SA at all. In other words, Anatomy neither perturbs SA values nor uses any QI-SA correlation beyond what will be eventually published. Therefore, this suffices to guarantee that Anatomy is immune from algorithm-based disclosure.

Although as shown in Theorem 5.3, simulatable publishing is a sufficient condition for ASP, it is not a necessary one. To see this, consider a simple publishing algorithm that changes SA to empty. The published table always satisfies ASP because no QI-SA correlation is disclosed. Nonetheless, it violates the second condition of simulatable publishing because the published SA values are changed (i.e., truncated) during the perturbation process (i.e., $t_i^*[S] \neq t_i[S]$).

6. OUR GENERIC TOOLS TO ELIMINATE ALGORITHM-BASED DISCLOSURE

This section introduces two generic tools by following our simulatable publishing paradigm: 1) worst-case eligibility test to amend the most common violation of ASP (i.e., QI-grouping strategy) in existing algorithms in terms of Theorem 5.1, and 2) stratified pick-up to improve utility. A study on the amendment regarding Theorem 5.2 will be our future work. To demonstrate our two tools,

³Note that such a simulation is *not* a duplication of the data publishing process. When the data publishing algorithm is non-deterministic, one can simulate the randomized part by using the same random number generator, but does not have to generate the same random number.

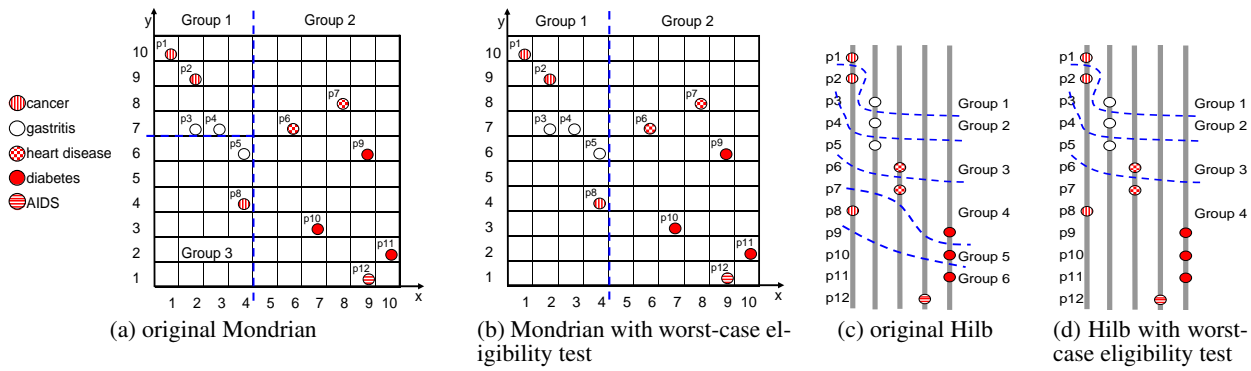


Figure 1: A running example when $\ell = 2$

we reuse the design of two popular ℓ -diversity algorithms: Mondrian [17] and Hilb [9], and then adapt them into SP-Mondrian and SP-Hilb which satisfy simulatable publishing.

For the ease of understanding, we use a simple dataset (in Figure 1a) to be our running example in this section. The simple dataset is a table \mathbf{T} of 12 tuples with two QI attributes x and y . Figure 1a shows point representations of all the 12 tuples in an x - y plane, where there are 5 different patterns to denote 5 different SA value, respectively.

6.1 Worst-case Eligibility Test

The main idea of worst-case eligibility test is to enable a data publishing algorithm to generate a partition only if a worst (i.e., most skewed) case of QI-SA correlation is able to achieve the pre-defined privacy model such as ℓ -diversity. The worst-case eligibility test guarantees that no information beyond unpublished QI-SA correlation information will be used. To see how it works, we now show how to integrate the tool of worst-case eligibility test into two existing ℓ -diversity algorithms: Mondrian and Hilb.

6.1.1 In the Case of Mondrian

We first elaborate the details of Mondrian ℓ -diversity, and then show how to adapt it by integrating worst-case eligibility test.

The original Mondrian ℓ -diversity algorithm works in a recursive fashion as follows. It starts with choosing the split attribute with the largest range of values [17], or based on certain utility measure [18]. Then Mondrian repetitively partitions G (initially to be \mathbf{T}) into two groups G_1 and G_2 , where G_1 and G_2 include the points of G divided by the median coordinate on the split attribute. Let $|\cdot|$ be the number of points in the set and $s_{max}(\cdot)$ be the number of most frequent SA value in the set. If either $|G_1| < \ell \times s_{max}(G_1)$ or $|G_2| < \ell \times s_{max}(G_2)$ holds, such partitioning has to *fall back*.

Refer to our example in Figure 1a. Mondrian ℓ -diversity first chooses the split attribute x and partitions the 12 tuples by the median coordinate ($x = 4$) into two groups $G_1 = \{p_1, p_2, p_3, p_4, p_5, p_8\}$ and $G_2 = \{p_6, p_7, p_9, p_{10}, p_{11}, p_{12}\}$ because $|G_1| = 6 \geq \ell \times s_{max}(G_1) = 2 \times 3 = 6$ and $|G_2| = 6 \geq \ell \times s_{max}(G_2) = 2 \times 3 = 6$. For the same reason, G_1 is further partitioned into *Group1* and *Group3* as shown in Figure 1a. Nevertheless, no matter which median coordinate in G_2 is chosen (i.e., $y = 3$ or $x = 8$), any further partitioning in G_2 has to fall back because given $y = 3$ as the median coordinate, for example, both generated group $g_1 = \{p_6, p_7, p_9\}$ and $g_2 = \{p_{10}, p_{11}, p_{12}\}$ result in $|g_1| = 3 < \ell \times s_{max}(g_1) = 4$ and $|g_2| = 3 < \ell \times s_{max}(g_2) = 4$. Likewise, fall-back would happen when the other median coordinate $x = 8$ is chosen.

