# Privacy-Preserving Data Publishing: A Survey of Recent Developments



BENJAMIN C. M. FUNG

*Concordia University, Montreal*

KE WANG

*Simon Fraser University, Burnaby*

RUI CHEN

*Concordia University, Montreal*

and

PHILIP S. YU

*University of Illinois at Chicago*



The collection of digital information by governments, corporations, and individuals has created tremendous opportunities for knowledge- and information-based decision making. Driven by mutual benefits, or by regulations that require certain data to be published, there is a demand for the exchange and publication of data among various parties. Data in its original form, however, typically contains sensitive information about individuals, and publishing such data will violate individual privacy. The current practice in data publishing relies mainly on policies and guidelines as to what types of data can be published and on agreements on the use of published data. This approach alone may lead to excessive data distortion or insufficient protection. *Privacy-preserving data publishing* (PPDP) provides methods and tools for publishing useful information while preserving data privacy. Recently, PPDP has received considerable attention in research communities, and many approaches have been proposed for different data publishing scenarios. In this survey, we will systematically summarize and evaluate different approaches to PPDP, study the challenges in practical data publishing, clarify the differences and requirements that distinguish PPDP from other related problems, and propose future research directions.

Categories and Subject Descriptors: H.2.7 [**Database Management**]: Database Administration—*Security, integrity, and protection*; H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms: Performance, Security

Additional Key Words and Phrases: Information sharing, privacy protection, anonymity, sensitive information, data mining

---

The research is supported in part by NSERC Discovery Grants (356065-2008).
Authors' addresses: Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC, Canada H3G 1M8; email: fung@ciise.concordia.ca.

DOI 10.1145/1749603.1749605 http://doi.acm.org/10.1145/1749603.1749605

**ACM Reference Format:**
Fung, B. C. M., Wang, K., Chen, R., and Yu, P. S. 2010. Privacy-Preserving data publishing: A survey of recent developments. ACM Comput. Surv. 42, 4, Article 14 (June 2010), 53 pages.
DOI = 10.1145/1749603.1749605 http://doi.acm.org/10.1145/1749603.1749605

## 1. INTRODUCTION

The collection of digital information by governments, corporations, and individuals has created tremendous opportunities for knowledge-based decision making. Driven by mutual benefits, or by regulations that require certain data to be published, there is a demand for the exchange and publication of data among various parties. For example, licensed hospitals in California are required to submit specific demographic data on every patient discharged from their facility [Carlisle et al. 2007]. In June 2004, the Information Technology Advisory Committee released a report entitled *Revolutionizing Health Care Through Information Technology* [President Information Technology Advisory Committee 2004]. One of its key points was to establish a nationwide system of electronic medical records that encourages sharing of medical knowledge through computer-assisted clinical decision support. Data publishing is equally ubiquitous in other domains. For example, Netflix, a popular online movie rental service, recently published a data set containing movie ratings of 500,000 subscribers, in a drive to improve the accuracy of movie recommendations based on personal preferences (New York Times, Oct. 2, 2006); AOL published a release of query logs but quickly removed it due to the reidentification of a searcher [Barbaro and Zeller 2006].

Detailed person-specific data in its original form often contains sensitive information about individuals, and publishing such data immediately violates individual privacy. The current practice primarily relies on policies and guidelines to restrict the types of publishable data and on agreements on the use and storage of sensitive data. The limitation of this approach is that it either distorts data excessively or requires a trust level that is impractically high in many data-sharing scenarios. For example, contracts and agreements cannot guarantee that sensitive data will not be carelessly misplaced and end up in the wrong hands.

A task of the utmost importance is to develop methods and tools for publishing data in a more hostile environment, so that the published data remains practically useful while individual privacy is preserved. This undertaking is called *privacy-preserving data publishing* (PPDP). In the past few years, research communities have responded to this challenge and proposed many approaches. While the research field is still rapidly developing, it is a good time to discuss the assumptions and desirable properties for PPDP, clarify the differences and requirements that distinguish PPDP from other related problems, and systematically summarize and evaluate different approaches to PPDP. This survey aims to achieve these goals.

### 1.1. Privacy-Preserving Data Publishing

A typical scenario for data collection and publishing is described in Figure 1. In the *data collection* phase, the *data publisher* collects data from *record owners* (e.g., Alice and Bob). In the *data publishing* phase, the data publisher releases the collected data to a data miner or to the public, called the *data recipient*, who will then conduct data mining on the published data. In this survey, data mining has a broad sense, not necessarily restricted to pattern mining or model building. For example, a hospital collects data from patients and publishes the patient records to an external medical center. In this example, the hospital is the data publisher, patients are record owners, and the

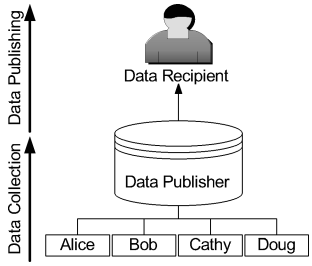


**Fig. 1**. Data collection and data publishing.

medical center is the data recipient. The data mining conducted at the medical center could be anything from a simple count of the number of men with diabetes to a sophisticated cluster analysis.

There are two models of data publishers [Gehrke 2006]. In the *untrusted* model, the data publisher is not trusted and may attempt to identify sensitive information from record owners. Various cryptographic solutions [Yang et al. 2005]; anonymous communications [Chaum 1981; Jakobsson et al. 2002]; and statistical methods [Warner 1965] were proposed to collect records anonymously from their owners without revealing the owners' identity. In the *trusted* model, the data publisher is trustworthy and record owners are willing to provide their personal information to the data publisher; however, the trust is not transitive to the data recipient. In this survey, we assume the trusted model of data publishers and consider privacy issues in the data publishing phase.

In practice, every data publishing scenario has its own assumptions and requirements of the data publisher, the data recipients, and the data publishing purpose. The following are several desirable assumptions and properties in practical data publishing:

*The nonexpert data publisher*. The data publisher is not required to have the knowledge to perform data mining on behalf of the data recipient. Any data mining activities have to be performed by the data recipient after receiving the data from the data publisher. Sometimes, the data publisher does not even know who the recipients are at the time of publication, or has no interest in data mining. For example, the hospitals in California publish patient records on the Web [Carlisle et al. 2007]. The hospitals do not know who the recipients are and how the recipients will use the data. The hospital publishes patient records because it is required by regulations [Carlisle et al. 2007] or because it supports general medical research, not because the hospital needs the result of data mining. Therefore, it is not reasonable to expect the data publisher to do more than anonymize the data for publication in such a scenario.

In other scenarios, the data publisher is interested in the data mining result, but lacks the in-house expertise to conduct the analysis, and hence outsources the data mining activities to some external data miners. In this case, the data mining task performed by the recipient is known in advance. In the effort to improve the quality of the data mining result, the data publisher could release a customized data set that preserves specific types of patterns for such a data mining task. Still, the actual data mining activities are performed by the data recipient, not by the data publisher.

*The data recipient could be an attacker.* In PPDP, one assumption is that the data recipient could also be an attacker. For example, the data recipient, say a drug research company, is a trustworthy entity; however, it is difficult to guarantee that all staff in the company is trustworthy as well. This assumption makes the PPDP problems and

solutions very different from the encryption and cryptographic approaches, in which only authorized and trustworthy recipients are given the private key for accessing the cleartext. A major challenge in PPDP is to simultaneously preserve both the privacy and information usefulness in the anonymous data.

*Publish data, not the data mining result*. PPDP emphasizes publishing data records about individuals (i.e., micro data). Clearly, this requirement is more stringent than publishing data mining results, such as classifiers, association rules, or statistics about groups of individuals. For example, in the case of the Netflix data release, useful information may be some type of associations of movie ratings. However, Netflix decided to publish data records instead of such associations because the participants, with data records, have greater flexibility in performing the required analysis and data exploration, such as mining patterns in one partition but not in other partitions; visualizing the transactions containing a specific pattern; trying different modeling methods and parameters, and so forth. The assumption for publishing data and not the data mining results, is also closely related to the assumption of a nonexpert data publisher. For example, Netflix does not know in advance how the interested parties might analyze the data. In this case, some basic "information nuggets" should be retained in the published data, but the nuggets cannot replace the data.

*Truthfulness at the record level*. In some data publishing scenarios, it is important that each published record corresponds to an existing individual in real life. Consider the example of patient records. The pharmaceutical researcher (the data recipient) may need to examine the actual patient records to discover some previously unknown side effects of the tested drug [Emam 2006]. If a published record does not correspond to an existing patient in real life, it is difficult to deploy data mining results in the real world. Randomized and synthetic data do not meet this requirement. Although an encrypted record corresponds to a real life patient, the encryption hides the semantics required for acting on the patient represented.

### 1.2. The Anonymization Approach

In the most basic form of PPDP, the data publisher has a table of the form

$$D(\text{Explicit\_Identifier, Quasi\_Identifier, Sensitive\_Attributes, Non-Sensitive\_Attributes}),$$

where Explicit_Identifier is a set of attributes, such as name and social security number (SSN), containing information that explicitly identifies record owners; Quasi_Identifier (QID) is a set of attributes that could potentially identify record owners; Sensitive_Attributes consists of sensitive person-specific information such as disease, salary, and disability status; and Non-Sensitive_Attributes contains all attributes that do not fall into the previous three categories [Burnett et al. 2003]. The four sets of attributes are disjoint. Most works assume that each record in the table represents a distinct record owner.

*Anonymization* [Cox 1980; Dalenius 1986] refers to the PPDP approach that seeks to hide the identity and/or the sensitive data of record owners, assuming that sensitive data must be retained for data analysis. Clearly, explicit identifiers of record owners must be removed. Even with all explicit identifiers being removed, Sweeney [2002a] showed a real-life privacy threat to William Weld, former governor of the state of Massachusetts. In Sweeney's example, an individual's name in a public voter list was linked with his record in a published medical database through the combination of zip code, date of birth, and sex, as shown in Figure 2. Each of these attributes does not uniquely identify a record owner, but their combination, called the *quasi-identifier* [Dalenius 1986], often singles out a unique or a small number of record

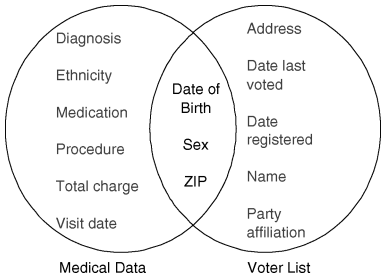


**Fig. 2**. Linking to reidentify record owner [Sweeney 2002a].

owners. According to Sweeney [2002a], 87% of the U.S. population had reported characteristics that likely made them unique based on only such quasi-identifiers.

In the above example, the owner of a record is reidentified by linking his quasi-identifier. To perform such *linking attacks*, the attacker needs two pieces of prior knowledge: the victim's record in the released data and the quasi-identifier of the victim. Such knowledge can be obtained by observation. For example, the attacker noticed that his boss was hospitalized, and therefore knew that his boss's medical record would appear in the released patient database. Also, it was not difficult for the attacker to obtain his boss's zip code, date of birth, and sex, which could serve as the quasi-identifier in linking attacks.

To prevent linking attacks, the data publisher provides an anonymous table,

$$T(QID', \text{Sensitive\_Attributes}, \text{Non-Sensitive\_Attributes}),$$

$QID'$ is an *anonymous* version of the original $QID$ obtained by applying *anonymization operations* to the attributes in $QID$ in the original table $D$. Anonymization operations hide some detailed information so that several records become indistinguishable with respect to $QID'$. Consequently, if a person is linked to a record through $QID'$, that person is also linked to all other records that have the same value for $QID'$, making the linking ambiguous. Alternatively, anonymization operations could generate synthetic data table $T$ based on the statistical properties of the original table $D$, or add noise to the original table $D$. The *anonymization problem* is to produce an anonymous $T$ that satisfies a given privacy requirement determined by the chosen privacy model and to retain as much data utility as possible. An *information metric* is used to measure the utility of an anonymous table. Note that the Non-Sensitive_Attributes are published if they are important to the data mining task.

### 1.3. The Scope

A closely related research area is *privacy-preserving data mining* [Aggarwal and Yu 2008c]. The term, privacy-preserving data mining (PPDM), emerged in 2000 [Agrawal and Srikant 2000]. The initial idea of PPDM *was* to extend traditional data mining techniques to work with the data modified to mask sensitive information. The key issues were how to modify the data and how to recover the data mining result from the modified data. The solutions were often tightly coupled with the data mining algorithms under consideration. In contrast, PPDP may not necessarily be tied to a specific data mining task, and the data mining task may be unknown at the time of data publishing. Furthermore, some PPDP solutions emphasize preserving the data truthfulness at the record level as discussed earlier, but often PPDM solutions do not preserve such a property. In recent years, the term "PPDM" has evolved to cover many other

privacy research problems, even though some of them may not directly relate to data mining.

Another related area is the study of the *noninteractive* query model in statistical disclosure control [Adam and Wortman 1989; Brand 2002], in which the data recipients can submit one query to the system. This type of noninteractive query model may not fully address the information needs of data recipients because, in some cases, it is very difficult for a data recipient to accurately construct a query for a data mining task in one shot. Consequently, there are a series of studies on the *interactive* query model [Blum et al. 2005; Dwork 2008; Dinur and Nissim 2003], in which the data recipients, unfortunately including attackers, can submit a sequence of queries based on previously received query results. One limitation of any privacy-preserving query system is that it can only answer a sublinear number of queries in total; otherwise, an attacker (or a group of corrupted data recipients) will be able to reconstruct all but $1 - o(1)$ fraction of the original data [Blum et al. 2008], which is a very strong violation of privacy. When the maximum number of queries is reached, the system must be closed to avoid privacy leak. In the case of a noninteractive query model, the attacker can issue an unlimited number of queries and, therefore, a noninteractive query model cannot achieve the same degree of privacy defined by the interactive model. This survey focuses mainly on the noninteractive query model, but the interactive query model will also be briefly discussed in Section 8.1.

In this survey, we review recent work on anonymization approaches to privacy-preserving data publishing (PPDP) and provide our own insights into this topic. There are several fundamental differences between the recent work on PPDP and the previous work proposed by the official statistics community. Recent work on PPDP considers background attacks, inference of sensitive attributes, generalization, and various notions of data utility measures, but the work of the official statistics community doesnot. The term "privacy-preserving data publishing" has been widely adopted by the computer science community to refer to the recent work discussed in this survey article. In fact, the official statistics community seldom uses the term "privacy-preserving data publishing" to refer to their work. In this survey, we do not intend to provide a detailed coverage of the official statistics methods because some decent surveys already exist [Adam and Wortman 1989; Domingo-Ferrer 2001; Moore 1996; Zayatz 2007].

We focus on several key issues in PPDP: attack models and privacy models (Section 2); anonymization operations (Section 3); information metrics (Section 4); and anonymization algorithms (Section 5). Most research focuses on a single release from a single data publisher. We also consider the work for more practical scenarios (Section 6) that deals with dynamic data, multiple releases, and multiple publishers. Much real-world data is nonrelational. We study some recently proposed anonymization techniques for transaction data, moving objects data, and textual data (Section 7). Then, we briefly discuss other privacy-preserving techniques that are orthogonal to PPDP. (Section 8). Finally, we conclude with a summary and discussion of future research directions (Section 9).

## 2. ATTACK MODELS AND PRIVACY MODELS

What is privacy protection? Dalenius [1977] provided a very stringent definition: access to the published data should not enable the attacker to learn anything extra about any target victim compared to no access to the database, even with the presence of any attacker's background knowledge obtained from other sources. Dwork [2006] showed that such absolute privacy protection is impossible due to the presence of background knowledge. Suppose the age of an individual is sensitive information. Assume an attacker knows that Alice's age is 5 years younger than the average age of American

**Table I.** Privacy Models

| Privacy Model | Attack Model | | | |
|---|---|---|---|---|
| | Record Linkage | Attribute Linkage | Table Linkage | Probabilistic Attack |
| $k$-Anonymity | ✓ | | | |
| MultiR $k$-Anonymity | ✓ | | | |
| $\ell$-Diversity | ✓ | ✓ | | |
| Confidence Bounding | | ✓ | | |
| $(\alpha, k)$-Anonymity | ✓ | ✓ | | |
| $(X, Y)$-Privacy | ✓ | ✓ | | |
| $(k, e)$-Anonymity | | ✓ | | |
| $(\epsilon, m)$-Anonymity | | ✓ | | |
| Personalized Privacy | | ✓ | | |
| $t$-Closeness | | ✓ | | ✓ |
| $\delta$-Presence | | | ✓ | |
| $(c, t)$-Isolation | ✓ | | | ✓ |
| $\epsilon$-Differential Privacy | | | ✓ | ✓ |
| $(d, \gamma)$-Privacy | | | ✓ | ✓ |
| Distributional Privacy | | | ✓ | ✓ |

women. If the attacker has access to a statistical database that discloses the average age of American women, then Alice's privacy is considered compromised according to Dalenius' definition, regardless whether or not Alice's record is in the database [Dwork 2006].

Most literature on PPDP considers a more relaxed, more practical notion of privacy protection by assuming the attacker has limited background knowledge. Below, the term "victim" refers to the record owner targeted by the attacker. We can broadly classify privacy models into two categories based on their attack principles.

The first category considers that a privacy threat occurs when an attacker is able to link a record owner to a record in a published data table, to a sensitive attribute in a published data table, or to the published data table itself. We call these *record linkage*, *attribute linkage*, and *table linkage*, respectively. In all three types of linkages, we assume that the attacker knows the *QID* of the victim. In record and attribute linkages, we further assume that the attacker knows that the victim's record is in the released table, and seeks to identify the victim's record and/or sensitive information from the table. In table linkage, the attack seeks to determine the presence or absence of the victim's record in the released table. A data table is considered to be privacy-preserving if it can effectively prevent the attacker from successfully performing these linkages. Sections 2.1 to 2.3 study this category of privacy models.

The second category aims at achieving the *uninformative principle* [Machanavajjhala et al. 2006]: The published table should provide the attacker with little additional information beyond the background knowledge. If the attacker has a large variation between the prior and posterior beliefs, we call it the *probabilistic attack*. Many privacy models in this family do not explicitly classify attributes in a data table into *QID* and Sensitive_Attributes, but some of them could also thwart the sensitive linkages in the first category, so the two categories overlap. Section 2.4 studies this family of privacy models. Table I summarizes the attack models addressed by the privacy models.

### 2.1. Record Linkage

In the attack of *record linkage*, some value *qid* on *QID* identifies a small number of records in the released table $T$, called a *group*. If the victim's *QID* matches the value *qid*, the victim is vulnerable to being linked to the small number of records in the group. In this case, the attacker faces only a small number of possibilities for the

**Table II.** Examples Illustrating Various Attacks

(a) Patient table

| Job | Sex | Age | Disease |
|---|---|---|---|
| Engineer | Male | 35 | Hepatitis |
| Engineer | Male | 38 | Hepatitis |
| Lawyer | Male | 38 | HIV |
| Writer | Female | 30 | Flu |
| Writer | Female | 30 | HIV |
| Dancer | Female | 30 | HIV |
| Dancer | Female | 30 | HIV |

(b) External table

| Name | Job | Sex | Age |
|---|---|---|---|
| Alice | Writer | Female | 30 |
| Bob | Engineer | Male | 35 |
| Cathy | Writer | Female | 30 |
| Doug | Lawyer | Male | 38 |
| Emily | Dancer | Female | 30 |
| Fred | Engineer | Male | 38 |
| Gladys | Dancer | Female | 30 |
| Henry | Lawyer | Male | 39 |
| Irene | Dancer | Female | 32 |

(c) 3-anonymous patient table

| Job | Sex | Age | Disease |
|---|---|---|---|
| Professional | Male | [35-40) | Hepatitis |
| Professional | Male | [35-40) | Hepatitis |
| Professional | Male | [35-40) | HIV |
| Artist | Female | [30-35) | Flu |
| Artist | Female | [30-35) | HIV |
| Artist | Female | [30-35) | HIV |
| Artist | Female | [30-35) | HIV |

(d) 4-anonymous external table

| Name | Job | Sex | Age |
|---|---|---|---|
| Alice | Artist | Female | [30-35) |
| Bob | Professional | Male | [35-40) |
| Cathy | Artist | Female | [30-35) |
| Doug | Professional | Male | [35-40) |
| Emily | Artist | Female | [30-35) |
| Fred | Professional | Male | [35-40) |
| Gladys | Artist | Female | [30-35) |
| Henry | Professional | Male | [35-40) |
| Irene | Artist | Female | [30-35) |

victim's record, and with the help of additional knowledge, there is a chance that the attacker could uniquely identify the victim's record from the group.

*Example* 2.1. Suppose that a hospital wants to publish the patient records in Table II(a) to a research center. Suppose that the research center has access to the external table Table II(b) and knows that every person with a record in Table II(b) has a record in Table II(a). Joining the two tables on the common attributes *Job*, *Sex*, and *Age* may link the identity of a person to his/her *Disease*. For example, *Doug*, a male lawyer who is 38 years old, is identified as an *HIV* patient by $qid = \langle Lawyer, Male, 38\rangle$ after the join.

*k-Anonymity*. To prevent record linkage through *QID*, Samarati and Sweeney [1998a, 1998b] proposed the notion of *k-anonymity*: if one record in the table has some value *qid*, at least $k-1$ other records also have the value *qid*. In other words, the minimum group size on *QID* is at least $k$. A table satisfying this requirement is called *k-anonymous*. In a $k$-anonymous table, each record is indistinguishable from at least $k-1$ other records with respect to *QID*. Consequently, the probability of linking a victim to a specific record through *QID* is at most $1/k$.

$k$-anonymity cannot be replaced by the privacy models in attribute linkage (Section 2.2). Consider a table $T$ that contains no sensitive attributes (such as the voter list in Figure 2). An attacker could possibly use the *QID* in $T$ to link to the sensitive information in an external source. A $k$-anonymous $T$ can still effectively prevent this type of record linkage without considering the sensitive information. In contrast, the privacy models in attribute linkage assume the existence of sensitive attributes in $T$.

*Example* 2.2. Table II(c) shows a 3-anonymous table by generalizing $QID = \{Job, Sex, Age\}$ from Table II(a) using the taxonomy trees in Figure 3. It has two distinct groups on *QID*, namely $\langle Professional, Male, [35-40)\rangle$ and $\langle Artist, Female, [30-35)\rangle$. Since each group contains at least 3 records, the table is 3-anonymous. If we link the records in Table II(b) to the records in Table II(c) through *QID*, each record is linked to either no record or at least 3 records in Table II(c).

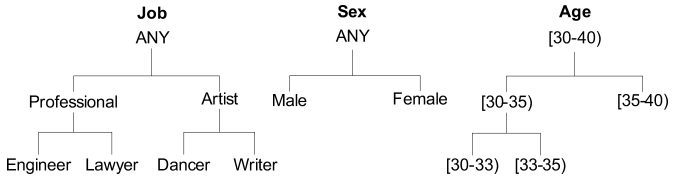


**Fig. 3**. Taxonomy trees for *Job*, *Sex*, *Age*.

The $k$-anonymity model assumes that *QID* is known to the data publisher. Most work considers a single *QID* containing all attributes that can be potentially used in the quasi-identifier. The more attributes included in *QID*, the more protection $k$-anonymity would provide. On the other hand, this also implies that more distortion is needed to achieve $k$-anonymity because the records in a group have to agree on more attributes. To address this issue, Fung et al. [2005, 2007] allow the specification of multiple *QID*s, assuming that the data publisher knows the potential *QID*s for record linkage. The following example illustrates the use of this specification.

*Example* 2.3. The data publisher wants to publish a table $T(A, B, C, D, S)$, where $S$ is the sensitive attribute, and knows that the data recipient has access to previously published tables $T1(A, B, X)$ and $T2(C, D, Y)$, where $X$ and $Y$ are attributes not in $T$. To prevent linking the records in $T$ to the information on $X$ or $Y$, the data publisher can specify $k$-anonymity on $QID_1 = \{A, B\}$ and $QID_2 = \{C, D\}$ for $T$. This means that each record in $T$ is indistinguishable from a group of at least $k$ records with respect to $QID_1$ and is indistinguishable from a group of at least $k$ records with respect to $QID_2$. The two groups are not necessarily the same. Clearly, this requirement is implied by $k$-anonymity on $QID = \{A, B, C, D\}$, but having $k$-anonymity on both $QID_1$ and $QID_2$ does not imply $k$-anonymity on *QID*.

Specifying multiple *QID*s is practical only if the data publisher knows how the attacker might perform the linking. Nevertheless, often the data publisher does not have such information. A wrong decision may cause higher privacy risks or higher information loss. Later, we discuss the dilemma and implications of choosing attributes in *QID*. In the presence of multiple *QID*s, some *QID*s may be redundant and can be removed by the following subset property:

*Observation* 2.1 (*Subset Property*). Let $QID' \subseteq QID$. If a table $T$ is $k$-anonymous on *QID*, then $T$ is also $k$-anonymous on $QID'$. In other words, $QID'$ is *covered by QID*, so $QID'$ can be removed from the privacy requirement [Fung et al. 2005, 2007; LeFevre et al. 2005].

The $k$-anonymity model assumes that each record represents a distinct individual. If several records in a table represent the same record owner, a group of $k$ records may represent fewer than $k$ record owners, and the record owner may be underprotected. The following example illustrates this point.

*Example* 2.4. A record in the table $Inpatient(Pid, Job, Sex, Age, Disease)$ represents that a patient identified by *Pid* has *Job*, *Sex*, *Age*, and *Disease*. A patient may have several records, one for each disease. In this case, $QID = \{Job, Sex, Age\}$ is not a key and $k$-anonymity on *QID* fails to ensure that each group on *QID* contains at least $k$ (*distinct*) patients. For example, if each patient has at least 3 diseases, a group of $k$ records will involve no more than $k/3$ patients.

$(X, Y)$-*Anonymity*. To address this problem of $k$-anonymity, Wang and Fung [2006] proposed the notion of $(X, Y)$-anonymity, where $X$ and $Y$ are disjoint sets of attributes.

*Definition* 2.1. Let $x$ be a value on $X$. The *anonymity* of $x$ with respect to $Y$, denoted $a_Y(x)$, is the number of distinct values on $Y$ that co-occur with $x$. Let $A_Y(X) = min\{a_Y(x) \mid x \in X\}$. A table $T$ *satisfies* the $(X, Y)$-anonymity for some specified integer $k$ if $A_Y(X) \geq k$.

$(X, Y)$-anonymity specifies that each value on $X$ is linked to at least $k$ *distinct* values on $Y$. The $k$-anonymity is the special case where $X$ is the *QID* and $Y$ is a key in $T$ that uniquely identifies record owners. $(X, Y)$-anonymity provides a uniform and flexible way to specify different types of privacy requirements. If each value on $X$ describes a group of record owners (e.g., $X = \{Job, Sex, Age\}$) and $Y$ represents the sensitive attribute (e.g., $Y = \{Disease\}$), this means that each group is associated with a diverse set of sensitive values, making it difficult to infer a specific sensitive value. The next example shows the usefulness of $(X, Y)$-anonymity for modeling $k$-anonymity in the case that several records may represent the same record owner.

*Example* 2.5. Continue from Example 2.4. With $(X, Y)$-anonymity, we specify $k$-anonymity with respect to *patients* by letting $X = \{Job, Sex, Age\}$ and $Y = \{Pid\}$. That is, each $X$ group is linked to at least $k$ distinct patient IDs, therefore, $k$ distinct patients.

*MultiRelational k-Anonymity*. Most work on $k$-anonymity focuses on anonymizing a single data table; however, a real-life database usually contains multiple relational tables. Nergiz et al. [2007] proposed a privacy model called *MultiR k-anonymity* to ensure $k$-anonymity on multiple relational tables. Their model assumes that a relational database contains a person-specific table $PT$ and a set of tables $T_1, \ldots, T_n$, where $PT$ contains a person identifier $Pid$ and some sensitive attributes, and $T_i$, for $1 \leq i \leq n$, contains some foreign keys, some attributes in *QID*, and sensitive attributes. The general privacy notion is to ensure that for each record owner $o$ contained in the join of all tables $PT \bowtie T_1 \bowtie \cdots \bowtie T_n$, there exists at least $k - 1$ other record owners who share the same *QID* with $o$. It is important to emphasize that the $k$-anonymization is applied at the *record owner* level, not at the *record* level in traditional $k$-anonymity. This idea is similar to $(X, Y)$-anonymity, where $X = QID$ and $Y = \{Pid\}$.

*Dilemma in choosing QID*. One challenge faced by a data publisher is how to classify the attributes in a data table into three disjoint sets: *QID*, Sensitive_Attributes, and Non-Sensitive_Attributes. In principle, *QID* should contain an attribute $A$ if the attacker could potentially obtain $A$ from other external sources. After the *QID* is determined, remaining attributes are grouped into Sensitive_Attributes and Non-Sensitive_Attributes based on their sensitivity. There is no definite answer to the question of how a data publisher can determine whether or not an attacker can obtain an attribute $A$ from some external sources, but it is important to understand the implications of a misclassification: misclassifying an attribute $A$ into Sensitive_Attributes or Non-Sensitive_Attributes may compromise another sensitive attribute $S$ because an attacker may obtain $A$ from other sources and then use $A$ to perform record linkage or attribute linkage on $S$. On the other hand, misclassifying a sensitive attribute $S$ into *QID* may directly compromise sensitive attribute $S$ of some target victim because an attacker may use attributes in $QID - S$ to perform attribute linkage on $S$. Furthermore, incorrectly including $S$ in *QID* causes unnecessary information loss due to the curse of dimensionality [Aggarwal 2005].

Motwani and Xu [2007] presented a method to determine the minimal set of quasi-identifiers for a data table $T$. The intuition is to identify a minimal set of attributes from $T$ that has the ability to (almost) distinctly identify a record and the ability to separate two data records. Nonetheless, the minimal set of *QID* does not imply

the most appropriate privacy protection setting because the method does not consider what attributes the attacker could potentially have. If the attacker can obtain a bit more information about the target victim beyond the minimal set, then he may able to conduct a successful linking attack. The choice of *QID* remains an open issue.

$k$-anonymity, $(X, Y)$-anonymity, and MultiR $k$-anonymity prevent record linkage by hiding the record of a victim in a large group of records with the same *QID*. However, if most records in a group have similar values on a sensitive attribute, the attacker can still associate the victim to her sensitive value without having to identify her record. This situation is illustrated in Table II(c), which is 3-anonymous. For a victim matching $qid = \langle Artist, Female, [30{-}35)\rangle$, the confidence of inferring that the victim has *HIV* is 75% because 3 out of the 4 records in the group have *HIV*. Though $(X, Y)$-anonymity requires that each $X$ group is linked to at least $k$ distinct $Y$ values, if some $Y$ values occur more frequently than others, there is a higher confidence of inferring the more frequent values. This leads us to the next family of privacy models for preventing this type of attribute linkage.

### 2.2. Attribute Linkage

In the attack of *attribute linkage*, the attacker may not precisely identify the record of the target victim, but could infer his/her sensitive values from the published data $T$, based on the set of sensitive values associated to the group that the victim belongs to. In case some sensitive values predominate in a group, a successful inference becomes relatively easy even if $k$-anonymity is satisfied. Clifton [2000] suggested eliminating attribute linkages by limiting the released data size. Limiting data size may not be desirable if data records such as *HIV* patient data, are valuable and are difficult to obtain. Several other approaches have been proposed to address this type of threat. The general idea is to diminish the correlation between *QID* attributes and sensitive attributes.

*Example* 2.6. From Table II(a), an attacker can infer that all female dancers at age 30 have *HIV*, i.e., $\langle Dancer, Female, 30\rangle \rightarrow HIV$ with 100% confidence. Applying this knowledge to Table II(b), the attacker can infer that *Emily* has *HIV* with 100% confidence provided that *Emily* comes from the same population in Table II(a).

*$\ell$-Diversity*. Machanavajjhala et al. [2006, 2007] proposed the diversity principle, called *$\ell$-diversity*, to prevent attribute linkage. The $\ell$-diversity requires every *qid* group to contain at least $\ell$ "well-represented" sensitive values. A similar idea was previously discussed in Ohrn and Ohno-Machado [1999]. There are several instantiations of this principle, which differ in the definition of being well-represented. The simplest understanding of "well-represented" is to ensure that there are at least $\ell$ distinct values for the sensitive attribute in each *qid* group. This *distinct $\ell$-diversity* privacy model (also known as $p$-sensitive $k$-anonymity [Truta and Bindu 2006]) automatically satisfies $k$-anonymity, where $k = \ell$, because each *qid* group contains at least $\ell$ records. Distinct $\ell$-diversity cannot prevent probabilistic inference attacks because some sensitive values are naturally more frequent than others in a group, enabling an attacker to conclude that a record in the group is very likely to have those values. For example, *Flu* is more common than *HIV*. This motivates the following two stronger notions of $\ell$-diversity.

A table is *entropy $\ell$-diverse* if for every *qid* group

$$-\sum_{s\in S} P(qid, s)log(P(qid, s)) \geq log(\ell) \tag{1}$$

where $S$ is a sensitive attribute, and $P(qid, s)$ is the fraction of records in a *qid* group having the sensitive value $s$. The left-hand side, called the entropy of the sensitive attribute, has the property that more evenly distributed sensitive values in a *qid* group produce a larger value. Therefore, a large threshold value $\ell$ implies less certainty of inferring a particular sensitive value in a group. Note that the inequality does not depend on the choice of the log base.

*Example* 2.7. Consider Table II(c). For the first group $\langle$*Professional, Male*, [35 − 40)$\rangle$, $-\frac{2}{3}log\frac{2}{3} - \frac{1}{3}log\frac{1}{3} = log(1.9)$, and for the second group $\langle$*Artist, Female*, [30−35)$\rangle$, $-\frac{3}{4}log\frac{3}{4} - \frac{1}{4}log\frac{1}{4} = log(1.8)$. So the table satisfies entropy $\ell$-diversity if $\ell \leq 1.8$.

One limitation of entropy $\ell$-diversity is that it does not provide a probability based risk measure, which tends to be more intuitive to the human data publisher. For example, in Table II(c), being entropy 1.8-diverse in Example 2.7 does not convey the risk level that the attacker has 75% probability of success to infer *HIV* where 3 out of the 4 record owners in the *qid* group have *HIV*. Also, it is difficult to specify different protection levels based on varied sensitivity and frequency of sensitive values.

The recursive $(c, \ell)$-diversity makes sure that the most frequent value does not appear too frequently, and that the less frequent values do not appear too rarely. Let $m$ be the number of sensitive values in a *qid* group. Let $f_i$ denote the frequency of the $i^{th}$ most frequent sensitive value in a *qid* group. A *qid* group is $(c, \ell)$-*diverse* if the frequency of the most frequent sensitive value is less than the sum of the frequencies of the $m - \ell + 1$ least frequent sensitive values multiplying by some publisher-specified constant $c$, that is, $f_1 < c \sum_{i=\ell}^{m} f_i$. The intuition is that even if the attacker excludes some possible sensitive values of a victim by applying background knowledge, the remaining ones remain hard to infer. A table is considered to have recursive $(c, \ell)$-diversity if all of its groups have $(c, \ell)$-diversity.