The problem of Mondrian ℓ -diversity, however, is that it uses un-

published QI-SA correlation information to determine a partitioning to fall back. As illustrated in previous example, $s_{max}(g_1) = 2$ and $s_{max}(g_2) = 2$ are the QI-SA correlations which Mondrian uses to decide a fall-back of the partitioning from G_2 into g_1 and g_2 . As we see in Figure 1a, G_2 is published as *Group2* without partitioning. Nevertheless, such information $s_{max}(g_1) = 2$ and $s_{max}(g_2) = 2$ cannot be inferred out from the published *Group2*. Therefore, Mondrian ℓ -diversity violates the first condition of simulatable publishing (i.e., Definition 5), though it satisfies the second condition because no SA would be perturbed in the process.

To fix the problem, we adapt Mondrian by integrating worst-case eligibility test, that is, to reject a partitioning from G into G_1 and G_2 only if either $|G_1| < \ell \times s_{max}(G)$ or $|G_2| < \ell \times s_{max}(G)$ holds. Such strategy assures that each group G_1 (and G_2) is generated in such a “stringent” way that G_1 (and G_2) always achieves ℓ -diversity even in the worst case, i.e., when all $s_{max}(G)$ tuples with the most frequent SA value are partitioned into G_1 (and G_2). The reason is, for each G_1 , we have:

$$\frac{s_{max}(G_1)}{|G_1|} \leq \frac{s_{max}(G)}{|G_1|} \leq \frac{s_{max}(G)}{\ell \cdot s_{max}(G)} = \frac{1}{\ell}. \quad (1)$$

For the same reason, $\frac{s_{max}(G_2)}{|G_2|} \leq \frac{1}{\ell}$ holds as well.

To illustrate it, let us consider Figure 1b. After the first partitioning, both groups $G_1 = \{p_1, p_2, p_3, p_4, p_5, p_8\}$ and $G_2 = \{p_6, p_7, p_9, p_{10}, p_{11}, p_{12}\}$ use the worst-case eligibility test, namely $\ell \times s_{max}(G_1) = 6$ and $\ell \times s_{max}(G_2) = 6$, to decide to reject any further partitioning. Thus, G_1 and G_2 are published as *Group1* and *Group2* in Figure 1b. Unlike the original Mondrian, the required QI-SA correlations in the worst-case eligibility test can be inferred out from the groups subsequently (no matter whether the partitioning is rejected or not). For example, we can deduce the above-mentioned QI-SA correlations $s_{max}(G_1)$ and $s_{max}(G_2)$ from counting the SA values in the generated *Group1* and *Group2*. As we can see, the first condition of simulatable publishing is satisfied. The second condition of simulatable publishing is automatically satisfied because the Mondrian ℓ -diversity algorithm originally do not perturb SA. Hence, both conditions of simulatable publishing are satisfied.

6.1.2 In the Case of Hilb

We next discuss how to adapt the state-of-the-art ℓ -diversity algorithm Hilb to be simulatable publishing. The original Hilb ℓ -diversity works as follows. First, it transforms by Hilbert-curve the multi-dimensional QI space of a microdata table \mathbf{T} to a 1-D space Q_T , sorts all tuples in \mathbf{T} based on their Q_T values in ascending order, and bucketizes all the ordered tuples based on their SA values. Figure 1c shows a simple example of Hilb ℓ -diversity. All the 12

tuples are bucketized into 5 buckets (because there are 5 different SA values) and ordered ascendingly from p_1 to p_{12} based on their Q_T values (suppose $p_1 \rightarrow p_{12}$ is the ascending order).

Second, Hilb repetitively generates a group G_1 by picking up $|G_1|$ (initially to be ℓ) tuples from distinct $|G_1|$ buckets with lowest Q_T in G (initially to be \mathbf{T}), and suspends such partitioning when $|G| - |G_1| < \ell \times s_{max}(G \setminus G_1)$ holds. The previous partitioning resumes via incrementing $|G_1|$ by one each time until $|G_1| > m$ where m is the number of buckets. If $|G_1| > m$ holds, Hilb starts the fall-back procedure by restoring $|G_1|$ to ℓ , turns to pick up $|G_1|$ tuples from distinct $|G_1|$ buckets with the largest number of tuples, and produces the group G_1 if $|G| - |G_1| \geq \ell \times s_{max}(G \setminus G_1)$ holds. If this condition does not hold, the fall-back procedure continues via incrementing $|G_1|$ by one each time until a group G_1 satisfying the condition is able to be generated.

Refer to Figure 1c. Hilb starts with picking up $G_1 = \{p_1, p_3\}$ from 2 distinct buckets with lowest Q_T in $G = \mathbf{T}$ and generates *Group1* from G_1 because $|G| - |G_1| = 10 \geq \ell \times s_{max}(G \setminus G_1) = 2 \times 3 = 6$ holds. In the same fashion, Hilb produces *Group2* and *Group3*. After that, fall-back has to happen in partitioning $G = \{p_7, p_8, p_9, p_{10}, p_{11}, p_{12}\}$ because there exists no partition G_1 of $|G_1|$ ($< m = 5$) tuples with lowest Q_T such that $|G| - |G_1| \geq \ell \times s_{max}(G \setminus G_1)$ holds. For example, consider $G_1 = \{p_7, p_8, p_9\}$ where $|G| - |G_1| = 3 < \ell \times s_{max}(G \setminus G_1) = 2 \times 2 = 4$ holds. Thus, the fall-back procedure restores $|G_1| = 2$, picks up $\{p_7, p_9\}$ from $|G_1|$ distinct buckets with the largest number of tuples in G , and terminates with generating $G_1 = \{p_7, p_9\}$ to be *Group4*. Afterwards, *Group5* and *Group6* are published in the same fashion as *Group1*, *Group2* and *Group3*.