Machanavajjhala et al. [2006, 2007] also presented two other instantiations, called *positive disclosure-recursive* $(c, \ell)$*-diversity* and *negative/positive disclosure-recursive* $(c, \ell)$*-diversity* to capture the attacker's background knowledge. Suppose a victim is in a *qid* group that contains three different sensitive values {*Flu, Cancer*, *HIV*}, and suppose the attacker knows that the victim has no symptom of having a flu. Given this piece of background knowledge, the attacker can eliminate *Flu* from the set of candidate-sensitive values of the victim. Martin et al. [2007] proposed a language to capture this type of background knowledge and to represent the knowledge as $k$ units of information. Furthermore, the language could capture the type of implication knowledge. For example, given that Alice, Bob, and Cathy have flu, the attacker infers that Doug is very likely to have flu, too, because all four of them live together. This implication is considered to be one unit of information. Given an anonymous table $T$ and $k$ units of background knowledge, Martin et al. [2007] estimated the maximum disclosure risk of $T$, which is the probability of the most likely predicted sensitive value assignment of any record owner in $T$.

$\ell$-diversity has the limitation of implicitly assuming that each sensitive attribute takes values uniformly over its domain, that is, the frequencies of the various values of a confidential attribute are similar. When this is not the case, achieving $\ell$-diversity may cause a large data utility loss. Consider a data table containing data of 1000 patients on some *QID* attributes and a single sensitive attribute *HIV* with two possible values, *Yes* or *No*. Assume that there are only 5 patients with *HIV* = *Yes* in the table. To achieve $k$-anonymity with $k = \ell$, at least one patient with HIV is needed in each *qid* group; therefore, at most 5 groups can be formed [Domingo-Ferrer and Torra 2008]. Enforcing $k$-anonymity with $k = \ell$ may lead to high information loss in this case.

*Confidence bounding*. Wang et al. [2005, 2007] considered bounding the confidence of inferring a sensitive value from a *qid* group by specifying one or more *privacy templates* of the form, $\langle QID \rightarrow s, h\rangle$; $s$ is a sensitive value, $QID$ is a quasi-identifier, and $h$ is a threshold. Let $Conf(QID \rightarrow s)$ be $max\{conf(qid \rightarrow s)\}$ over all *qid* groups on $QID$, where $conf(qid \rightarrow s)$ denotes the percentage of records containing $s$ in the *qid* group. A table satisfies $\langle QID \rightarrow s, h\rangle$ if $Conf(QID \rightarrow s) \leq h$. In other words, $\langle QID \rightarrow s, h\rangle$ bounds the attacker's confidence of inferring the sensitive value $s$ in any group on $QID$ to the maximum $h$.

For example, with $QID = \{Job, Sex, Age\}$, $\langle QID \rightarrow HIV, 10\%\rangle$ states that the confidence of inferring *HIV* from any group on $QID$ is no more than 10%. For the data in Table II(c), this privacy template is violated because the confidence of inferring *HIV* is 75% in the group $\{Artist, Female, [30{-}35)\}$.

The confidence measure has two advantages over recursive $(c, \ell)$-diversity and entropy $\ell$-diversity. First, the confidence measure is more intuitive because the risk is measured by the probability of inferring a sensitive value. The data publisher relies on this intuition to specify the acceptable maximum confidence threshold. Second, it allows the flexibility for the data publisher to specify a different threshold $h$ for each combination of $QID$ and $s$ according to the perceived sensitivity of inferring $s$ from a group on $QID$. The recursive $(c, \ell)$-diversity cannot be used to bound the frequency of sensitive values that are not the most frequent. Confidence bounding provides greater flexibility than $\ell$-diversity in this aspect. However, recursive $(c, \ell)$-diversity can still prevent attribute linkages, even in the presence of background knowledge discussed earlier. Confidence bounding does not share the same merit.

*$(X, Y)$-Privacy*. $(X, Y)$-anonymity in Section 2.1 states that each group on $X$ has at least $k$ distinct values on $Y$ (e.g., diseases). However, if some $Y$ values occur more frequently than others, the probability of inferring a particular $Y$ value can be higher than $1/k$. To address this issue, Wang and Fung [2006] proposed a general privacy model, called $(X, Y)$-Privacy, which combines both $(X, Y)$-anonymity and confidence bounding. The general idea is to require each group $x$ on $X$ to contain at least $k$ records and $conf(x \rightarrow y) \leq h$ for any $y \in Y$, where $Y$ is a set of selected sensitive values and $h$ is a maximum confidence threshold.

*$(\alpha, k)$-Anonymity*. Wong et al. [2006] proposed a similar integrated privacy model, called $(\alpha, k)$-*anonymity*, requiring every *qid* in a Table $T$ to be shared by at least $k$ records and $conf(qid \rightarrow s) \leq \alpha$ for any sensitive value $s$, where $k$ and $\alpha$ are data publisher-specified thresholds. Nonetheless, both $(X, Y)$-Privacy and $(\alpha, k)$-anonymity may result in high distortion if the sensitive values are skewed.

*$(k, e)$-Anonymity*. Most work on $k$-anonymity and its extensions assumes categorical sensitive attributes. Zhang et al. [2007] proposed the notion of $(k, e)$-anonymity to address numerical sensitive attributes such as salary. The general idea is to partition the records into groups so that each group contains at least $k$ different sensitive values with a range of at least $e$. However, $(k, e)$-anonymity ignores the distribution of sensitive values within the range $\lambda$. If some sensitive values occur frequently within a subrange of $\lambda$, then the attacker could still confidently infer the subrange in a group. This type of attribute linkage attack is called the *proximity attack* [Li et al. 2008]. Consider a *qid* group of 10 data records with 7 different sensitive values, where 9 records have sensitive values in [30-35], and 1 record has value 80. The group is $(7, 50)$-anonymous because $80 - 30 = 50$. Still, the attacker can infer that a victim inside the group has a sensitive value falling into [30-35] with 90% confidence. Li et al. [2008] proposed an alternative privacy model, called $(\epsilon, m)$-*anonymity*. Given any numerical sensitive value $s$ in $T$, this privacy model bounds the probability of inferring $[s - \epsilon, s + \epsilon]$ to be at most $1/m$.

*t-Closeness*. In a spirit similar to the uninformative principle discussed earlier, Li et al. [2007] observed that when the overall distribution of a sensitive attribute is skewed, $\ell$-diversity does not prevent attribute linkage attacks. Consider a patient table where 95% of records have *Flu* and 5% of records have *H*IV. Suppose that a *qid* group has 50% of *F*lu and 50% of *HIV* and, therefore, satisfies 2-diversity. However, this group presents a serious privacy threat because any record owner in the group could be inferred as having *HIV* with 50% confidence, compared to 5% in the overall table.

To prevent *skewness attack*, Li et al. [2007] proposed a privacy model, called *t*-Closeness, which requires the distribution of a sensitive attribute in any group on *QID* to be close to the distribution of the attribute in the overall table. *t*-closeness uses the *Earth Mover Distance* (*EMD*) function to measure the closeness between two distributions of sensitive values, and requires the closeness to be within $t$. *t*-closeness has several limitations and weaknesses. First, it lacks the flexibility of specifying different protection levels for different sensitive values. Second, the *EMD* function is not suitable for preventing attribute linkage on numerical sensitive attributes [Li et al. 2008]. Third, enforcing *t*-closeness would greatly degrade the data utility because it requires the distribution of sensitive values to be the same in all *qid* groups. This would significantly damage the correlation between *QID* and sensitive attributes. One way to decrease the damage is to relax the requirement by adjusting the thresholds with the increased risk of skewness attack [Domingo-Ferrer and Torra 2008], or to employ the probabilistic privacy models in Section 2.4.

*Personalized Privacy*. Xiao and Tao [2006b] proposed the notion of *personalized privacy* to allow each record owner to specify her own privacy level. This model assumes that each sensitive attribute has a taxonomy tree and that each record owner specifies a guarding node in this tree. The record owner's privacy is violated if an attacker is able to infer any domain sensitive value within the subtree of her guarding node with a probability, called *breach probability*, greater than a certain threshold. For example, suppose *HIV* and *SARS* are child nodes of *Infectious Disease* in the taxonomy tree. A *HIV* patient Alice can set the guarding node to *Infectious Disease*, meaning that she allows people to infer that she has some infectious diseases, but not any specific type of infectious disease. Another *HIV* patient, Bob, does not mind disclosing his medical information, so he does not set any guarding node for this sensitive attribute.

Although both confidence bounding and personalized privacy take an approach to bound the confidence or probability of inferring a sensitive value from a *qid* group, they have differences. In the confidence bounding approach, the data publisher imposes a universal privacy requirement on the entire data set, so the minimum level of privacy protection is the same for every record owner. In the personalized privacy approach, a guarding node is specified for each record by its owner. The advantage is that each record owner may specify a guarding node according to her own tolerance on sensitivity. Experiments show that this personalized privacy requirement could result in lower information loss than the universal privacy requirement [Xiao and Tao 2006b]. In practice, however, it is unclear how individual record owners would set their guarding node. Often, a reasonable guarding node depends on the distribution of sensitive values in the whole table or in a group. For example, knowing that her disease is very common, a record owner may set a more special (lower privacy protected) guarding node for her record. Nonetheless, the record owners usually have no access to the distribution of sensitive values in their *qid* group or in the whole table before the data is published. Without such information, the tendency is to play safe by setting a more general (higher privacy protected) guarding node, which may negatively affect the utility of data.

*FF-Anonymity*. All previous work assumes that the data table can be divided into quasi-identifying (*QID*) attributes and sensitive attributes. Yet, this assumption does

not hold when an attribute contains both sensitive values and quasi-identifying values. Wang et al. [2009] identify a class of *freeform attacks* of the form $X \rightarrow s$, where $s$ and the values in $X$ can be any value of any attribute in the table $T$. $X \rightarrow s$ is a privacy breach if any record in $T$ matching $X$ can infer a sensitive value $s$ with a high probability. Their proposed privacy model, *FF-anonymity*, bounds the probability of all potential privacy breaches in the form $X \rightarrow s$ to be below a given threshold.

### 2.3. Table Linkage

Both record linkage and attribute linkage assume that the attacker already knows the victim's record is in the released table $T$. However, in some cases, the presence (or the absence) of the victim's record in $T$ already reveals the victims's sensitive information. Suppose a hospital releases a data table with a particular type of disease. Identifying the presence of the victim's record in the table is already damaging. A *table linkage* occurs if an attacker can confidently infer the presence or the absence of the victim's record in the released table. The following example illustrates the privacy threat of a table linkage.

*Example* 2.8. Suppose the data publisher has released a 3-anonymous patient table $T$ (Table II(c)). To launch a table linkage on a target victim, for instance, Alice, on $T$, the attacker is presumed to also have access to an external public table $E$ (Table II(d)) where $T \subseteq E$. The probability that Alice is present in $T$ is $\frac{4}{5} = 0.8$ because there are 4 records in $T$ (Table II(c)) and 5 records in $E$ (Table II(d)) containing $\langle Artist, Female, [30-35)\rangle$. Similarly, the probability that Bob is present in $T$ is $\frac{3}{4} = 0.75$.

*$\delta$-Presence*. To prevent table linkage, Nergiz et al. [2007] proposed to bound the probability of inferring the presence of any potential victim's record within a specified range $\delta = (\delta_{min}, \delta_{max})$. Formally, given an external public table $E$ and a private table $T$, where $T \subseteq E$, a generalized table $T'$ satisfies $(\delta_{min}, \delta_{max})$*-presence* if $\delta_{min} \leq P(t \in T|T') \leq \delta_{max}$ for all $t \in E$. $\delta$-presence can indirectly prevent record and attribute linkages because if the attacker has at most $\delta\%$ of confidence that the target victim's record is present in the released table, then the probability of a successful linkage to her record and sensitive attribute is at most $\delta\%$. Though $\delta$-presence is a relatively "safe" privacy model, it assumes that the data publisher has access to the same external table $E$ as the attacker does. This may not be a practical assumption.

### 2.4. Probabilistic Attack

There is another family of privacy models that does not focus on exactly what records, attributes, and tables the attacker can link to a target victim, but focuses on how the attacker would change his/her probabilistic belief on the sensitive information of a victim after accessing the published data. In general, this group of privacy models aims at achieving the uninformative principle [Machanavajjhala et al. 2006], whose goal is to ensure that the difference between the prior and posterior beliefs is small.

*$(c, t)$-Isolation*. Chawla et al. [2005] suggested that having access to the published anonymous data table should not enhance an attacker's power of isolating any record owner. Consequently, they proposed a privacy model to prevent *$(c, t)$-isolation* in a statistical database. Suppose $p$ is a data point of a target victim $v$ in a data table, and $q$ is the attacker's inferred data point of $v$ by using the published data and the background information. Let $\delta_p$ be the distance between $p$ and $q$. We say that point $q$ $(c, t)$*-isolates* point $p$ if $B(q, c\delta_p)$ contains fewer than $t$ points in the table, where $B(q, c\delta_p)$ is a ball of

radius $c\delta_p$ centered at point $q$. Preventing $(c, t)$-isolation can be viewed as preventing record linkages. Their model considers distances among data records and, therefore, is more suitable for numerical attributes in statistical databases.

*$\epsilon$-Differential privacy*. Dwork [2006] proposed an insightful privacy notion: the risk to the record owner's privacy should not substantially increase as a result of participating in a statistical database. Instead of comparing the prior probability and the posterior probability before and after accessing the published data, Dwork proposed to compare the risk with and without the record owner's data in the published data. Consequently, Dwork [2006] proposed a privacy model called *$\epsilon$-differential privacy* to ensure that the removal or addition of a single database record does not significantly affect the outcome of any analysis. It follows that no risk is incurred by joining different databases. Based on the same intuition, if a record owner does not provide his/her actual information to the data publisher, it will not make much difference in the result of the anonymization algorithm.

The following is a more formal definition of $\epsilon$-differential privacy [Dwork 2006]: A randomized function $F$ ensures $\epsilon$-differential privacy if for all data sets $T_1$ and $T_2$ differing on at most one record, $|ln\frac{P(F(T_1)=S)}{P(F(T_2)=S)}| \leq \epsilon$ for all $S \in Range(F)$, where $Range(F)$ is the set of possible outputs of the randomized function $F$. Although $\epsilon$-differential privacy does not prevent record and attribute linkages studied in earlier sections, it assures record owners that they may submit their personal information to the database securely in the knowledge that nothing, or almost nothing, can be discovered from the database with their information that could not have been discovered without their information. Dwork [2006] formally proved that $\epsilon$-differential privacy can provide a guarantee against attackers with arbitrary background knowledge. This strong guarantee is achieved by comparison with and without the record owner's data in the published data. Dwork [2007] proved that if the number of queries is sublinear in $n$, the noise to achieve differential privacy is bounded by $o(\sqrt{n})$, where $n$ is the number of records in the database. Dwork [2008] further showed that the notion of differential privacy is applicable to both interactive and noninteractive query models, discussed in Sections 1.1 and 8.1; refer to Dwork [2008] for a survey on differential privacy.

*$(d, \gamma)$-Privacy*. Rastogi et al. [2007] presented a probabilistic privacy definition *$(d, \gamma)$-privacy*. Let $P(r)$ and $P(r|T)$ be the prior probability and the posterior probability of the presence of a victim's record in the data table $T$ before and after examining the published table $T$. $(d, \gamma)$-privacy bounds the difference of the prior and posterior probabilities and provides a provable guarantee on privacy and information utility, while most previous work lacks such a formal guarantee. Rastogi et al. [2007] showed that a reasonable trade-off between privacy and utility can be achieved only when the prior belief is small. Nonetheless, $(d, \gamma)$-privacy is designed to protect only against attacks that are *d-independent*: an attack is $d$-independent if the prior belief $P(r)$ satisfies the conditions $P(r) = 1$ or $P(r) \leq d$ for all records $r$, where $P(r) = 1$ means that the attacker already knows that $r$ is in $T$ and no protection on $r$ is needed. Machanavajjhala et al. [2008] pointed out that this $d$-independence assumption may not hold in some real-life applications. Differential privacy in comparison does not have to assume that records are independent or that an attacker has a prior belief bounded by a probability distribution.

*Distributional privacy*. Motivated by the learning theory, Blum et al. [2008] presented a privacy model called *distributional privacy* for a noninteractive query model. The key idea is that when a data table is drawn from a distribution, the table should reveal only information about the underlying distribution, and nothing else. Distributional privacy is a strictly stronger privacy notion than differential privacy, and

can answer all queries over a discretized domain in a concept class of polynomial VC-dimension.[1] Yet, the algorithm has high computational cost. Blum et al. [2008] presented an efficient algorithm specifically for simple interval queries with limited constraints. The problems of developing efficient algorithms for more complicated queries remain open.

## 3. ANONYMIZATION OPERATIONS

Typically, the original table does not satisfy a specified privacy requirement and the table must be modified before being published. The modification is done by applying a sequence of anonymization operations to the table. An anonymization operation comes in several flavors: generalization, suppression, anatomization, permutation, and perturbation. Generalization and suppression replace values of specific description, typically the *QID* attributes, with less specific description. Anatomization and permutation deassociate the correlation between *QID* and sensitive attributes by grouping and shuffling sensitive values in a *qid* group. Perturbation distorts the data by adding noise, aggregating values, swapping values, or generating synthetic data based on some statistical properties of the original data. Below, we discuss these anonymization operations in detail.

### 3.1. Generalization and Suppression

Each generalization or suppression operation hides some details in *QID*. For a categorical attribute, a specific value can be replaced with a general value according to a given taxonomy. In Figure 3, the parent node *Professional* is more general than the child nodes *Engineer* and *Lawyer*. The root node, *ANY_Job*, represents the most general value in *Job*. For a numerical attribute, exact values can be replaced with an interval that covers exact values. If a taxonomy of intervals is given, the situation is similar to categorical attributes. More often, however, no predetermined taxonomy is given for a numerical attribute. Different classes of anonymization operations have different implications on privacy protection, data utility, and search space. But they all result in a less precise but consistent representation of the original data.

A *generalization* replaces some values with a parent value in the taxonomy of an attribute. The reverse operation of generalization is called *specialization*. A *suppression* replaces some values with a special value, indicating that the replaced values are not disclosed. The reverse operation of suppression is called *disclosure*. Below, we summarize five generalization schemes.

*Full-domain generalization scheme* [LeFevre et al. 2005; Samarati 2001; Sweeney 2002b]. In this scheme, all values in an attribute are generalized to the same level of the taxonomy tree. For example, in Figure 3, if *Lawyer* and *Engineer* are generalized to *Professional*, then it also requires generalizing *Dancer* and *Writer* to *Artist*. The search space for this scheme is much smaller than the search space for other schemes below, but the data distortion is the largest because of the same granularity level requirement on all paths of a taxonomy tree.

*Subtree generalization scheme* [Bayardo and Agrawal 2005; Fung et al. 2005, 2007; Iyengar 2002; LeFevre et al. 2005]. In this scheme, at a nonleaf node, either all child values or none are generalized. For example, in Figure 3, if *Engineer* is generalized to *Professional*, this scheme also requires the other child node, *Lawyer*, to be generalized to *Professional*, but *Dancer* and *Writer*, which are child nodes of *Artist*, can remain ungeneralized. Intuitively, a generalized attribute has values that form a "cut" through

[1]Vapnik-Chervonenkis dimension is a measure of the capacity of a statistical classification algorithm.

its taxonomy tree. A *cut* of a tree is a subset of values in the tree that contains exactly one value on each root-to-leaf path.

*Sibling generalization scheme* [LeFevre et al. 2005]. This scheme is similar to the subtree generalization, except that some siblings may remain ungeneralized. A parent value is then interpreted as representing all missing child values. For example, in Figure 3, if *Engineer* is generalized to *Professional*, and *Lawyer* remains ungeneralized, *Professional* is interpreted as all jobs covered by *Professional* except for *Lawyer*. This scheme produces less distortion than subtree generalization schemes because it only needs to generalize the child nodes that violate the specified threshold.

*Cell generalization scheme* [LeFevre et al. 2005; Wong et al. 2006; Xu et al. 2006]. In all of the above schemes, if a value is generalized, all its instances are generalized. Such schemes are called *global recoding*. In cell generalization, also known as *local recoding*, some instances of a value may remain ungeneralized while other instances are generalized. For example, in Table II(a) the *Engineer* in the first record is generalized to *Professional*, while the *Engineer* in the second record can remain ungeneralized. Compared with global recoding schemes, this scheme is more flexible; and therefore it produces a smaller data distortion. Nonetheless, it is important to note that the utility of data is adversely affected by this flexibility, which causes a data exploration problem: most standard data mining methods treat *Engineer* and *Professional* as two independent values, but, in fact, they are not. For example, building a decision tree from such a generalized table may result in two branches, *Professional* $\rightarrow$ *class*2 and *Engineer* $\rightarrow$ *class*1. It is unclear which branch should be used to classify a new engineer. Though very important, this aspect of data utility has been ignored by all work that employed the local recoding scheme. Data produced by global recoding does not suffer from this data exploration problem.

*Multidimensional generalization* [LeFevre et al. 2006a, 2006b]. Let $D_i$ be the domain of an attribute $A_i$. A single-dimensional generalization, such as full-domain generalization and subtree generalization, is defined by a function $f_i : D_{A_i} \rightarrow D'$ for each attribute $A_i$ in *QID*. In contrast, a multidimensional generalization is defined by a single function $f : D_{A_1} \times \cdots \times D_{A_n} \rightarrow D'$, which is used to generalize $qid = \langle v_1, \ldots, v_n \rangle$ to $qid' = \langle u_1, \ldots, u_n \rangle$ where for every $v_i$, either $v_i = u_i$ or $v_i$ is a child node of $u_i$ in the taxonomy of $A_i$. This scheme flexibly allows two *qid* groups, even having the same value $v$, to be independently generalized into different parent groups. For example $\langle Engineer, Male \rangle$ can be generalized to $\langle Engineer, ANY\_Sex \rangle$ while $\langle Engineer, Female \rangle$ can be generalized to $\langle Professional, Female \rangle$. The generalized table contains both *Engineer* and *Professional*. This scheme produces less distortion than the full-domain and subtree generalization schemes because it needs to generalize only the *qid* groups that violate the specified threshold. Note that in this multidimensional scheme, *all* records in a *qid* are generalized to the same $qid'$, but cell generalization does not have such constraint. Although both schemes suffer from the data exploration problem, Nergiz and Clifton [2007] further evaluated a family of clustering-based algorithms that even attempted to improve data utility by ignoring the restrictions of the given taxonomies.

There are also different suppression schemes. *Record suppression* [Bayardo and Agrawal 2005; Iyengar 2002; LeFevre et al. 2005; Samarati 2001] refers to suppressing an entire record. *Value suppression* [Wang et al. 2005, 2007] refers to suppressing every instance of a given value in a table. *Cell suppression* (or *local suppression*) [Cox 1980; Meyerson and Williams 2004] refers to suppressing *some* instances of a given value in a table.

In summary, the choice of anonymization operations has an implication on the search space of anonymous tables and data distortion. The full-domain generalization

has the smallest search space but the largest distortion, and the local recoding scheme has the largest search space but the least distortion. For a categorical attribute with a taxonomy tree $H$, the number of possible cuts in subtree generalization, denoted $C(H)$, is equal to $C(H_1) \times \cdots \times C(H_u) + 1$ where $H_1, \ldots, H_u$ are the subtrees rooted at the children of the root of $H$, and 1 is for the trivial cut at the root of $H$. The number of potential modified tables is equal to the product of such numbers for all the attributes in *QID*. The corresponding number is much larger if a local recoding scheme is adopted because any subset of values can be generalized while the rest remains ungeneralized for each attribute in *QID*.

A table is *minimally anonymous* if it satisfies the given privacy requirement and its sequence of anonymization operations cannot be reduced without violating the requirement. A table is *optimally anonymous* if it satisfies the given privacy requirement and contains most information according to the chosen information metric among all satisfying tables. See Section 4 for different types of information metrics. Various works have shown that finding the optimal anonymization is NP-hard: Samarati [2001] showed that the optimal $k$-anonymity by full-domain generalization is very costly; Meyerson and Williams [2004] and Aggarwal et al. [2005] proved that the optimal $k$-anonymity by cell suppression, value suppression, and cell generalization is NP-hard; Wong et al. [2006] proved that the optimal $(\alpha, k)$-anonymity by cell generalization is NP-hard. In most cases, finding a minimally anonymous table is a reasonable solution, and can be done efficiently.

### 3.2. Anatomization and Permutation

*Anatomization* [Xiao and Tao 2006a]. Unlike generalization and suppression, anatomization does not modify the quasi-identifier or the sensitive attribute, but deassociates the relationship between the two. Precisely, the method releases the data on *QID* and the data on the sensitive attribute in two separate tables: a *quasi-identifier table* (*QIT*) contains the *QID* attributes, a *sensitive table* (*ST*) contains the sensitive attributes, and both QIT and ST have one common attribute, *GroupID*. All records in the same group will have the same value on *GroupID* in both tables, and therefore are linked to the sensitive values in the group in the exact same way. If a group has $\ell$ distinct sensitive values and each distinct value occurs exactly once in the group, then the probability of linking a record to a sensitive value by *GroupID* is $1/\ell$. The attribute linkage attack can be distorted by increasing $\ell$.

*Example* 3.1. Suppose that the data publisher wants to release the patient data in Table III(a), where *Disease* is a sensitive attribute and $QID = \{Age, Sex\}$. First, partition (or generalize) the original records into *qid* groups so that, in each group, at most $1/\ell$ of the records contain the same *Disease* value. This intermediate Table II(b) contains two *qid* groups: $\langle [30{-}35), Male \rangle$ and $\langle [35{-}40), Female \rangle$. Next, create QIT (Table III(c)) to contain all records from the original Table III(a), but replace the sensitive values by the *GroupID*s, and create ST (Table III(d)) to contain the count of each *Disease* for each *qid* group. QIT and ST satisfy the privacy requirement with $\ell \leq 2$ because each *qid* group in QIT infers any associated *Disease* in ST with probability at most $1/\ell = 1/2 = 50\%$.

The major advantage of *anatomy* is that the data in both QIT and ST is unmodified. Xiao and Tao [2006a] showed that the anatomized tables can more accurately answer aggregate queries involving domain values of the *QID* and sensitive attributes than the generalization approach. The intuition is that, in a generalized table, domain values are lost, and without additional knowledge, the uniform distribution assumption is the best that can be used to answer a query about domain values. In contrast, all

**Table III.** Anatomy

(a) Original patient data

| Age | Sex | Disease (sensitive) |
|---|---|---|
| 30 | Male | Hepatitis |
| 30 | Male | Hepatitis |
| 30 | Male | HIV |
| 32 | Male | Hepatitis |
| 32 | Male | HIV |
| 32 | Male | HIV |
| 36 | Female | Flu |
| 38 | Female | Flu |
| 38 | Female | Heart |
| 38 | Female | Heart |

(b) Intermediate *QID*-grouped table

| Age | Sex | Disease (sensitive) |
|---|---|---|
| [30–35) | Male | Hepatitis |
| [30–35) | Male | Hepatitis |
| [30–35) | Male | HIV |
| [30–35) | Male | Hepatitis |
| [30–35) | Male | HIV |
| [30–35) | Male | HIV |
| [35–40) | Female | Flu |
| [35–40) | Female | Flu |
| [35–40) | Female | Heart |
| [35–40) | Female | Heart |

(c) Quasi-identifier table (QIT) for release

| Age | Sex | GroupID |
|---|---|---|
| 30 | Male | 1 |
| 30 | Male | 1 |
| 30 | Male | 1 |
| 32 | Male | 1 |
| 32 | Male | 1 |
| 32 | Male | 1 |
| 36 | Female | 2 |
| 38 | Female | 2 |
| 38 | Female | 2 |
| 38 | Female | 2 |

(d) Sensitive table (ST) for release

| GroupID | Disease (sensitive) | Count |
|---|---|---|
| 1 | Hepatitis | 3 |
| 1 | HIV | 3 |
| 2 | Flu | 2 |
| 2 | Heart | 2 |

domain values are retained in the anatomized tables, which give the exact distribution of domain values. For instance, suppose that the data recipient wants to count the number of patients of age 38 having heart disease. The correct count from the original Table III(a) is 2. The expected count from the anatomized Table III(c) and Table III(d) is $3 \times \frac{2}{4} = 1.5$, since 2 out of the 4 records in $GroupID = 2$ in Table III(d) have heart disease. This count is more accurate than the expected count $2 \times \frac{1}{5} = 0.4$, from the generalized Table III(b), where the $\frac{1}{5}$ comes from the fact that the 2 patients with heart disease have an equal chance to be of age $\{35, 36, 37, 38, 39\}$.

Yet, with the data published in two tables, it is unclear how standard data mining tools such as classification, clustering, and association mining tools can be applied to the published data, and new tools and algorithms need to be designed. Also, anatomy is not suitable for continuous data publishing, which will be discussed further in Section 6.3. The generalization approach does not suffer from the same problem because all attributes are released in the same table.

*Permutation*. Sharing the same spirit of anatomization, Zhang et al. [2007] proposed an approach called *permutation*. The idea is to deassociate the relationship between a quasi-identifier and a *numerical* sensitive attribute by partitioning a set of data records into groups and shuffling their sensitive values within each group.

### 3.3. Perturbation

Perturbation has a long history in statistical disclosure control [Adam and Wortman 1989] due to its simplicity, efficiency, and ability to preserve statistical information. The general idea is to replace the original data values with some synthetic data values, so that the statistical information computed from the perturbed data does not differ significantly from the statistical information computed from the original data. The perturbed data records do not correspond to real-world record owners, so the attacker

cannot perform the sensitive linkages or recover sensitive information from the published data.

Compared to the other anonymization operations discussed earlier, one limitation of the perturbation approach is that the published records are "synthetic" in that they do not correspond to the real-world entities represented by the original data; therefore, individual records in the perturbed data are basically meaningless to the human recipients. Only the statistical properties explicitly selected by the data publisher are preserved. In such a case, the data publisher may consider releasing the statistical information or the data mining results rather than the perturbed data [Domingo-Ferrer 2008]. In contrast, generalization and suppression make the data less precise than, but semantically consistent with, the raw data, and hence preserve the truthfulness of the data. For example, after analyzing the statistical properties of a collection of perturbed patient records, a drug company wants to focus on a small number of patients for further analysis. This stage requires the truthful record information instead of perturbed record information. Below, we discuss several commonly used perturbation methods, including additive noise, data swapping, and synthetic data generation.

*Additive noise*. Additive noise is a widely used privacy protection method in statistical disclosure control [Adam and Wortman 1989; Brand 2002]. It is often used for hiding sensitive numerical data (e.g., salary). The general idea is to replace the original sensitive value $s$ with $s+, r$ where $r$ is a random value drawn from some distribution. Privacy was measured by how closely the original values of a modified attribute can be estimated [Agrawal and Aggarwal 2001]. Fuller [1993] and Kim and Winkler [1995] showed that some simple statistical information, like means and correlations, can be preserved by adding random noise. Experiments in Agrawal and Srikant [2000], Du and Zhan [2003], and Evfimievski et al. [2002] further suggested that some data mining information can be preserved in the randomized data. However, Kargupta et al. [2003] pointed out that some reasonably close sensitive values can be recovered from the randomized data when the correlation among attributes is high but the noise is not. Huang et al. [2005] presented an improved randomization method to limit this type of privacy breach. Some representative statistical disclosure control methods that employ additive noise are discussed in Sections 2.4 and 5.5.

*Data swapping*. The general idea of data swapping is to anonymize a data table by exchanging values of sensitive attributes among individual records, while the swaps maintain the low-order frequency counts or marginals for statistical analysis. It can be used to protect numerical attributes [Reiss et al. 1982] and categorical attributes [Reiss 1984]. An alternative swapping method is *rank swapping*: First rank the values of an attribute $A$ in ascending order. Then for each value $v \in A$, swap $v$ with another value $u \in A$, where $u$ is randomly chosen within a restricted range $p\%$ of $v$. Rank swapping can better preserve statistical information than the ordinary data swapping [Domingo-Ferrer and Torra 2002].

*Synthetic data generation.* Many statistical disclosure control methods use synthetic data generation to preserve record owners' privacy and retain useful statistical information [Rubin]. The general idea is to build a statistical model from the data and then to sample points from the model. These sampled points form the synthetic data for data publication instead of the original data. An alternative synthetic data generation approach is *condensation* [Aggarwal and Yu 2008a, 2008b]. The idea is to first condense the records into multiple groups. For each group, extract some statistical information, such as sum and covariance, that suffices to preserve the mean and correlations across the different attributes. Then, based on the statistical information, for publication generate points for each group following the statistical characteristics of the group.

## 4. INFORMATION METRICS

Privacy preservation is one side of anonymization. The other side is retaining information so that the published data remains practically useful. There are broad categories of information metrics for measuring data usefulness. A *data metric* measures the data quality in the entire anonymous table with respect to the data quality in the original table. A *search metric* guides each step of an anonymization (search) algorithm to identify an anonymous table with maximum information or minimum distortion. Often, this is achieved by ranking a set of possible anonymization operations and then greedily performing the "best" one at each step in the search. Since the anonymous table produced by a search metric is eventually evaluated by a data metric, the two types of metrics usually share the same principle for measuring data quality.

Alternatively, an information metric can be categorized by its information purposes, including *general purpose*, *special purpose*, or *trade-off purpose*. Below, we discuss some commonly used data and search metrics according to their purposes.

### 4.1. General Purpose Metrics

In many cases, the data publisher does not know how the published data will be analyzed by the recipient. This is very different from privacy-preserving data mining (PPDM), which assumes that the data mining task is known. In PPDP, for example, the data may be published on the Web and a recipient may analyze the data according to her own purpose. An information metric good for one recipient may not be good for another recipient. In such scenarios, a reasonable information metric is to measure "similarity" between the original data and the anonymous data, which underpins the *principle of minimal distortion* [Samarati 2001; Sweeney 1998, 2002b]. In the minimal distortion metric or *MD*, a penalty is charged to each instance of a value that is generalized or suppressed. For example, generalizing 10 instances of *Engineer* to *Professional* causes 10 units of distortion, and further generalizing these instances to *ANY_Job* causes another 10 units of distortion. This metric is a single attribute measure, and was previously used in Samarati [2001], Sweeney [2002a, 2002b], and Wang and Fung [2006] as a data metric and search metric.

*ILoss* is a data metric proposed in Xiao and Tao [2006b] to capture the information loss of generalizing a specific value to a general value $v_g$: $ILoss(v_g) = \frac{|v_g|-1}{|D_A|}$ where $|v_g|$ is the number of domain values that are descendants of $v_g$, and $|D_A|$ is the number of domain values in the attribute $A$ of $v_g$. This data metric requires all original data values to be at the leaves in the taxonomy. $ILoss(v_g) = 0$ if $v_g$ is an original data value in the table. In words, $ILoss(v_g)$ measures the fraction of domain values generalized by $v_g$. For example, generalizing one instance of *Dancer* to *Artist* in Figure 3 has $ILoss(Artist) = \frac{2-1}{4} = 0.25$. The loss of a generalized record $r$ is given by

$$ILoss(r) = \sum_{v_g \in r} (w_i \times ILoss(v_g)), \tag{2}$$

where $w_i$ is a positive constant specifying the penalty weight of attribute $A_i$ of $v_g$. The overall loss of a generalized table $T$ is given by

$$ILoss(T) = \sum_{r \in T} ILoss(r). \tag{3}$$

Both *MD* and *ILoss* charge a penalty for generalizing a value in a record independently of other records. For example, generalizing 99 instances of *Engineer* and 1 instance of *Lawyer* to *Professional* will have the same penalty as generalizing 50 instances of *Engineer* and 50 instances of *Lawyer*. In both cases, 100 instances are made indistinguishable. The difference is that, before the generalization, 99 instances were already indistinguishable in the first case, whereas only 50 instances are indistinguishable in the second case. Therefore, the second case makes more originally distinguishable records become indistinguishable. The *discernibility metric*, or *DM* [Skowron and Rauszer 1992], addresses this notion of loss by charging a penalty to each record for being indistinguishable from other records with respect to *QID*. If a record belongs to a group of size $s$, the penalty for the record will be $s$. This data metric, used in Bayardo and Agrawal [2005], LeFevre et al. [2006a], Machanavajjhala et al. [2006, 2007], Vinterbo [2004], and Xu et al. [2006], works exactly against the $k$-anonymization that seeks to make records indistinguishable with respect to *QID*.