Similarly to the problem in the case of Mondrian, Hilb ℓ -diversity uses unpublished QI-SA correlation information to determine a partitioning to fall back, which violates the first condition of simulatable publishing. Return to the previous example to partition $G = \{p_7, p_8, p_9, p_{10}, p_{11}, p_{12}\}$. Fall-back procedure has to happen because the algorithm fails to generate a group G_1 with lowest Q_T given any $|G_1| < m = 5$. Among them all, consider the case of $|G_1| = 3$ for example. As illustrated previously, $s_{max}(G \setminus G_1) = 2$ is the only QI-SA correlation used to fail the attempt of generating $G_1 = \{p_7, p_8, p_9\}$. However, this piece of information cannot be calculated from the ultimate published groups in Figure 1c.

Likewise, we fix the problem by integrating the worst-case eligibility test into the original Hilb ℓ -diversity algorithm. Specifically, the worst-case eligibility test demands to reject a partition G_1 out of G only if $|G| - |G_1| < \ell \times s_{max}(G)$.

Consider Figure 1d for illustration, where *Group1*, *Group2* and *Group3* are generated in the same fashion as Hilb ℓ -diversity does in Figure 1c. However, the worst-case eligibility test requires not to further partition $G = \{p_7, p_8, p_9, p_{10}, p_{11}, p_{12}\}$, but instead publishes the entire G as *Group4*. The reason is that when $|G_1| = \ell = 2$, the worst-case eligibility test in our adapted algorithm rejects the partitioning because: 1) $(|G| - |G_1|) = 4 < \ell \times s_{max}(G) = 2 \times 3 = 6$ holds, and 2) there is no other $|G_1|$ such that $(|G| - |G_1|) \geq \ell \times s_{max}(G)$ holds. Herein, the only QI-SA correlation used by the worst-case eligibility test is $s_{max}(G)$, which can be obtained by counting the SA values in the generated *Group4*. Thereby, the first condition of simulatable publishing can be satisfied. Similarly to the case of Mondrian, the original Hilb ℓ -diversity algorithm does not perturb SA, such that the second condition of simulatable publishing is satisfied as well. Therefore, both conditions of simulatable publishing are satisfied.

6.2 Stratified Pick-up

Our first tool of worst-case eligibility test alone suffices to guarantee simulatable publishing. Nonetheless, a drawback of it is that

this may incur large sized groups and thus reduce the utility of published data. To work around it, we introduce our second tool: stratified pick-up for improving utility.

Stratified pick-up takes as input the anonymous groups from any simulatable publishing algorithm and tires to further partition each of these groups iteratively based solely on the distinctness of SA values. The design of this phase is principled on two objectives: 1) the algorithm should still satisfy simulatable publishing; and 2) each output group size should be minimized.

In particular, a simple solution to achieve these two objectives is to apply Anatomy [36] on each generated group from the algorithm with the worst-case eligibility test. Note that as discussed in Section 5.3, Anatomy does not perturb SA values and not use any QI-SA correlation beyond what will be eventually published. Thus, this solution satisfies simulatable publishing. Recall our example of Figure 1b. Stratified pick-up can be applied on the generated *Group1* and *Group2*, respectively. A possible output after stratified pick-up on *Group1* is $\{\{p_2, p_3\}, \{p_1, p_4\}, \{p_5, p_8\}\}$, which has obviously higher utility than the publication of *Group1*. Likewise, stratified pick-up can be applied on groups such as *Group4* in Figure 1d to improve the utility. Formally, we have the following theorem on the output group size after stratified pick-up.

THEOREM 6.1. *Each output group of stratified pick-up contains ℓ' tuples ($\ell' \in [\ell, 2\ell)$), each of which has a distinct SA value. If the algorithm which generates the input to stratified pick-up is simulatable publishing, then the algorithm with stratified pick-up is still simulatable publishing.*

Algorithm 1 Stratified Pick-up

```

1: DoneSet  $\leftarrow \emptyset$ .
2: InputSet  $\leftarrow$  anonymous groups from simulatable publishing algorithm.
3: repeat
4:    $G \leftarrow \text{InputSet}[i]$   $\triangleright$   $i$ -th element in InputSet.
5:   if  $\ell \leq \frac{|G|}{2}$  then
6:      $\{g_1, \dots, g_p\} \leftarrow \text{ANATOMY}(G, \ell)$ 
7:     DoneSet  $\leftarrow \text{DoneSet} \cup \{g_1, \dots, g_p\}$ .
8:   else
9:     DoneSet  $\leftarrow \text{DoneSet} \cup G$ .
10:  end if
11:  InputSet  $\leftarrow \text{InputSet} \setminus G$ .
12: until InputSet =  $\emptyset$ .
13: return DoneSet.

```

Details of stratified pick-up are shown in Algorithm 1. We test the condition $\ell \leq \frac{|G|}{2}$ in Line 5 because if a ℓ -diversity group G has size less than 2ℓ , G must have $|G|$ distinct SA values and cannot be further partitioned. Such minimized group G can be directly added to the output (Line 9).

The efficiency of stratified pick-up depends on Anatomy. Following the results from [36], the time complexity of stratified pick-up is $O(n)$ and the I/O cost is $O(\lambda)$, where n is the total number of tuples and λ is count of distinct SA values.