A simple search metric, called *distinctive attribute*, or *DA*, was employed in Sweeney [1998] to guide the search for a minimally anonymous table in a full-domain generalization scheme. The heuristic selects the attribute having the most number of distinctive values in the data for generalization. Note that this type of simple heuristic only serves the purpose of guiding the search, but does not quantify the utility of an anonymous table.

### 4.2. Special Purpose Metrics

If the purpose of the data is known at the time of publication, the purpose can be taken into account during anonymization to better retain information. For example, if the data is published for modeling the classification of a target attribute in the table, then it is important not to generalize the values whose distinctions are essential for discriminating the class labels in the target attribute. An often-asked question is if the purpose of data is known, why not extract and publish a data mining result for that purpose (such as a classifier) instead of the data [Nergiz and Clifton 2007]? The answer is that publishing a data mining result is a commitment at the algorithmic level, which is neither practical for the nonexpert data publisher nor desirable for the data recipient. In practice, there are many ways to mine the data even for a given purpose, and typically it is unknown which one is the best until the data is received and different ways are tried. A real-life example is the release of the Netflix data (New York Times, Oct. 2, 2006) discussed in Section 1. Netflix wanted to provide the participants the greatest flexibility in performing their desired analysis, instead of limiting them to a specific type of analysis.

For concreteness, let us consider the classification problem where the goal is to classify *future cases* into some predetermined classes, drawn from the same underlying population as the *training cases* in the published data. The training cases contain both the useful *classification information* that can improve the classification model, and the useless *noise* that can degrade the classification model. Specifically, the useful classification information is the information that can differentiate the target classes, and holds not only for training cases, but also for future cases. In contrast, the useless noise holds only for training cases. Clearly, only the useful classification information that helps classification should be retained. For example, a patient's birth year is likely to be part of information for classifying lung cancer if the disease occurs more frequently among elderly people, but the exact birth date is likely to be noise. In this case, generalizing birth date to birth year helps classification because it eliminates the noise. This example shows that simply minimizing the distortion to the data, as adopted by

all general purpose metrics and optimal $k$-anonymization, is not addressing the right problem.

To address the classification goal, the distortion should be measured by the classification error on future cases. Since future data is not available in most scenarios, most developed methods [Fung et al. 2005, 2007; Iyengar 2002] measure accuracy on the training data. Research results in Fung et al. [2005, 2007] suggest that the useful classification knowledge is captured by different combinations of attributes. Generalization and suppression may destroy some of these useful "classification structures," but other useful structures may emerge to help. In some cases, generalization and suppression may even improve the classification accuracy because some noise has been removed.

Iyengar [2002] presented the first work on PPDP for classification. He proposed the *classification metric*, or *CM*, to measure the classification error on the training data. The idea is to charge a penalty for each record suppressed or generalized to a group in which the record's class is not the majority class. The intuition is that a record having a non-majority class in a group will be classified as the majority class, which is an error because it disagrees with the record's original class.

*CM* is a data metric, and hence penalizes modification to the training data. This does not quite address the classification goal, which is actually better off by generalizing useless noise into useful classification information. For classification, a more relevant approach is searching for a "good" anonymization according to some heuristics. In other words, instead of optimizing a data metric, this approach employs a search metric to rank anonymization operations at each step in the search. An anonymization operation is ranked high if it retains useful classification information. The search metric could be adapted by different anonymization algorithms. For example, a greedy algorithm or a hill-climbing optimization algorithm can be used to identify a minimal sequence of anonymization operations for a given search metric. We discuss anonymization algorithms in Section 5.

Neither a data metric nor a search metric guarantees a good classification for future cases. It is essential to experimentally evaluate the impact of anonymization by building a classifier from the anonymous data and seeing how it performs on testing cases. Few works [Fung et al. 2005, 2007; Iyengar 2002; LeFevre et al. 2006b; Wang et al. 2004] have actually conducted such experiments, although many, such as Bayardo and Agrawal [2005], adopted *CM* in an attempt to address the classification problem.

### 4.3. Trade-off Metrics

The special purpose information metrics aim at preserving data usefulness for a given data mining task. The catch is that the anonymization operation that gains maximum information may also lose so much privacy that no other anonymization operation can be performed. The idea of trade-off metrics is to consider both the privacy and information requirements at every anonymization operation and to determine an optimal trade-off between the two requirements.

Fung et al. [2005, 2007] proposed a search metric based on the principle of *information/privacy trade-off*. Suppose that the anonymous table is searched by iteratively specializing a general value into child values. Each specialization operation splits each group containing the general value into a number of groups, one for each child value. Each specialization operation $s$ gains some information, denoted $IG(s)$, and loses some privacy, $PL(s)$. This search metric prefers the specialization $s$ that maximizes the

information gained per each loss of privacy:

$$IGPL(s) = \frac{IG(s)}{PL(s) + 1}. \tag{4}$$

The choice of $IG(s)$ and $PL(s)$ depends on the information metric and privacy model. For example, in classification analysis, $IG(s)$ could be the information gain [Quinlan 1993] defined as the decrease of the class entropy [Shannon 1948] after specializing a general group into several specialized groups. Alternatively, $IG(s)$ could be the decrease of distortion measured by $MD$, described in Section 4.1, after performing $s$. For $k$-anonymity, Fung et al. [2005, 2007] measured the privacy loss $PL(s)$ by the average decrease of anonymity over all $QID_j$ that contain the attribute of $s$, that is,

$$PL(s) = avg\{A(QID_j) - A_s(QID_j)\},$$

where $A(QID_j)$ and $A_s(QID_j)$ denote the anonymity of $QID_j$ before and after the specialization. One variant is to maximize the gain of information by setting $PL(s)$ to zero. The catch is that the specialization that gains maximum information may also lose so much privacy that no other specializations can be performed. Note that the principle of information/privacy trade-off can also be used to select a generalization $g$, in which case it will minimize

$$ILPG(g) = \frac{IL(g)}{PG(g) + 1}, \tag{5}$$

where $IL(g)$ denotes the information loss and $PG(g)$ denotes the privacy gain by performing $g$.

## 5. ANONYMIZATION ALGORITHMS

In this section, we examine some representative anonymization algorithms. Refer to Table IV for a characterization based on the privacy model (Section 2), anonymization operation (Section 3), and information metric (Section 4). Our presentation of algorithms is organized according to linkage models; we then discuss the potential privacy threats, even though a data table has been optimally anonymized.

### 5.1. Algorithms for the Record Linkage Model

We broadly classify record linkage anonymization algorithms into three families: the first two, *optimal anonymization* and *minimal anonymization*, use generalization and suppression methods; the third family uses *perturbation methods*.

*5.1.1. Optimal Anonymization Algorithms.* The first family finds an optimal $k$-anonymization, for a given data metric by limiting to full-domain generalization and record suppression. Since the search space for the full-domain generalization scheme is much smaller than other schemes, finding an optimal solution is feasible for small data sets. This type of exhaustive search, however, is not scalable to large data sets, especially if a more flexible anonymization scheme is employed.

*MinGen.* Sweeney's [2002b] *MinGen* algorithm exhaustively examines all potential full-domain generalizations to identify the optimal generalization measured in $MD$. Sweeney acknowledged that this exhaustive search is impractical even for the modest-sized data sets, motivating the second family of $k$-anonymization algorithms for later discussion. Samarati [2001] proposed a *binary search* algorithm that first identifies all minimal generalizations, and then finds the optimal generalization measured in

**Table IV.** Characterization of Anonymization Algorithms

| Algorithm | Operation | Metric | Optimality |
|---|---|---|---|
| **Record Linkage** | | | |
| Binary Search [Samarati 2001] | FG,RS | *MD* | optimal |
| MinGen [Sweeney 2002b] | FG,RS | *MD* | optimal |
| Incognito [LeFevre et al. 2005] | FG,RS | *MD* | optimal |
| K-Optimize [Bayardo and Agrawal 2005] | SG,RS | *DM*,*CM* | optimal |
| $\mu$-argus [Hundepool and Willenborg 1996] | SG,CS | *MD* | minimal |
| Datafly [Sweeney 1998] | FG,RS | *DA* | minimal |
| Genetic Algorithm [Iyengar 2002] | SG,RS | *CM* | minimal |
| Bottom-Up Generalization [Wang et al. 2004] | SG | *ILPG* | minimal |
| Top-Down Specialization (TDS) [Fung et al. 2005, 2007] | SG,VS | *IGPL* | minimal |
| TDS for Cluster Analysis [Fung et al. 2009] | SG,VS | *IGPL* | minimal |
| Mondrian Multidimensional [LeFevre et al. 2006a] | MG | *DM* | minimal |
| Bottom-Up & Top-Down Greedy [Xu et al. 2006] | CG | *DM* | minimal |
| TDS2P [Wang et al. 2005; Mohammed et al. 2009] | SG | *IGPL* | minimal |
| Condensation [Aggarwal and Yu 2008a, 2008b] | CD | heuristics | minimal |
| $r$-Gather Clustering [Aggarwal et al. 2006] | CL | heuristics | minimal |
| **Attribute Linkage** | | | |
| Top-Down Disclosure [Wang et al. 2005, 2007] | VS | *IGPL* | minimal |
| Progressive Local Recoding [Wong et al. 2006] | CG | *MD* | minimal |
| $\ell$-Diversity Incognito [Machanavajjhala et al. 2007] | FG,RS | *MD*,*DM* | optimal |
| InfoGain Mondrian [LeFevre et al. 2006b] | MG | *IG* | minimal |
| Anatomy [Xiao and Tao 2006a] | AM | heuristics | minimal |
| $(k, e)$-Anonymity Permutation [Zhang et al. 2007] | PM | min. error | optimal |
| Greedy Personalized [Xiao and Tao 2006b] | SG,CG | *ILoss* | minimal |
| $t$-Closeness Incognito [Li et al. 2007] | FG,RS | *DM* | optimal |
| **Table Linkage** | | | |
| SPALM [Nergiz et al. 2007] | FG | *DM* | optimal |
| MPALM [Nergiz et al. 2007] | MG | heuristics | minimal |
| **Probabilistic Attack** | | | |
| Cross-Training Round Sanitization [Chawla et al. 2005] | AN | statistical | N/A |
| $\epsilon$-Differential Privacy Additive Noise [Dwork 2006] | AN | statistical | N/A |
| $\alpha\beta$ Algorithm [Rastogi et al. 2007] | AN,SP | statistical | N/A |

FG = Full-domain Generalization, SG = Subtree Generalization, CG = Cell Generalization, MG = Multidimensional Generalization, RS = Record Suppression, VS =Value Suppression, CS = Cell Suppression, AM = Anatomization, PM = Permutation, AN = Additive Noise, SP = Sampling, CD = Condensation, CL=Clustering

*MD*. Enumerating all minimal generalizations is an expensive operation, and hence not scalable for large data sets.

*Incognito.* LeFevre et al. [2005] presented a suite of optimal bottom-up generalization algorithms, called *Incognito*, to generate all possible $k$-anonymous full-domain generalizations. These algorithms exploit the rollup property for computing the size of $qid$ groups.

*Observation* 5.1 (*Rollup Property*). If $qid$ is a generalization of $\{qid_1, \ldots, qid_c\}$, then $|qid| = \sum_{i=1}^{c} |qid_i|$.

The rollup property states that the parent group size $|qid|$ can be directly computed from the sum of all child group sizes $|qid_i|$, implying that the group size $|qid|$ of all possible generalizations can be incrementally computed in a bottom-up manner. This property not only allows efficient computation of group sizes, but also provides a terminating condition for further generalizations, leading to the generalization property:

*Observation* 5.2 (*Generalization Property*). Let $T'$ be a table not more specific than table $T$ on all attributes in *QID*. If $T$ is $k$-anonymous on *QID*, then $T'$ is also $k$-anonymous on *QID*.

The generalization property provides the basis for effectively pruning the search space of generalized tables. This property is essential for efficiently determining an optimal $k$-anonymization [LeFevre et al. 2005; Samarati 2001]. Consider a $qid$ in a table $T$. If $qid'$ is a generalization of $qid$ and $|qid| \geq k$, then $|qid'| \geq k$. Thus, if $T$ is $k$-anonymous, there is no need to generalize $T$ further because any further generalizations of $T$ must also be $k$-anonymous but with higher distortion, and therefore not optimal according to, for example, the minimal distortion metric *MD*. Although Incognito significantly outperforms the binary search in efficiency [Samarati 2001], the complexity of all three algorithms, namely MinGen, binary search, and Incognito, increases exponentially with the size of *QID*.

*K-Optimize.* Another algorithm called *K-Optimize* [Bayardo and Agrawal 2005] effectively prunes nonoptimal anonymous tables by modeling the search space using a set enumeration tree. Each node represents a $k$-anonymous solution. The algorithm assumes a totally ordered set of attribute values and examines the tree in a top-down manner, starting from the most general table, and prunes a node in the tree when none of its descendants could be a global optimal solution based on discernibility metric *DM* and classification metric *CM*. Unlike the above algorithms, K-Optimize employs the subtree generalization and record suppression schemes. It is the only efficient optimal algorithm that uses the flexible subtree generalization.

*5.1.2. Minimal Anonymization Algorithms.* The second family of algorithms produces a minimal $k$-anonymous table by employing a greedy search guided by a search metric. Being heuristic in nature, these algorithms find a minimally anonymous solution, but are more scalable than the previous family.

*$\mu$-argus.* The *$\mu$-argus* algorithm [Hundepool and Willenborg 1996] computes the frequency of all 3-value combinations of domain values, then greedily applies subtree generalizations and cell suppressions to achieve $k$-anonymity. Since the method limits the size of attribute combination, the resulting data may not be $k$-anonymous when more than 3 attributes are considered.

*Datafly.* Sweeney's [1998] *Datafly* system was the first $k$-anonymization algorithm scalable to handle real-life large data sets. It achieves $k$-anonymization by generating an array of $qid$ group sizes and greedily generalizing those combinations with less than $k$ occurrences based on a heuristic search metric *DA* that selects the attribute with the largest number of distinct values. Datafly employs full-domain generalization and record suppression schemes.

*Genetic.* Iyengar [2002] was among the first to aim at preserving classification information in $k$-anonymous data by employing a genetic algorithm with an incomplete stochastic search based on classification metric *CM* and a subtree generalization scheme. The idea is to encode each state of generalization as a "chromosome" and encode data distortion by a fitness function. The search process is a genetic evolution that converges to the fittest chromosome. Iyengar's experiments suggested that, by considering the classification purpose, the classifier built from the anonymous data produces lower classification error than the classifier built from the anonymous data using a general purpose metric. However, experiments also showed that this genetic algorithm is inefficient for large data sets.

*Bottom-Up Generalization.* To address the efficiency issue in $k$-anonymization, a bottom-up generalization algorithm was proposed in Wang et al. [2004] for finding a minimal $k$-anonymization for classification. The algorithm starts from the original

data that violates $k$-anonymity and greedily selects a generalization operation at each step according to a search metric similar to *ILPG* in Eq. (5). Each operation increases the group size according to the rollup property in Observation 5.1. The generalization process is terminated as soon as all groups have the minimum size $k$. To select a generalization operation, it first considers those that will increase the minimum group size, called *critical generalizations*, with the intuition that a loss of information should trade for some gain on privacy. When there are no critical generalizations, it considers other generalizations. Wang et al. [2004] showed that this heuristic significantly reduces the search space.

*Top-Down Specialization.* Instead of bottom-up, the *top-down specialization* (TDS) method [Fung et al. 2005, 2007] generalizes a table by specializing it from the most general state in which all values are generalized to the most general values of their taxonomy trees. At each step, TDS selects the specialization according to the search metric *IGPL* in Eq. (4). The specialization process terminates if no specialization can be performed without violating $k$-anonymity. The data on termination is a minimal $k$-anonymization according to the generalization property in Observation 5.2. TDS handles both categorical and numerical attributes in a uniform way, except that the taxonomy tree for a numerical attribute is grown on-the-fly as specializations are searched at each step.

Fung et al. [2008, 2009] further extended the $k$-anonymization algorithm to preserve the information for cluster analysis. The major challenge in anonymizing data for cluster analysis is the lack of class labels that could be used to guide the anonymization process. Fung et al.'s solution is to first partition the original data into clusters on the original data; convert the problem into the counterpart problem for classification analysis, where class labels encode the cluster information in the data; and then apply TDS to preserve $k$-anonymity and the encoded cluster information.

In contrast to the bottom-up approach [LeFevre et al. 2005; Samarati 2001; Wang et al. 2004], the top-down approach has several advantages. First, the user can stop the specialization process at *any time* and have a $k$-anonymous table. In fact, every step in the specialization process produces a $k$-anonymous solution. Second, TDS handles multiple *QID*s, which is essential for avoiding the excessive distortion suffered by a single high-dimensional *QID*. Third, the top-down approach is more efficient by going from the most generalized table to a more specific table. Once a group cannot be specialized further, all data records in the group can be discarded. In contrast, the bottom-up approach has to keep all data records until the end of computation. However, data publishers employing TDS may encounter the dilemma of choosing (multiple) *QID*, discussed in Section 2.1.