6.3 SP-Mondrian and SP-Hilb Algorithms

We are now ready to present two adapted simulatable publishing algorithms: SP-Mondrian and SP-Hilb, by integrating worst-case eligibility test and stratified pick-up into the original Mondrian and Hilb, respectively. Meanwhile, we will present their corresponding versions in the bucketization publishing scheme.

Algorithm 2 details the steps of SP-Mondrian ℓ -diversity algorithm. Line 4 implements the worst-case eligibility test. In Line 5,

MONDRIAN(G, k) denotes a function that invokes Mondrian k -anonymity algorithm [17] on the dataset G . Moreover, we design a version called SPBuck-Mondrian using the bucketization publishing scheme. All the steps are actually the same as SP-Mondrian except that in Line 6-Line 7, SPBuck-Mondrian keeps the original QI values of tuples in the generated G_1 and G_2 . Line 15 invokes the procedure of stratified pick-up as shown in Algorithm 1.

Algorithm 2 SP-Mondrian and SPBuck-Mondrian

```

1: DoneSet  $\leftarrow \emptyset$ . InputSet  $\leftarrow \{\mathbf{T}\}$ .
2: repeat
3:    $G \leftarrow$  the largest group in InputSet.
4:   if  $|G_1| \geq \ell \times s_{\max}(G)$  &&  $|G_2| \geq \ell \times s_{\max}(G)$  then
5:      $\{G_1, G_2\} \leftarrow$  MONDRIAN( $G, \ell \times s_{\max}(G)$ ).
6:     if Type = SPBuck-Mondrian then
7:       Replace generalized QI values in  $G_1$  and  $G_2$  with
       their original values.
8:       InputSet  $\leftarrow \{InputSet \setminus G\} \cup \{G_1, G_2\}$ .
9:     end if
10:  else
11:    InputSet  $\leftarrow InputSet \setminus G$ .
12:    DoneSet  $\leftarrow DoneSet \cup G$ .
13:  end if
14: until InputSet =  $\emptyset$ .
15: DoneSet  $\leftarrow$  STRATIFIED_PICK_UP(DoneSet,  $\ell$ ).
16: return DoneSet.

```

THEOREM 6.2. *SP-Mondrian and SPBuck-Mondrian are simulatable publishing ℓ -diversity algorithms.*

We now analyze the time complexity of Algorithm 2 as follows. Let $n = |\mathbf{T}|$ be the number of tuples in the microdata table \mathbf{T} . The number of iterations from Line 2 to Line 14 is at most $O(\log n)$. It is known from [17] that Line 5 takes $O(n \log n)$ time. Furthermore, Line 15 costs $O(n)$ as discussed earlier. Hence, the overall time complexity of Algorithm 2 is $O(n(\log n)^2)$.

However, we have to mention that since the condition in the worst-case eligibility test (Line 5) is more “stringent” than the original Mondrian, SP-Mondrian (and SPBuck-Mondrian) usually terminates earlier than the original version in terms of iterations (Line 2). Therefore, SP-Mondrian (and SPBuck-Mondrian) runs much faster than the original version in practice. We will demonstrate it later in Section 7.2.4.

Algorithm 3 describes SP-Hilb in details. Line 1-Line 4 processes the microdata \mathbf{T} by Hilbert curve transformation, sorting and bucketization as with the original Hilb ℓ -diversity. Line 6-Line 19 describes the procedure of generating a group G_1 from G . Line 7 and Line 20 implement the worst-case eligibility test and stratified pick-up, respectively. Unlike the original Hilb, there are no attempts of incrementing $|G_1|$ or fall-back procedure during generating a group G_1 . The reason is that once $|G| - |G_1| < \ell \times s_{\max}(G)$ holds, there exists no other $|G_1| \in [\ell, |G|]$ such that $|G| - |G_1| \geq \ell \times s_{\max}(G)$ is able to hold.

Like SP-Buck Mondrian, we also provide a simulatable publishing algorithm called SPBuck-Hilb in the bucketization publishing scheme. Line 12-Line 13 shows the only difference, that is, in the bucketization publishing scheme, the algorithm keeps the original QI values of tuples in each generated group.

THEOREM 6.3. *SP-Hilb and SPBuck-Hilb are simulatable publishing ℓ -diversity algorithms.*

The overall time complexity of Algorithm 3 is $O(n \log n)$, where $n = |\mathbf{T}|$. The reasons are: 1) Following the analysis in [9], Line 1

Algorithm 3 SP-Hilb and SPBuck-Hilb

```

1: DoneSet  $\leftarrow \emptyset$ .  $G \leftarrow \{\mathbf{T}\}$ .
2: Apply Hilbert curve to transform multi-dimensional QI space
  of  $G$  into 1-D dimensional space  $Q_T$ . Sort all the tuples in  $G$ 
  in ascending order of  $Q_T$ .
3: Split sorted tuples in  $m$  buckets based on SA values.
4: frontier  $\mathcal{F} \leftarrow$  set of first record in each bucket.
5: repeat
6:    $|G_1| \leftarrow \ell$ .
7:   if  $(|G| - |G_1|) < \ell \times s_{\max}(G)$  then
8:      $|G_1| \leftarrow |G|$ .
9:   end if
10:   $G_1 \leftarrow$  set of  $|G_1|$  tuples of  $\mathcal{F}$  with lowest  $Q_T$ .
11:   $G \leftarrow G \setminus G_1$ .
12:  if Type = SPBuck-Hilb then
13:    Keep the original QI values of tuples in  $G_1$ .
14:  else
15:    Generalize the QI values of tuples in  $G_1$  to an identical
    generalized value.
16:  end if
17:  Update  $\mathcal{F}$ .
18:  DoneSet  $\leftarrow DoneSet \cup G_1$ .
19: until  $G = \emptyset$ .
20: DoneSet  $\leftarrow$  STRATIFIED_PICK_UP(DoneSet,  $\ell$ ).
21: return DoneSet.

```

to Line 19 takes $O(n \log n)$ time; and 2) Line 20 costs $O(n)$ time as discussed in Section 6.2.

7. EXPERIMENTS

In this section, we describe our experimental setup, compare the data utility of our simulatable publishing algorithms with the existing ℓ -diversity algorithms, and evaluate the impact of our two tools: worst-case eligibility test and stratified pick-up.

7.1 Experimental Setup

7.1.1 Hardware

All experiments were conducted on a machine with Intel Core 2 Duo 2.6GHz CPU with 2GB RAM and Windows XP OS. All our algorithms were implemented using C++.

7.1.2 Dataset

We conducted the experiments on the Census dataset from <http://ipums.org> with attribute *Occupation* as SA, which has been extensively used as benchmarks in the literature. We followed the procedure in [36] to sample 300,000 tuples without replacement as our testing bed. To test generalization techniques, we adopted the generalization concept hierarchies used in [8] and [36].

7.1.3 Utility Measure

We adopted the same relative error measure proposed in [36]. Consider query workload of the form:

```

SELECT COUNT(*) FROM Dataset
WHERE  $pred(Q_1), \dots, pred(Q_{qd}), pred(S)$ 

```

where qd is the *query dimension* and $pred(Q_i)$ (resp. $pred(S)$) denotes the predicate of Q_i (resp. S) belonging to a range of randomly generated values in its domain. The cardinality of the range is determined by a parameter called *selectivity*. Let *Act* and *Est* be the query result from the microdata table \mathbf{T} and published table \mathbf{T}^* , respectively. The *relative error* is defined as $|Act - Est|/Act$. For