*Mondrian Multidimensional.* LeFevre et al. [2006a] presented a greedy top-down specialization algorithm for finding a minimal $k$-anonymization in the case of the multidimensional generalization scheme. This algorithm is very similar to TDS. Both algorithms perform a specialization on a value $v$ one at a time. The major difference is that TDS specializes in all *qid* groups containing $v$. In other words, a specialization is performed only if each specialized *qid* group contains at least $k$ records. In contrast, Mondrian performs a specialization on *one* qid group if each of its specialized *qid* groups contains at least $k$ records. Due to such a relaxed constraint, the resulting anonymous data in multidimensional generalization usually has a better quality than in single generalization. The trade-off is that multidimensional generalization is less scalable than other schemes due to the increased search space. Xu et al. [2006] showed that employing cell generalization could further improve the data quality. Although the multidimensional and cell generalization schemes cause less information loss, they suffer from the data exploration problem discussed in Section 3.

*5.1.3. Perturbation Algorithms.* This family of anonymization methods employs perturbation to deassociate the linkages between a target victim and a record while preserving some statistical information.

*Condensation.* Aggarwal and Yu [2008a, 2008b] presented a condensation method to thwart record linkages. The method first assigns records into multiple nonoverlapping groups in which each group has a size of at least $k$ records. For each group, extract some statistical information, such as sum and covariance, that suffices to preserve the mean and correlations across the different attributes. Then, for publishing, based on the statistical information, generate points for each group following the statistical characteristics of the group. This method does not require the use of taxonomy trees and can be effectively used in situations with dynamic data updates as in the case of data streams. As each new data record is received, it is added to the nearest group, as determined by the distance to each group centroid. As soon as the number of data records in the group equals $2k$, the corresponding group needs to be split into two groups of $k$ records each. The statistical information of the new group is then incrementally computed from the original group.

*r-Gather Clustering.* In a similar spirit, Aggarwal et al. [2006] proposed a perturbation method called *r-gather clustering*. This method partitions records into several clusters such that each cluster contains at least $r$ data points (i.e., records). Instead of generalizing individual records, this approach releases the cluster centers, together with their size, radius, and a set of associated sensitive values. To eliminate the impact of outliers, they relaxed this requirement to $(r, \epsilon)$*-gather clustering* so that at most $\epsilon$ fraction of data records in the data set can be treated as outliers for removal from the released data.

*Cross-Training Round Sanitization.* Recall from Section 2.1 that point $q$ $(c, t)$*-isolates* point $p$ if $B(q, c\delta_p)$ contains fewer than $t$ points in the table, where $B(q, c\delta_p)$ is a ball of radius $c\delta_p$ centered at point $q$. Chawla et al. [2005] proposed two sanitization (anonymization) techniques, *recursive histogram sanitization* and *density-based perturbation*, to prevent $(c, t)$-isolation.

Recursive histogram sanitization recursively divides original data into a set of subcubes according to local data density until all subcubes have no more than $2t$ data points. The method outputs the boundaries of the subcubes and the number of points in each subcube. However, this method cannot handle high-dimensional spheres and balls. Chawla et al. [2005] proposed an extension to handle high-dimensionality. Density-based perturbation, a variant of the one proposed by Agrawal and Srikant [2000], in which the magnitude of the added noise is relatively fixed, takes into consideration the local data density near the point that needs to be perturbed. Points in dense areas are perturbed much less than points in sparse areas. Although the privacy of the perturbed points is protected, the privacy of the points in the $t$-neighborhood of the perturbed points could be compromised because the sanitization radius itself could leak information about these points. To prevent such privacy leakage from $t$-neighborhood points, Chawla et al. [2005] further suggested a *cross-training round sanitization* method by combining recursive histogram sanitization and density-based perturbation. In cross-training round sanitization, a dataset is randomly divided into two subsets, $A$ and $B$. $B$ is sanitized using only recursive histogram sanitization, while $A$ is perturbed by adding Gaussian noise generated according to the histogram of $B$.

### 5.2. Algorithms for the Attribute Linkage Model

The following algorithms anonymize the data to prevent attribute linkages. They use the privacy models discussed in Section 2.2. Though their privacy models are different

from those of record linkage, many algorithms for attribute linkage are simple extensions from algorithms for record linkage.

The following algorithms adopt $\ell$-diversity as the privacy model. Recall that $\ell$-diversity requires every *qid* group to contain at least $\ell$ "well-represented" sensitive values.

*$\ell$-Diversity Incognito.* Machanavajjhala et al. [2006, 2007] modified the bottom-up Incognito [LeFevre et al. 2005] to identify an optimal $\ell$-diverse table. The $\ell$-Diversity Incognito operates based on the generalization property, similar to Observation 5.2, that $\ell$-diversity is nondecreasing with respect to generalization. In other words, generalizations help to achieve $\ell$-diversity, just as generalizations help achieve $k$-anonymity. Therefore, $k$-anonymization algorithms that employ full-domain and subtree generalization can also be extended into $\ell$-diversity algorithms.

*InfoGain Mondrian.* LeFevre et al. [2006b] proposed a suite of greedy algorithms to identify a minimally anonymous table satisfying $k$-anonymity and/or entropy $\ell$-diversity with the consideration of a specific data analysis task such as classification modeling multiple target attributes and query answering with minimal imprecision. Their top-down algorithms are similar to TDS [Fung et al. 2005], but LeFevre et al. [2006b] employed multidimensional generalization.

*Top-Down Disclosure.* Recall that a privacy template has the form $\langle QID \rightarrow s, h\rangle$, and states that the confidence of inferring the sensitive value $s$ from any group on $QID$ is no more than $h$. Wang et al. [2005, 2007] proposed an efficient algorithm to minimally suppress a table to satisfy a set of privacy templates. Their algorithm, called *Top-Down Disclosure* (*TDD*), iteratively discloses domain values starting from the table in which all domain values are suppressed. In each iteration, it discloses the suppressed domain value that maximizes the search metric *IGPL* in Eq. (4), and terminates the iterative process when a further disclosure leads to a violation of some privacy templates. This approach is based on the following key observation.

*Observation* 5.3 (*Disclosure Property*). Consider a privacy template $\langle QID \rightarrow s, h\rangle$. If a table violates the privacy template, so does any table obtained by disclosing a suppressed value [Wang et al. 2007].

This property ensures that the algorithm finds a minimally suppressed table. This property, and therefore the algorithm, is extendable to full-domain, subtree, and sibling generalization schemes, with the disclosure operation being replaced by the specialization operation. The basic observation is that the confidence in at least one of the specialized groups will be as large as the confidence in the general group. Based on a similar idea, Wong et al. [2006] employed the cell generalization scheme and proposed some greedy top-down and bottom-up methods to identify a minimally anonymous solution that satisfies $(\alpha, k)$-anonymity.

*$(k, e)$-Anonymity Permutation.* To achieve $(k, e)$-anonymity, Zhang et al. [2007] proposed an optimal permutation method to assign data records into groups together, so that the sum of error $E$ is minimized, where $E$, for example, could be measured by the range of sensitive values in each group. The optimal algorithm has time and space complexity in $O(n^2)$, where $n$ is the number of data records. $(k, e)$-anonymity is also closely related to a *range coding* technique, which is used in both process control [Rosen et al. 1992] and official statistics [Hegland et al. 1999]. In process control, range coding (also known as coarse coding) permits generalization by allowing the whole numerical area to be mapped to a set of groups defined by a set of boundaries, which is similar to the idea of grouping data records by ranges and keeping boundaries of each group for fast computation in $(k, e)$-anonymity. Hegland et al. [1999] also suggested handling large data sets as population census data, by dividing them into

generalized groups (blocks) and applying a computational model to each group. Any aggregate computation can hence be performed based on manipulation of individual groups. Similarly, $(k, e)$-anonymity exploits the group boundaries to efficiently answer aggregate queries.

*Personalized Privacy.* Refer to the requirement of personalized privacy discussed in Section 2.2. Xiao and Tao [2006b] proposed a greedy algorithm to achieve every record owner's privacy requirement in terms of a guarding node, as follows: initially, all *QID* attributes are generalized to the most general values, and the sensitive attributes remain ungeneralized. At each iteration, the algorithm performs a top-down specialization on a *QID* attribute and, for each *qid* group, performs cell generalization on the sensitive attribute to achieve the personalized privacy requirement; the breach probability of inferring any domain-sensitive values within the subtree of guarding nodes is below a certain threshold. Since the breach probability is nonincreasing with respect to generalization on the sensitive attribute and the sensitive values could possibly be generalized to the most general values, the generalized table found at every iteration is publishable without violating the privacy requirement, although a table with lower information loss *ILoss*, measured by Eq. (3), is preferable. When no better solution with lower *ILoss* is found, the greedy algorithm terminates and outputs a minimal anonymization. Since this approach generalizes the sensitive attribute, *ILoss* is measured on both *QID* and sensitive attributes.

### 5.3. Algorithms for the Table Linkage Model

The following algorithms aim at preventing table linkages, that is, preventing attackers from determining the presence or the absence of a target victim's record in a released table.

*Presence Algorithms SPALM and MPALM*: Recall that a generalized table $T'$ satisfies $(\delta_{min}, \delta_{max})$-*presence* (or simply $\delta$-presence) with respect to an external table $E$ if $\delta_{min} \leq P(t \in T|T') \leq \delta_{max}$ for all $t \in E$. To achieve $\delta$-presence, Nergiz et al. [2007] presented two anonymization algorithms, SPALM and MPALM. SPALM is an optimal algorithm that employs a full-domain single-dimensional generalization scheme. Nergiz et al. [2007] proved the anti-monotonicity property of $\delta$-presence with respect to full-domain generalization; if table $T$ is $\delta$-present, then a generalized version of $T'$ is also $\delta$-present. SPALM is a top-down specialization approach and exploits the anti-monotonicity property of $\delta$-presence to prune the search space effectively. MPALM is a minimal algorithm that employs a multidimensional generalization scheme, with complexity $O(|C||E|log_2|E|)$, where $|C|$ is the number of attributes in private table $T$ and $|E|$ is the number of records in the external table $E$. Their experiments showed that MPALM usually results in much lower information loss than SPALM because MPALM employs a more flexible generalization scheme.

### 5.4. Minimality Attack on Anonymous Data

Most privacy models assume that the attacker knows the *QID* of a target victim and/or the presence of the victim's record in the published data. In addition to this background knowledge, the attacker can possibly determine the privacy requirement (e.g., 10-anonymity or 5-diversity), the anonymization operations (e.g., subtree generalization scheme) to achieve the privacy requirement, and the detailed mechanism of an anonymization algorithm. The attacker can possibly determine the privacy requirement and anonymization operations by examining the published data, or its documentation, and learn the mechanism of the anonymization algorithm by, for example, reading research papers. Wong et al. [2007] pointed out that such additional

**Table V.** Example Illustrating Minimality Attacks

(a) Original patient table

| Job | Sex | Disease |
|---|---|---|
| Engineer | Male | HIV |
| Engineer | Male | HIV |
| Lawyer | Male | Flu |
| Lawyer | Male | Flu |
| Lawyer | Male | Flu |
| Lawyer | Male | Flu |
| Lawyer | Male | Flu |

(b) Published anonymous table

| Job | Sex | Disease |
|---|---|---|
| Professional | Male | HIV |
| Professional | Male | Flu |
| Professional | Male | Flu |
| Professional | Male | Flu |
| Professional | Male | Flu |
| Professional | Male | Flu |
| Professional | Male | HIV |

(c) External table

| Name | Job | Sex |
|---|---|---|
| Andy | Engineer | Male |
| Calvin | Lawyer | Male |
| Bob | Engineer | Male |
| Doug | Lawyer | Male |
| Eddy | Lawyer | Male |
| Fred | Lawyer | Male |
| Gabriel | Lawyer | Male |

background knowledge can lead to extra information that facilitates an attack to compromise data privacy. This is called the *minimality attack*.

Many anonymization algorithms discussed in this section follow an implicit minimality principle. For example, when a table is generalized from bottom-up to achieve $k$-anonymity, the table is not further generalized once it minimally meets the $k$-anonymity requirement. Minimality attack exploits this minimality principle to reverse the anonymization operations and filter out the impossible versions of the original table [Wong et al. 2007]. The following example illustrates a minimality attack on confidence bounding [Wang et al. 2007].

*Example* 5.1. Consider the original patient Table V(a), the anonymous Table V(b), and an external Table V(c) in which each record has a corresponding original record in Table V(a). Suppose the attacker knows that the confidence bounding requirement is $\langle \{Job, Sex\} \rightarrow HIV, 60\% \rangle$. With the minimality principle, the attacker can infer that Andy and Bob have $HIV$ based on the following reason: From Table V(a), $qid = \langle Lawyer, Male \rangle$ has 5 records, and $qid = \langle Engineer, Male \rangle$ has 2 records. Thus, $\langle Lawyer, Male \rangle$ in the original table must already satisfy $\langle \{Job, Sex\} \rightarrow HIV, 60\% \rangle$ because even if both records with $HIV$ have $\langle Lawyer, Male \rangle$, the confidence for inferring $HIV$ is only $2/5 = 40\%$. Since a subtree generalization has been performed, $\langle Engineer, Male \rangle$ must be the $qid$ that has violated the 60% confidence requirement on $HIV$, and that is possible only if both records with $\langle Engineer, Male \rangle$ have a disease value of $HIV$.

To thwart minimality attack, Wong et al. [2007] proposed a privacy model, called *m-confidentiality*, that limits the probability of the linkage from any record owner to any sensitive value set in the sensitive attribute. Wong et al. [2007] also showed that this type of minimality attack is applicable to both optimal and minimal anonymization algorithms that employ generalization, suppression, anatomization, or permutation to achieve privacy models, including, but not limited to, $\ell$-diversity [Machanavajjhala et al. 2007]; $(\alpha, k)$-anonymity [Wong et al. 2006]; $(k, e)$-anonymity [Zhang et al. 2007]; personalized privacy [Xiao and Tao 2006b]; anatomy [Xiao and Tao 2006a]; $t$-closeness [Li et al. 2007]; $m$-invariance [Xiao and Tao 2007]; and $(X, Y)$-privacy [Wang and Fung 2006]. To avoid minimality attack on $\ell$-diversity, Wong et al. [2007] proposed to first $k$-anonymize the table, then, for each $qid$ group in the $k$-anonymous table that violates $\ell$-diversity, their method distorts the sensitive values to satisfy $\ell$-diversity.

### 5.5. Algorithms for the Probabilistic Attack Model

Many algorithms for achieving the probabilistic privacy models studied in Section 2.4 employ perturbation methods, so they do not suffer from the problem of minimality attacks. The perturbation algorithms are nondeterministic; therefore, the anonymization operations are nonreversible. The perturbation algorithms for the probabilistic attack model can be divided into two groups. The first group is local perturbation [Agrawal

and Haritsa 2005], which assumes that a record owner does not trust anyone except himself and perturbs his own data record by adding noise before submission to the untrusted data publisher. The second group is to perturb all records together by a trusted data publisher, which is the data publishing scenario studied in this survey. Although the methods in the first group are also applicable to the second by adding noise to each individual record, Rastogi et al. [2007] and Dwork [2007] demonstrated that the information utility can be improved with a stronger lower bounds by assuming a trusted data publisher who has the capability to access all records and exploit the overall distribution to perturb the data, rather than perturbing the records individually.

A number of PPDP methods [Agrawal and Srikant 2000; Zhang et al. 2005] have been proposed for preserving classification information with randomization. Agrawal and Srikant [2000] presented a randomization method for decision tree classification with the use of the aggregate distributions reconstructed from the randomized distribution. The general idea is to construct the distribution separately from the different classes. Then, a *special* decision tree algorithm is developed to determine the splitting conditions based on the relative presence of the different classes, derived from the aggregate distributions. Zhang et al. [2005] presented a randomization method for a naive Bayes classifier. The major shortcoming of this approach is that *ordinary* classification algorithms will not work on this randomized data.

The statistics community conducts substantial research in the disclosure control of statistical information and aggregate query results [Cox 1980; Chawla et al. 2005; Duncan and Fienberg 1998; Matloff 1988; Ozsoyoglu and Su 1990]. The goal is to prevent attackers from obtaining sensitive information by correlating different published statistics. Cox [1980] proposed the $k$%-dominance rule which suppresses a sensitive cell if the values of two or three entities in the cell contribute more than $k$% of the corresponding SUM statistic. The proposed mechanisms include query size and query overlap control, aggregation, data perturbation, and data swapping. Nevertheless, such techniques are often complex and difficult to implement [Farkas and Jajodia 2003], or address privacy threats that are unlikely to occur. There are some decent surveys [Adam and Wortman 1989; Domingo-Ferrer 2001; Moore 1996; Zayatz 2007] in the statistics community.

*$\epsilon$-Differential Additive Noise*: One representative work that aims to thwart probabilistic attack is *differential privacy* [Dwork 2006]; its definition can be found in Section 2.4. Dwork [2006] proposed an additive noise method to achieve $\epsilon$-differential privacy. The added noise is chosen over a scaled symmetric exponential distribution with variance $\sigma^2$ in each component, and $\sigma \geq \epsilon/\Delta f$, where $\Delta f$ is the maximum difference of outputs of a query $f$ caused by the removal or addition of a single data record. Machanavajjhala et al. [2008] proposed a revised version of differential privacy, called *probabilistic differential privacy*, that yields a practical privacy guarantee for synthetic data generation. The idea is to first build a model from the original data, then sample points from the model to substitute for original data. The key idea is to filter unrepresentative data and shrink the domain. Other algorithms [Blum et al. 2005; Dinur and Nissim 2003; Dwork et al. 2006; Dwork and Nissim 2004] have been proposed to achieve differential privacy; refer to Dwork [2008] for a decent survey on the recent developments in this line of privacy model.