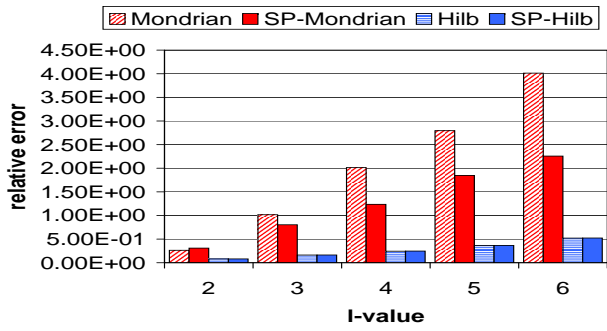


Figure 2: SP-Mondrian & SP-Hilb, vary ℓ

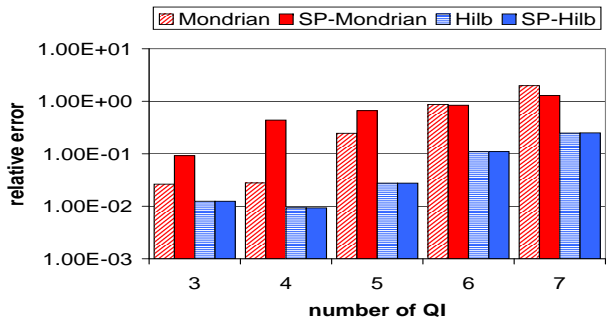


Figure 3: SP-Mondrian & SP-Hilb, vary q_i

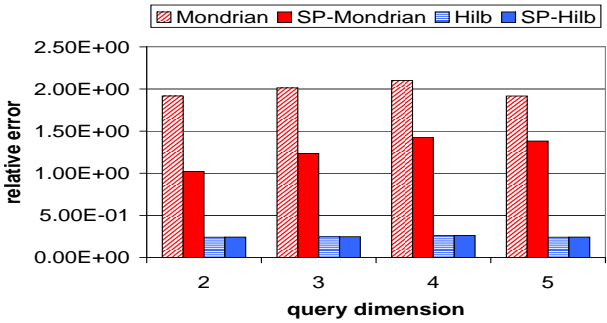


Figure 4: SP-Mondrian & SP-Hilb, vary q_d

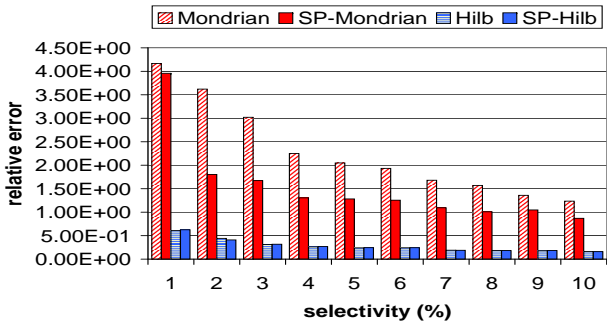


Figure 5: SP-Mondrian & SP-Hilb, vary s

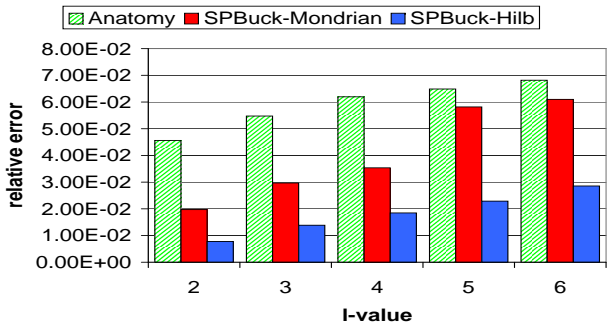


Figure 6: SPBuck-Mondrian & SPBuck-Hilb, vary ℓ

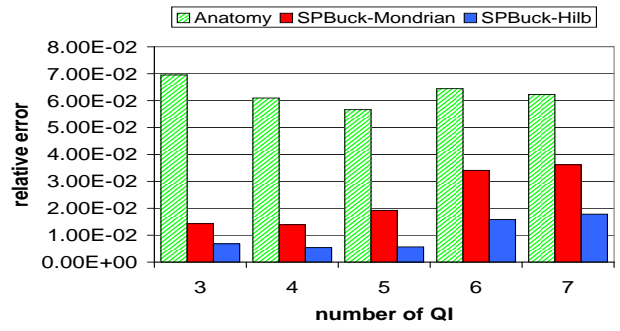


Figure 7: SPBuck-Mondrian & SPBuck-Hilb, vary q_i

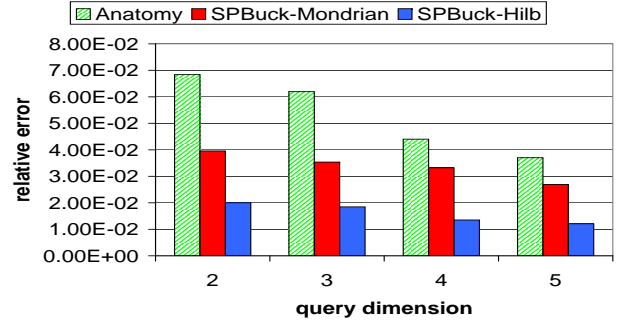


Figure 8: SPBuck-Mondrian & SPBuck-Hilb, vary q_d

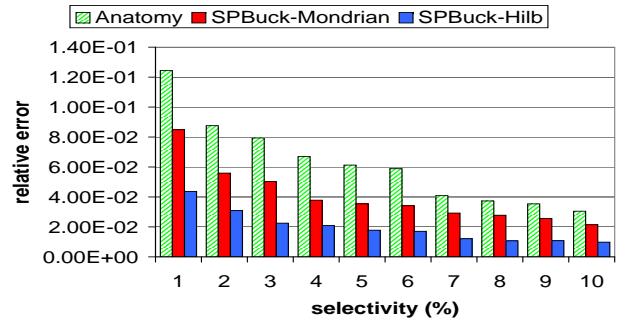


Figure 9: SPBuck-Mondrian & SPBuck-Hilb, vary s

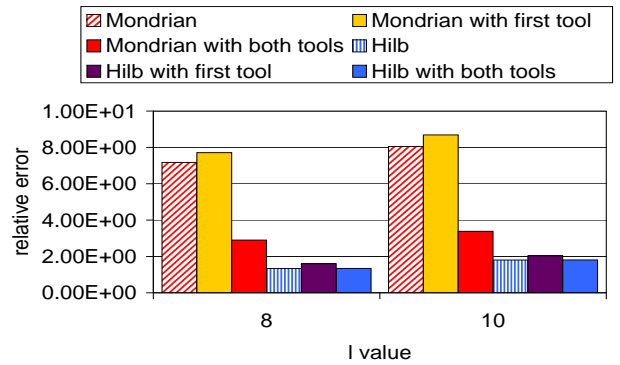


Figure 10: Utility impact testing of each tool individually

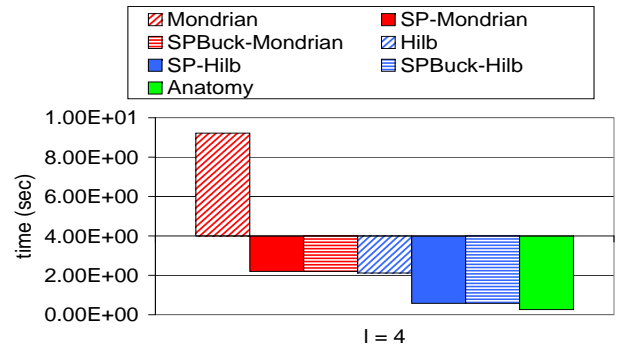


Figure 11: Time performance

each set of experiments, we ran a workload of 10000 queries, and calculated the average relative error as the utility measure.

7.2 Evaluation of SP-Mondrian and SP-Hilb

We first evaluate our simulatable algorithms: SP-Mondrian and SP-Hilb against the original Mondrian [18] and Hilb [9]. For the fairness of comparison, we adapt SP-Mondrian and SP-Hilb into bucketization scheme as discussed in Section 6.3, and compare them with the existing simulatable publishing algorithm Anatomy [36]. We then demonstrate the effect of our two generic tools: worst-case eligibility test and stratified pick-up individually. Finally, we test the efficiency. Note that we did not compare with MASK [34] because the authors' implementation <http://www.cse.ust.hk/~raywong/code/cred.zip> does not address the case when all the SA values are sensitive.

7.2.1 Utility Comparison with Mondrian and Hilb

We fix the number of QI $qi = 7$, query dimension $qd = 3$, selectivity $s = 5\%$. Figure 2 illustrates the utility of SP-Mondrian and SP-Hilb when varying ℓ value. It shows that both SP-Hilb and SP-Mondrian are not only able to eliminate the algorithm-based disclosure by worst-case eligibility testing, but also able to attain via stratified pick-up comparable or even better utility against Hilb, and Mondrian respectively.

We now set $\ell = 4$, and vary qi from 3 to 7. Figure 3 shows the impact of qi on the utility. As we see, SP-Hilb provides comparable utility with Hilb in all the cases. Whereas, when $qi \leq 5$, SP-Mondrian achieves less accuracy than Mondrian. Nonetheless, the accuracy difference is decreasing when qi increases. This is because Mondrian tends to generate larger groups when there are more QI attributes. This makes the utility improvement by stratified pick-up in SP-Mondrian more significant.

Figure 4 examines the utility of SP-Mondrian and SP-Hilb when query dimension qd ranges from 2 to 5, and $\ell = 4$, $qi = 7$, $sel = 5\%$. Figure 5 investigates the effect of selectivity sel on the utility when $\ell = 4$, $qi = 7$, $qd = 3$. One can see in both two figures that SP-Mondrian and SP-Hilb maintain comparable or significantly better utility (in the case of SP-Mondrian).

7.2.2 Utility Comparison with Anatomy

We now performed the evaluation on the bucketization publishing scheme, where SPBuck-Mondrian and SPBuck-Hilb were compared against Anatomy [36]. Recall that since Anatomy also satisfies simulatable publishing, the objective of adapting SP-Mondrian and SP-Hilb into SPBuck-Mondrian and SPBuck-Hilb is to provide better utility by taking into account QI-locality information. Using the same parameter settings as the previous generalization case, we conducted experiments as shown from Figures 6 to 9. As expected, both SPBuck-Mondrian and SPBuck-Hilb significantly outperform Anatomy in terms of utility.

7.2.3 Effects of Worst-case Eligibility Test and Stratified Pick-up

We previously integrated both tools: worst-case eligibility test and stratified pick-up. Now, we demonstrated the effect on utility of each tool separately. We set $qi = 7$, $qd = 3$, $sel = 5\%$, and tested the cases when $\ell = 8$ and $\ell = 10$. The reason why we chose higher ℓ values is only for the ease of illustration, because SP-Hilb achieves almost the same query accuracy as Hilb when $\ell \leq 7$.

To show the effect of our first tool (i.e., worst-case eligibility test), we tested in Figure 10 two simulatable publishing algorithms, which were adapted from Mondrian and Hilb via integrating only the first tool (i.e., without stratified pick-up). We compared them against the original Mondrian and Hilb. As expected, both adapted

algorithms achieves less utility than Mondrian and Hilb, respectively, which is the cost of eliminating algorithm-based disclosure.

To show the effect of our second tool (i.e., stratified pick-up), we compared algorithms integrated with both tools (i.e., worst-case eligibility test and stratified pick-up) against the previously developed algorithms with only the first tool. As we can see from Figure 10, stratified pick-up improves the utility, leading to comparable or even better utility than Hilb and Mondrian, respectively.

7.2.4 Time Performance

Figure 11 depicts the running time of Mondrian, SP-Mondrian, SPBuck-Mondrian, Hilb, SP-Hilb, SPBuck-Hilb and Anatomy when we set ℓ to 4. As expected, SP-Mondrian costs the same time as SPBuck-Mondrian and it is the same case between SP-Hilb and SPBuck-Hilb. SP-Mondrian/SPBuck-Mondrian runs much faster than the original Mondrian because, as we mentioned in Section 6, worst-case eligibility test usually leads to earlier termination than their original algorithms. Besides, recall Section 6 that the time complexity of SP-Hilb/SPBuck-Hilb is lower than Mondrian and SP-Mondrian/SPBuck-Mondrian, but higher than Anatomy. Therefore, SP-Hilb/SPBuck-Hilb is running faster than Mondrian and SP-Mondrian/SPBuck-Mondrian, but slower than Anatomy.

8. RELATED WORK

Since the introduction of k -anonymity [32] and ℓ -diversity [26], various privacy models have been proposed including (α, k) -anonymity [35], personalized privacy [37], t -closeness [20], (k, e) -anonymity [42], (ϵ, m) -anonymity [19], etc. To achieve these privacy models, researchers studied numerous data publishing algorithms [1–3, 8–12, 14, 16–18, 28, 30, 35, 36, 39, 42].

Orthogonal to the study mentioned above, there has been a large body of works on addressing the threats from external knowledge held by adversaries. [4, 6, 22, 27] considered the knowledge about an individual or relationship between individuals. [33] studied the presence of corruption. [21] studied the negative association rule. [29, 31] studied the privacy disclosure from learning whether a certain individual is present in the database or not.

Differential privacy proposed in [7] is able to eliminate algorithm-based disclosure by providing a new privacy model and developing a corresponding algorithm. [41] defined another new privacy model called p -safety to address the problem. Orthogonal to their work, the focus of our paper as mentioned in our introduction part is to develop generic tools to adapt the existing data publishing algorithms, such that these algorithms can be immune from algorithm-based disclosure.

[34] also studied the problem of algorithm-based disclosure by providing a new privacy model m -confidentiality and designing a new algorithm MASK to achieve it. However, we have shown in Section 2.2 that the MASK algorithm in [34] is still vulnerable to algorithm-based disclosure.

9. CONCLUSION

This paper addressed the problem of algorithm-based disclosure in privacy-preserving data publishing. We proposed a novel privacy model ASP to define the space of algorithm-based disclosure. Two necessary conditions and one sufficient condition of ASP were given as a tool to determine the vulnerabilities of existing algorithms. To eliminate algorithm-based disclosure, we proposed two generic tools for revising their design, and used them to generate two ℓ -diversity algorithms: SP-Mondrian and SP-Hilb by adapting the existing Mondrian and Hilb algorithms, respectively. We conducted extensive experiments to demonstrate the efficiency and utility of our algorithms.

10. ACKNOWLEDGMENTS

We would like to thank all the anonymous reviewers for their careful reading our paper and insightful comments.