*$\alpha\beta$ Algorithm*: Recall that $(d, \gamma)$-privacy in Section 2.4 bounds the difference of $P(r)$ and $P(r|T)$, where $P(r)$ and $P(r|T)$ are the prior probability and the posterior probability of the presence of a victim's record in the data table $T$ before and after examining the published table $T$. To achieve $(d, \gamma)$-Privacy, Rastogi et al. [2007] proposed a perturbation method, called $\alpha\beta$ algorithm, consisting of two steps. The first step is to select a subset of records from the original table $D$ with probability $\alpha + \beta$ and insert them

into the data table $T$, which is to be published. The second step is to generate some counterfeit records from the domain of all attributes. If the counterfeit records are not in the original table $D$, then insert them into $T$ with probability $\beta$. Hence, the resulting perturbed table $T$ consists of both records randomly selected from the original table and counterfeit records from the domain. The number of records in the perturbed data could be larger than the original data table, in comparison with FRAPP [Agrawal and Haritsa 2005] which has a fixed table size. The drawback of inserting counterfeits is that the released data can no longer preserve the truthfulness of the original data at the record level, which is important in some applications, as explained in Section 1.1.

## 6. EXTENDED SCENARIOS

All the work discussed so far focuses on anonymizing and publishing a single release. In practical applications, data publishing is more complicated. For example, the same data may be published several times. Each time, the data is anonymized differently for different purposes, or the data is published incrementally as new data is collected. In this section, we consider such extended publishing scenarios.

### 6.1. Multiple Release Publishing

Different data recipients may be interested in different attributes of a data table. Suppose there is a person-specific data table $T(Job, Sex, Age, Race, Disease, Salary)$. A data recipient (for example, a pharmaceutical company) is interested in classification modeling the target attribute *Disease* with attributes $\{Job, Sex, Age\}$. Another data recipient (such as a social service department) is interested in clustering analysis on $\{Job, Age, Race\}$. One approach is to publish a single release on $\{Job, Sex, Age, Race\}$ for both purposes. A drawback is that information is released unnecessarily, in that neither of the two purposes needs all four attributes, which makes it is more vulnerable to attacks. Moreover, if the information needed in the two cases is different, the data anonymized in a single release may not be good for either of the two cases. A better approach is to anonymize and publish a customized release for each data mining purpose; each release is anonymized to best address the specific purpose. Given that both releases are published, there is a possibility that the data recipients have access to both releases; it is difficult to prevent them from colluding with each other behind the scenes. In particular, an attacker can combine attributes from the two views to form a sharper *QID* that contains attributes from both views. The following example illustrates the *join attack* in multiple releases.

*Example* 6.1. Consider the data in Table VI(a). Suppose that the data publisher releases one projection view $T_1$ to one data recipient and releases another projection view $T_2$ to another data recipient. Both views are from the same underlying patient table. Further suppose that the data publisher does not want $\{Age, Birthplace\}$ to be linked to *Diseaset*. When $T_1$ and $T_2$ are examined separately, the $Age = 40$ group and the $Birthplace = France$ group have size 2. However, by joining $T_1$ and $T_2$ using $T_1.Job = T_2.Job$, an attacker can uniquely identify the record owner in the $\{40, France\}$ group, thus linking $\{Age, Birthplace\}$ to *Disease* without difficulty. Moreover, the join reveals the inference $\{30, US\} \rightarrow Cancer$ with 100% confidence for the record owners in the $\{30, US\}$ group. Such an inference cannot be made when $T_1$ and $T_2$ are examined separately [Wang and Fung 2006].

Several works measured information disclosure arising from linking two or more views. Yao et al. [2005] presented a method for detecting $k$-anonymity violation on a

**Table VI.** Multiple/Sequential Release

(a) $T_1$

| Age | Job | Class |
|---|---|---|
| 30 | Lawyer | c1 |
| 30 | Lawyer | c1 |
| 40 | Carpenter | c2 |
| 40 | Electrician | c3 |
| 50 | Engineer | c4 |
| 50 | Clerk | c4 |

(b) $T_2$

| Job | Birthplace | Disease |
|---|---|---|
| Lawyer | US | Cancer |
| Lawyer | US | Cancer |
| Carpenter | France | HIV |
| Electrician | UK | Cancer |
| Engineer | France | HIV |
| Clerk | US | HIV |

(c) The join of $T_1$ and $T_2$

| Age | Job | Birthplace | Disease | Class |
|---|---|---|---|---|
| 30 | Lawyer | US | Cancer | c1 |
| 30 | Lawyer | US | Cancer | c1 |
| 40 | Carpenter | France | HIV | c2 |
| 40 | Electrician | UK | Cancer | c3 |
| 50 | Engineer | France | HIV | c4 |
| 50 | Clerk | US | HIV | c4 |
| 30 | Lawyer | US | Cancer | c1 |
| 30 | Lawyer | US | Cancer | c1 |

**Table VII.** Marginals

(a) *Job* marginal

| Job | Count |
|---|---|
| Engineer | 2 |
| Lawyer | 1 |
| Writer | 2 |
| Dancer | 2 |

(b) *Sex* marginal

| Sex | Count |
|---|---|
| Male | 3 |
| Female | 4 |

set of views, each view was obtained from a projection and selection query; they also considered functional dependency as prior knowledge.

In addition to the anonymous base table, Kifer and Gehrke [2006] proposed increasing the utility of published data by releasing several anonymous *marginals* that are essentially duplicate preserving projection views. For example, Table VII(a) and Table VII(b) are the *Job* and *Sex* marginals for the $k$-anonymous base Table II(c). The availability of additional marginals (views) provides additional information for data mining, but also poses new privacy threats. For example, if a combination of attribute values has a low count, it can be used as *QID* to reveal sensitive attributes in other databases. Thus, Kifer and Gehrke [2006] extended $k$-anonymity and $\ell$-diversity for marginals and presented a method to check whether published marginals violate the privacy requirement on the anonymous base table.

Barak et al. [2007] also studied the privacy threats caused by marginals, but along the lines of differential privacy [Dwork 2006]. Their primary contribution was providing a formal guarantee to preserve *all* the privacy, accuracy, and consistency in the published marginals. Accuracy bounds the difference between the original marginals and published marginals. Consistency ensures that there exists a contingency table whose marginals equal the published marginals. Instead of adding noise to the original data records at the cost of accuracy, or adding noise to the published marginals at the cost of consistency, they have proposed transforming the original data into the *Fourier* domain, applying differential privacy to the transformed data by perturbation, and employing linear programming to obtain a non-negative contingency table based on the given Fourier coefficients.

### 6.2. Sequential Release Publishing

In the multiple release publishing scenario, several releases, for different purposes, are published at one time. In some other scenarios, the data is released continuously and sequentially as new information becomes available. Consider the problem of *sequential anonymization* [Wang and Fung 2006]: a data publisher has *previously* released $T_1, \ldots, T_{p-1}$ and *now* wants to publish the next release $T_p$, where all $T_i$ are projections of the same underlying table, and each *individual release*, not the join, serves a data mining purpose. The data publisher wants to prevent record and attribute linkages through the join of $T_1, \ldots, T_p$. This requirement is specified by a privacy model such as $(X, Y)$-privacy (Section 2.2) on the join of all releases because the attacker has access to all releases. Unlike the multiple release publishing scenario, the previous releases $T_1, \ldots, T_{p-1}$ have been published and, therefore, cannot be modified. Any attempt at prevention of privacy violation has to rely on anonymizing the next release $T_p$.

To address the sequential anonymization problem, Wang and Fung [2006] introduced the *lossy join*, a negative property in relational database design, as a way to hide the join relationship among releases. A lossy join of $T_1$ and $T_2$ will result in a table containing some records that are not original records in the underlying tables $T_1$ and $T_2$. Such records are the result of matching some records in $T_1$ and $T_2$ that belong to different owners. The next example illustrates this point.

*Example* 6.2. Example 6.1 shows that the record owner in the $\{40, \mathit{France}\}$ group becomes uniquely identifiable, thus revealing his/her contracted disease, after the join of $T_1$ and $T_2$ in Table VI. In fact, the join is a double-edged sword, in that a lossy join could also weaken identification. For example, after the join, the $\{30, \mathit{US}\}$ group has size 4 because the records for different owners are matched (i.e., the last two records in the join table), whereas $T_1$ and $T_2$ are examined separately, both $\mathit{Age} = 30$ group and $\mathit{Birthplace} = \mathit{US}$ group have a smaller size.

A join attack depends critically on matching the records in $T_1$ and $T_2$ that represent the *same* record owner; therefore, a lossy join, which matches records of different record owners, can be used to combat the join attack. To make the join of $T_1$ and $T_2$ lossy, Wang and Fung [2006] proposed generalizing the join attributes in $T_2$ (recall that $T_1$ has been published and cannot be modified) so that a generalized record in $T_2$ will match more records in $T_1$. For example, for the join attribute *Job*, all records in $T_2$ generalized to *Professional* will match all records in $T_1$ that contain *Professional* or an ancestor or a descendant of *Professional* in the taxonomy of *Job*. Intuitively, two records, one in $T_1$ and one in $T_2$, match if their *Job* values are on the same generalization path in the taxonomy of *Job*. Note that this match condition is more relaxed than the traditional equality join which requires an exact match.

To satisfy a given requirement on $(X, Y)$-privacy, the anonymization algorithm generalizes $T_2$ on the attributes $X \cap \mathit{att}(T_2)$, where $\mathit{att}(T_2)$ denotes the set of attributes in $T_2$. A top-down specialization process is employed to iteratively specialize $T_2$ on $X \cap \mathit{att}(T_2)$ starting from the most general state of $T_2$. Recall from Section 2.2 that $(X, Y)$-privacy is composed of $(X, Y)$-anonymity and confidence bounding defined on the join of $T_1$ and $T_2$. Significantly, the *generalization property* holds in $(X, Y)$-privacy: for the subtree generalization scheme, $(X, Y)$-anonymity is nonincreasing and confidence bounding is nondecreasing with respect to a specialization on $T_2$ on $X \cap \mathit{att}(T_2)$. Essentially, this property means that if any of these requirements is violated, it remains violated after a specialization. Therefore, the top-down specialization approach can prune the remaining search space and efficiently identify a minimal anonymization of $T_2$.

**Table VIII.** Continuous Data Publishing

(a) 2-diverse $T_1$

| Job | Sex | Disease |
|---|---|---|
| Professional | Female | Cancer |
| Professional | Female | Diabetes |
| Artist | Male | Fever |
| Artist | Male | Cancer |

(b) 2-diverse $T_2$ after an insertion to $T_1$

| Job | Sex | Disease |
|---|---|---|
| Professional | Female | Cancer |
| Professional | Female | Diabetes |
| Professional | Female | HIV |
| Artist | Male | Fever |
| Artist | Male | Cancer |

(c) 2-diverse $T_2$ after a deletion from and an insertion to $T_1$

| Job | Sex | Disease |
|---|---|---|
| Professional | Female | Cancer |
| Professional | Female | Fever |
| Artist | Male | Fever |
| Artist | Male | Cancer |

This sequential anonymization problem was briefly discussed in some pioneering work on $k$-anonymity, but none provided a practical solution. For example, Samarati and Sweeney [1998b] suggested to $k$-anonymize all potential join attributes as the *QID* in the next release $T_p$. Sweeney [2002a] suggested generalizing $T_p$ based on the previous releases $T_1, \ldots, T_{p-1}$ to ensure that all values in $T_p$ are not more specific than in any $T_1, \ldots, T_{p-1}$. Both solutions suffer from monotonically distorting the data in a later release. The third solution is to release a "complete" cohort in which all potential releases are anonymized at one time, after which no additional mechanism is required. This requires predicting future releases. The "under-prediction" means that there will be no room for additional releases and the "over-prediction" means there will be unnecessary data distortion. Also, this solution does not accommodate the new data added at a later time.

### 6.3. Continuous Data Publishing

In the model of continuous data publishing, the data publisher has previously published $T_1, \ldots, T_{p-1}$ and now wants to publish $T_p$, where $T_i$ is an updated release of $T_{i-1}$ with record insertions and/or deletions. The problem assumes that all records for the same individual remain the same in all releases. Even though each release $T_1, \ldots, T_p$ is individually anonymous, the privacy requirement could be compromised by comparing different releases and eliminating some possible sensitive values for a victim. This problem assumes that the data is dynamically updated, unlike the sequential anonymization problem which assumes all data is static and is available at the time of release. Furthermore, this problem assumes all releases share the same database schema, while the sequential problem assumes all releases are projections of the same underlying data table.

This continuous data publishing problem assumes that the attacker knows the timestamp and *QID* of the victim, so the attacker knows exactly which releases contain the victim's data record. The following examples show the privacy threats caused by record insertions and deletions.

*Example* 6.3. Let Table VIII(a) be the first release $T_1$. Let Table VIII(b) be the second release $T_2$ after inserting a new record. Both $T_1$ and $T_2$ satisfy 2-diversity independently. Suppose the attacker knows that a female lawyer, Alice, has a record in $T_2$ but not in $T_1$, based on the timestamp that Alice was admitted to a hospital. From $T_2$, the attacker can infer that Alice must have contracted either *Flu*, *Fever*, or *HIV*. By comparing $T_2$ with $T_1$, the attacker can identify that the first two records in $T_2$ must be old records from $T_1$ and, thus, infer that Alice must have contracted *HIV*.

*Example* 6.4. Let Table VIII(a) be the first release $T_1$. Let Table VIII(b) be the second release $T_2$ after deleting the record $\langle Professional, Female, Diabetes\rangle$ and inserting a new record $\langle Professional, Female, Fever\rangle$. Both $T_1$ and $T_2$ satisfy 2-diversity independently. Suppose the attacker knows that a female engineer, Beth, must be in both $T_1$ and $T_2$. From $T_1$, the attacker can infer that Beth must have contracted either *Cancer* or *Diabetes*. Since $T_2$ contains no *Diabetes*, the attacker can infer that Beth must have contracted *Cancer*.

Byun et al. [2006] were the pioneers who proposed an anonymization technique that enables privacy-preserving continuous data publishing after new records have been inserted. Specifically, it guarantees every release to satisfy $\ell$-diversity, which requires each *qid* group contain at least $\ell$ distinct sensitive values. Since this instantiation of $\ell$-diversity does not consider the frequencies of sensitive values, an attacker could still confidently infer a sensitive value of a victim if the value occurs frequently in a *qid* group. Thus, this instantiation cannot prevent attribute linkage attacks.

Byun et al. [2006] addressed the threats caused by record insertions but not deletions, so the current release $T_p$ contains all records in previous releases. The algorithm inserts new records into the current release $T_p$ only if two privacy requirements remain satisfied after the insertion: (1) $T_p$ is $\ell$-diverse; (2) given any previous release $T_i$ and the current release $T_p$ together, there are at least $\ell$ distinct sensitive values in the remaining records that could potentially be the victim's record. This requirement can be verified by comparing the difference and intersection of the sensitive values in any two "comparable" *qid* groups in $T_i$ and $T_p$. The algorithm prefers to specialize $T_p$ as much as possible to improve the data quality, provided that the two privacy requirements are satisfied. If the insertion of some new records would violate any of the privacy requirements, even after generalization, the insertions are delayed until later releases. Nonetheless, this strategy may sometimes run into a situation in which no new data could be released. Also, it requires a very large memory buffer to store those delayed data records.

Xiao and Tao [2007] proposed a new privacy notion called *m-invariance* and an anonymization method, addressing both record insertions and deletions. In this continuous data publishing model, a sequence of releases $T_1, \ldots, T_p$ is $m$-invariant if (1) every *qid* group in any $T_i$ contains at least $m$ records and all records in *qid* have different sensitive values; and (2) for any record $r$ with published lifespan $[x, y]$ where $1 \leq x, y \leq p$, $qid_x, \ldots, qid_y$ have the same set of sensitive values where $qid_x, \ldots, qid_y$ are the generalized *qid* groups containing $r$ in $T_x, \ldots, T_y$. The rationale of $m$-invariance is that, if a record $r$ has been published in $T_x, \ldots, T_y$, then all *qid* groups containing $r$ must have the same set of sensitive values. This will ensure the intersection of sensitive values over all such *qid* groups does not reduce the set of sensitive values compared to each *qid* group.

Given a sequence of $m$-invariant $T_1, \ldots, T_{p-1}$, Xiao and Tao [2007] maintained a sequence of $m$-invariant $T_1, \ldots, T_p$ by minimally adding counterfeit data records and generalizing the current release $T_p$. A table with counterfeit records could no longer preserve the data truthfulness at the record level, which is important in some applications, as explained in Section 1.1.

Recently, Fung et al. [2008] showed a method to systematically quantify the exact number of records that can be "cracked" by comparing all $k$-anonymous releases. A record in a $k$-anonymous release is "cracked" if it is impossible to be a candidate record of the target victim. After excluding the cracked records from a release, a table may no longer be $k$-anonymous. In some cases, data records, with sensitive information of some victims, can even be uniquely identified from the releases. Fung et al. [2008] proposed a privacy requirement, called *BCF-anonymity*, to measure the true

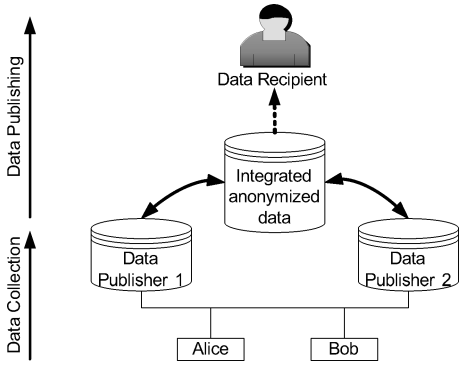


**Fig. 4**. Collaborative data publishing.

anonymity in a release after excluding the cracked records, and presented a generalization method to achieve *BCF*-anonymity without delaying record publication or inserting counterfeit reords. Some work misinterprets Fung et al. [2008] to mean that they allow only record insertion but not record deletion. Indeed, $T_1$ and $T_2$ are independent of each other (i.e., $T_2$ is not an insertion or deletion of $T_1$). Fung et al. [2008] anonymize ($T_1 \cup T_2$) as one release for utility on the whole data set. In contrast, all other work [Xiao and Tao 2007] anonymizes each $T_i$ independently, so the publishing model in Xiao and Tao [2007] does not benefit from new data because each $T_i$ is small, resulting in a large distortion. Bu et al. [2008] further relax the PPDP scenario and assume that the *QID* and sensitive values of a record owner could change in subsequent releases.