11. REFERENCES

- [1] G. Aggarwal, T. Feder, K. Kenthapadi, S. Khuller, R. Panigrahy, D. Thomas, and A. Zhu. Achieving anonymity via clustering. In *PODS*, pages 153–162, 2006.
- [2] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu. Anonymizing tables. In *ICDT*, pages 246–258, 2005.
- [3] R. J. Bayardo and R. Agrawal. Data privacy through optimal k -anonymization. In *ICDE*, pages 217–228, 2005.
- [4] B. Chen, R. Ramakrishnan, and K. LeFevre. Privacy skyline: Privacy with multidimensional adversarial knowledge. In *VLDB*, pages 770–781, 2007.
- [5] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.
- [6] W. Du, Z. Teng, and Z. Zhu. Privacy-maxent: Integrating background knowledge in privacy quantification. In *SIGMOD*, pages 459–472, 2008.
- [7] C. Dwork. Differential privacy. In *ICALP*, pages 1–12, 2006.
- [8] B. C. M. Fung, K. Wang, and P. S. Yu. Top-down specialization for information and privacy preservation. In *ICDE*, pages 205–216, 2005.
- [9] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. Fast data anonymization with low information loss. In *VLDB*, pages 758–769, 2007.
- [10] G. Ghinita, Y. Tao, and P. Kalnis. On the anonymization of sparse high-dimensional data. In *ICDE*, pages 715–724, 2008.
- [11] T. Iwuchukwu and J. Naughton. k -anonymization as spatial indexing: Toward scalable and incremental anonymization. In *VLDB*, pages 746–757, 2007.
- [12] V. S. Iyengar. Transforming data to satisfy privacy constraints. In *KDD*, pages 279–288, 2002.
- [13] A. Kerckhoffs. La cryptographie militaire (military cryptography). *Journal des sciences militaires*, IX:5–83, 161–191, 1883.
- [14] D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. In *SIGMOD*, pages 217–228, 2006.
- [15] N. Koudas, D. Srivastava, T. Yu, and Q. Zhang. Distribution-based microdata anonymization. In *VLDB*, 2009.
- [16] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: efficient full-domain k -anonymity. In *SIGMOD*, pages 49–60, 2005.
- [17] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k -anonymity. In *ICDE*, pages 25–35, 2006.
- [18] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Workload-aware anonymization. In *KDD*, pages 277–286, 2006.
- [19] J. Li, Y. Tao, and X. Xiao. Preservation of proximity privacy in publishing numeric sensitive data. In *SIGMOD*, pages 473–486, 2008.
- [20] N. Li, T. Li, and S. Venkatasubramanian. t -closeness: Privacy beyond k -anonymity and ℓ -diversity. In *ICDE*, pages 106–115, 2007.
- [21] T. Li and N. Li. Injector: Mining background knowledge for data anonymization. In *ICDE*, pages 446–455, 2008.
- [22] T. Li, N. Li, and J. Zhang. Modeling and integrating background knowledge in data anonymization. In *ICDE*, 2009.
- [23] K. Liu and E. Terzi. Towards identity anonymization on graphs. In *SIGMOD*, 2009.
- [24] C.-Y. C. M. F. Mokbel and W. G. Aref. The new casper: Query processing for location services without compromising privacy. In *VLDB*, 2006.
- [25] A. Machanavajjhala, J. Gehrke, and M. Goetz. Data publishing against realistic adversaries. In *VLDB*, 2009.
- [26] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkatasubramanian. ℓ -diversity: Privacy beyond k -anonymity. 2006.
- [27] D. J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J. Halpern. Worst-case background knowledge for privacy-preserving data publishing. In *ICDE*, pages 126–135, 2007.
- [28] A. Meyerson and R. Williams. On the complexity of optimal k -anonymity. In *PODS*, pages 223–228, 2004.
- [29] M. E. Nergiz, M. Atzori, and C. Clifton. Hiding the presence of individuals from shared databases. In *SIGMOD*, pages 665–676, 2007.
- [30] H. Park and K. Shim. Approximate algorithms for k -anonymity. In *SIGMOD*, pages 67–78, 2007.
- [31] V. Rastogi, S. Hong, and D. Suciu. The boundary between privacy and utility in data publishing. In *VLDB*, pages 531–542, 2007.
- [32] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k -anonymity and its enforcement through generalization and suppression. Technical report, CMU, SRI, 1998.
- [33] Y. Tao, X. Xiao, J. Li, and D. Zhang. On anti-corruption privacy preserving publication. In *ICDE*, pages 725–734, 2008.
- [34] R. C. Wong, A. W. Fu, K. Wang, and J. Pei. Minimality attack in privacy-preserving data publishing. In *VLDB*, pages 543–554, 2007.
- [35] R. C. Wong, J. Li, A. W. Fu, and K. Wang. (α, k) -anonymity: An enhanced k -anonymity model for privacy-preserving data publishing. In *KDD*, pages 754–759, 2006.
- [36] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. In *VLDB*, pages 139–150, 2006.
- [37] X. Xiao and Y. Tao. Personalized privacy preservation. In *SIGMOD*, pages 229–240, 2006.
- [38] X. Xiao and Y. Tao. m -invariance: towards privacy preserving re-publication of dynamic datasets. In *SIGMOD*, pages 689–700, 2007.
- [39] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W. Fu. Utility-based anonymization using local recoding. In *KDD*, pages 785–790, 2006.
- [40] H. Yeye and J. Naughton. Anonymization of set-valued data via top-down, local generalization. In *Advances in Cryptography: Proceedings of Eurocrypt 2004*, 2009.
- [41] L. Zhang, S. Jajodia, and A. Brodsky. Information disclosure under realistic assumptions: Privacy versus optimality. In *CCS*, 2007.
- [42] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu. Aggregate query answering on anonymized tables. In *ICDE*, pages 116–125, 2007.