### 6.4. Collaborative Data Publishing

So far, we have considered only a single data publisher. In real-life data publishing, a single organization often does not hold the complete data. Organizations need to share data for mutual benefits or for publishing to a third party. For example, two credit card companies want to integrate their customer data for developing a fraud-detection system or for publishing to a bank. However, the credit card companies do not want to indiscriminately disclose their data to each other or to the bank for reasons such as privacy protection and business competitiveness. Figure 4 depicts this scenario, called *collaborative data publishing*, where several data publishers own different sets of attributes on the same set of records and want to publish the integrated data on all attributes. Say, publisher 1 owns $\{RecID, Job, Sex, Age\}$, and publisher 2 owns $\{RecID, Salary, Disease\}$, where *RecID*, such as the $SSN$, is the record identifier shared by all data publishers. They want to publish an integrated $k$-anonymous table on all attributes. Also, no data publisher should learn more specific information, owned by the other data publishers, than the information that appears in the final integrated table.

There are two obvious but insecure approaches. The first one is "integrate-then-generalize": that is, first integrate the tables and then generalize the integrated table using any single table $k$-anonymization method discussed in previous sections. This approach does not preserve privacy because the data publisher holding the integrated table will immediately know all the private information of all data publishers. The second approach is "generalize-then-integrate": that is, first generalize each table locally and then integrate the generalized tables. This approach does not work if the $k$-anonymity involves a global *QID* spanning two or more data publishers.

Wang et al. [2005] proposed an algorithm called *Top-Down Specialization for 2-Party* (TDS2P) to solve the collaborative publishing problem. Essentially, TDS2P produces the same final anonymous table as the integrate-then-generalize approach, but does not reveal local data until the data has been generalized to satisfy a given $k$-anonymity requirement. First, all data publishers generalize their attributes in *QID* to the most general value *ANY*. At each iteration, each data publisher identifies a local specialization that has the highest *IGPL*, measured by Eq. (4), based on her own data, and then collaboratively identifies the global specialization $w$ that has the maximum *IGPL* across all local specializations. The data publisher $P$, who owns the attribute of $w$, first performs the specialization $w$ on her own data. The other data publishers, who do not own the attribute of $w$, have to get the "instruction" from $P$ to partition their local data. The instruction, represented by $\langle GroupNo, RecID\rangle$, tells how records (identified by *RecID*) are specialized into different groups (identified by *GroupNo*). Repeat this process, and stop if any further specialization leads to a violation of $k$-anonymity. Mohammed et al. [2009] extended the idea to distributed data mashup applications.

Jiang and Clifton [2005, 2006] addressed a similar problem by using a cryptographic approach. First, each data publisher determines a locally $k$-anonymous table. Then, the intersection of *RecID*s for the *qid* groups in the two locally $k$-anonymous tables is determined. If the intersection size of each pair of the *qid* group is at least $k$, then the algorithm returns the join of the two locally $k$-anonymous tables that is globally $k$-anonymous; otherwise, further generalization is performed on both tables and the *RecID* comparison procedure is repeated. To prevent the other data publisher from learning more specific information than that appearing in the final integrated table through *RecID*, a commutative encryption scheme [Pohlig and Hellman 1978] is employed to encrypt the *RecID*'s for comparison. This scheme ensures the equality of two values encrypted in a different order on the same set of keys, that is, $E_{Key1}(E_{Key2}(RecID)) = E_{Key2}(E_{Key1}(RecID))$.

## 7. ANONYMIZING OTHER TYPES OF DATA

All the work discussed so far focuses on anonymizing relational and statistical data. What about other types of (nonrelational) data? Recent studies have shown that publishing transaction data, moving object data, and textual data may also result in privacy threats and sensitive information leakages. Below, we discuss the privacy threats, together with some privacy-preserving solutions, on these nonrelational data types.

### 7.1. High-Dimensional Transaction Data

Publishing high-dimensional data is part of the daily operations in commercial and public activity. A classic example of high-dimensional data is transaction databases. Each transaction corresponds to a record owner and consists of a set of items selected from a large universe. Examples of transactions are web queries, click streams, e-mails, market baskets, and medical notes. Such data often contains rich information and is an excellent source for data mining. Detailed transaction data provides an electronic image of a record owner's life, possibly containing sensitive information.

A recent case demonstrates the privacy threats caused by publishing transaction data: AOL released a database of query logs to the public for research purposes [Barbaro and Zeller 2006]. However, by examining query terms, AOL user No. 4417749 was traced back to Ms. Thelma Arnold, a 62-year-old widow who lives in Lilburn. Even if a query does not contain an address or name, a record owner (the AOL user in

this example) may still be re-identified from combinations of query terms that are adequately unique to the record owner. This scandal led not only to the disclosure of private information of AOL users, but also damaged data publishers' enthusiasm in offering anonymized transaction data for research purposes. Kumar et al. [2007] further showed that some token-hashed anonymous query logs could be cracked by inverting the hash function based on the co-occurrences of tokens in some other "reference" query logs. Clearly, there is a need for a proper anonymization method for transaction data.

Transaction data is usually high-dimensional. For example, Amazon.com has several million catalog items. Each dimension could be a potential *QID* attribute used for record or attribute linkages; therefore, employing traditional privacy models, such as $k$-anonymity, would require including all dimensions into a single *QID*. Due to the curse of high-dimensionality [Aggarwal 2005], it is very likely that lots of data has to be suppressed or generalized to the top-most values in order to satisfy $k$-anonymity, even if $k$ is small. Obviously, such anonymous data is useless for data analysis.

There are some recent studies on anonymizing high-dimensional data. Ghinita et al. [2008] proposed a permutation method whose general idea is to first group transactions with close proximity and then associate each group to a set of diversified sensitive values. In any real-life privacy attack, it is unlikely that the attacker would know *all* quasi-identifying attributes of a target victim due to the effort it would take to gather every piece of background knowledge. Thus, it is reasonable to bound the attacker's background knowledge in the privacy model. Terrovitis et al. [2008] proposed an algorithm to $k$-anonymize transactions by generalization. Xu et al. [2008] extended the traditional $k$-anonymity model by assuming that the attacker knows at most $m$ transaction items of the target victim. Specifically, the privacy model in Xu et al. [2008] ensures that (1) every itemset $I$ with size not greater than $m$ in the published table is shared by at least $k$ records; and (2) that the confidence of inferring the sensitive value $s$ from $I$ is less than a maximum confidence threshold $h$. Their results show that this relaxation can substantially improve data utility. In another work, Xu et al. [2008] consider preserving frequent itemsets as data utility. To deal with the scalability bottleneck caused by exponential explosion of itemsets, Xu et al. [2008] use sets of maximal and minimal itemsets, called *borders*, to represent the itemsets that violate the privacy requirement and the frequent itemsets. Both papers [Xu et al. 2008; Xu et al. 2008] use item suppression, instead of generalization, because the taxonomy trees for transaction data tend to be flat and fanout. In this case, employing generalization loses more information than employing item suppression.

Aggarwal and Yu [2007] formalized an anonymity model for the sketch-based approach, and utilized it to construct sketch-based privacy-preserving representations of the original data. The sketch-based approach [Alon et al. 1999] reduces the dimensionality of the data by generating a new representation with a much smaller number of features, where each one uses a different set of random weights to produce a weighted sum of the original feature values. This technique is quite effective for high-dimensional data sets, as long as the data is sparse. The sketch-based method provides privacy protection while allowing effective reconstruction of many aggregate distance measures. Therefore, it can be used for a variety of data mining algorithms such as clustering and classification.

### 7.2. Moving Object Data

Location-based services (LBS) are information services provided to mobile subscribers based on their specific physical locations. In recent years, a variety of location-based services has been developed due to increasing demand from subscribers. Although the

**Table IX.** Patient-Specific Path Table $T$

| Pid | Path | Disease | ... |
|---|---|---|---|
| 1 | $\langle a1 \rightarrow d2 \rightarrow b3 \rightarrow e4 \rightarrow f6 \rightarrow c7 \rangle$ | HIV | |
| 2 | $\langle b3 \rightarrow e4 \rightarrow f6 \rightarrow e8 \rangle$ | Flu | |
| 3 | $\langle b3 \rightarrow c7 \rightarrow e8 \rangle$ | Flu | |
| 4 | $\langle d2 \rightarrow f6 \rightarrow c7 \rightarrow e8 \rangle$ | Allergy | |
| 5 | $\langle d2 \rightarrow c5 \rightarrow f6 \rightarrow c7 \rangle$ | HIV | |
| 6 | $\langle c5 \rightarrow f6 \rightarrow e9 \rangle$ | Allergy | |
| 7 | $\langle d2 \rightarrow c5 \rightarrow c7 \rightarrow e9 \rangle$ | Fever | |
| 8 | $\langle f6 \rightarrow c7 \rightarrow e9 \rangle$ | Fever | |

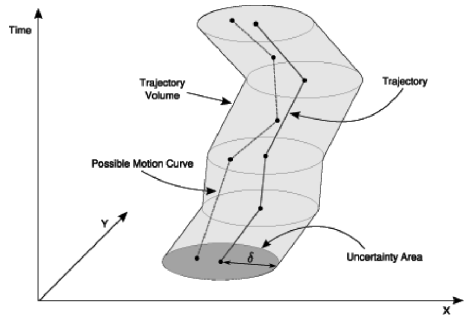


**Fig. 5**. Time and spatial trajectory volume [Abul et al. 2008].

advancement of telecommunication technology has improved our quality of life, research has shown that 24% of potential LBS users are seriously concerned about the privacy implications of disclosing their locations in conjunction with other personal data [Beinat 2001]. Moving object data poses new challenges to traditional database, data mining, and privacy-preserving technologies due to its unique characteristics: it is time-dependent, location-dependent, and is generated in large volumes of high-dimensional stream data. The following example shows the privacy threats caused by publishing moving object data.

*Example* 7.1. A hospital wants to release the patient-specific path table, Table IX, to a third party for data analysis. Explicit identifiers, such as patient names and *Pid*, have been removed. Each record contains the *moving path* of a patient in the hospital and some patient-specific (sensitive) information, for example, contracted diseases. A moving path contains a sequence of *pairs* $(loc_i t_i)$ indicating the patient's visited location $loc_i$ at timestamp $t_i$. For example, *Pid*#3 has a path $\langle b3 \rightarrow c7 \rightarrow e8 \rangle$, meaning that the patient has visited locations $b$, $c$, and $e$ at timestamps 3, 7, and 8, respectively.

An attacker seeks to perform record and/or attribute linkages by using the moving path as *QID* for matching. (1) *Record linkage*: suppose the attacker knows that the target victim, Alice, has visited $e$ and $c$ at timestamps 4 and 7, respectively. Alice's record, together with her sensitive value (HIV in this case), can be uniquely identified because *Pid*#1 is the *only* record that contains $e4$ and $c7$. (2) *Attribute linkage*: suppose the attacker knows that another target victim, Bob, has visited $d2$ and $f6$, matching (*Pid*#1,4,5), the attacker can infer that Bob has HIV with $2/3 = 67\%$ confidence.

There are a few recent works on anonymizing moving objects. Abul et al. [2008] extended the traditional $k$-anonymity model to anonymize a set of moving objects. The intuition is to have at least $k$ moving objects appearing within the radius $\delta$ of the path of every moving object in the same period of time, as depicted in Figure 5. In addition to the traditional anonymization operations discussed in Section 3, Abul et al. [2008] also

explored *space translation* by adding noise to the original paths so that more objects appear at the same time and spatial trajectory volume. Terrovitis and Mamoulis [2008] assumed that the locations are sensitive information, and that the attacker will attempt to infer some sensitive locations visited by the target victim which are unknown to the attacker. Malin and Airoldi [2006] studied the privacy threats in location-based data in the hospital environment.

Fung et al. [2009] presented the first work to anonymize high-dimensional RFID moving object data. Their proposed privacy model, *LKC-privacy*, ensures that every RFID moving path with length not greater than $L$ is shared by least $K-1$ other moving paths, and the confidence in inferring any prespecified sensitive value is not greater than $C$.

Papadimitriou et al. [2007] studied the privacy issue in publishing time-series data and examined the trade-offs between time-series compressibility and partial information hiding and their fundamental implications for how one should introduce uncertainty about individual values by perturbing them. The study found that by making the perturbation "similar" to the original data, we can both preserve the structure of the data better, and simultaneously make breaches harder. However, as data becomes more compressible, a fraction of the uncertainty can be removed if true values are leaked, revealing how they were perturbed.

### 7.3. Textual Data

Most previous work focused on anonymizing the structural or semistructural data. What about the unstructural data, such as text documents? Saygin et al. [2006] describes implicit and explicit privacy threats in text document repositories. Sanitization of text documents involves removing sensitive information or removing potential linking information that can associate an individual person to the sensitive information in a document. This research direction is in its infancy.

Kokkinakis and Thurin [2007] implemented a system for automatically anonymizing hospital discharge letters by identifying and deliberately removing all phrases from clinical text that satisfy some predefined types of sensitive entities. The identification phase is achieved by collaborating with an underlying generic named entity recognition system.

Instead of simply removing phases containing predefined types of sensitive entities, Chakaravarthy et al. [2008] presented the *ERASE* system to sanitize a document with the least distortion. External knowledge is required to associate a database of entities with their context. *ERASE* prevents disclosure of protected entities by removing certain terms of their context so that no protected entity can be inferred from the remaining document text. *k-safety*, in the same spirit of $k$-anonymity, is thereafter defined. A set of terms is *k-safe* if its intersection with every protected entity contains at least $k$ entities. Then the proposed problem is to find the maximum cardinality subset of a document satisfying $k$-safety. Chakaravarthy et al. [2008] proposed and evaluated both a global optimal algorithm and an efficient greedy algorithm to achieve $k$-safety.

## 8. PRIVACY-PRESERVING TECHNIQUES IN OTHER DOMAINS

### 8.1. Interactive Query Model

Closely related, but orthogonal to PPDP, is the extensive literature on inference control in multilevel secure databases [Farkas and Jajodia 2003; Jajodia and Meadows 1995]. Attribute linkages are identified and eliminated either at the database design phase [Goguen and Meseguer 1984; Hinke 1988; Hinke et al. 1995], by

**Table X.** Interactive Query Model

(a) Original examination data

| ID | University | Department | Score |
|---|---|---|---|
| 1 | Concordia | CS | 92 |
| 2 | Simon Fraser | EE | 91 |
| 3 | Concordia | CS | 97 |
| 4 | Illinois | CS | 96 |

(b) Added one new record

| ID | University | Department | Score |
|---|---|---|---|
| 1 | Concordia | CS | 92 |
| 2 | Simon Fraser | EE | 91 |
| 3 | Concordia | CS | 97 |
| 4 | Illinois | CS | 96 |
| 5 | Illinois | CS | 99 |

modifying the schemes and meta-data, or during the interactive query time [Denning 1985; Thuraisingham 1987], by restricting and modifying queries. These techniques, which focus on query database-answering, are not readily applicable to PPDP, where the data publisher may not have sophisticated database management knowledge, or does not want to provide an interface for database query. A data publisher, such as a hospital, has no intention of being a database server; answering database queries is not part of its normal business. Therefore, query-answering is quite different from the PPDP scenarios studied in this survey. Here, we briefly discuss the interactive query model.

In the interactive query model, the user can submit a sequence of queries based on previously received query results. Although this query model could improve the satisfaction of the data recipients' information needs [Dwork et al. 2006], the dynamic nature of queries makes the returned results even more vulnerable to attack, as illustrated in the following example. (Refer to Blum et al. [2005, 2008], Dwork [2008], Dinur and Nissim [2003] for more privacy-preserving techniques on the interactive query model).

*Example* 8.1. Suppose that an examination center allows a data miner to access its database, Table X(a), for research purposes. The attribute *Score* is sensitive. An attacker wants to identify the *Score* of a target victim, Bob, who is a student at the computer science department at Illinois. The attacker can first submit the query:

$Q1$: COUNT (*University* = *Illinois*) AND (*Department* = *CS*)

Since the count is 1, the attacker can determine Bob's *Score* = 96 by the following query:

$Q2$: AVERAGE *Score* WHERE (*University* = *Illinois*) AND (*Department* = *CS*).

Suppose that the data publisher has inserted a new record as shown in Table 9(b). Now the attacker tries to identify another victim by resubmitting query $Q1$. Since the answer is 2, the attacker knows another student at the computer science department at Illinois took this exam and can then submit the query:

$Q3$: SUM *Score* WHERE (*University* = *Illinois*) AND (*Department* = *CS*)

Benefiting from this update, the attacker can learn the *Score* of the new record by calculating $Q3 - Q2 = 99$.

Query auditing has a long history in statistical disclosure control. It can be broadly divided into two categories: *online auditing* and *offline auditing*.

*Online auditing*: The objective of online query auditing is to detect and deny queries that violate privacy requirements. Miklau and Suciu [2004] measured information disclosure of a view set, $V$, with respect to a secret view $S$. $S$ is secure if publishing $V$ does not alter the probability of inferring the answer to $S$. Deutsch and Papakonstantinou [2005] studied whether a new view disclosed more information than the existing views with respect to a secret view. To put the data publishing scenario considered in this survey in their terms: the anonymous release can superficially be considered as the "view" and the underlying data can be considered as the "secret query." However, the two problems have two major differences: First, the anonymous release is

obtained by anonymization operations, not by conjunctive queries as in Deutsch and Papakonstantinou [2005] and Miklau and Suciu [2004]. Second, the publishing scenarios employ anonymity as the privacy measure, whereas Miklau and Suciu [2004] and Deutsch and Papakonstantinou [2005] adopted the *perfect secrecy* for the security measure. The released data satisfies perfect secrecy if the probability that the attacker finds the original data after observing the anonymous data is the same as the probability or difficulty of getting the original data before observing the anonymous data.

Kenthapadi et al. [2005] proposed another privacy model, called *stimulatable auditing*, as an interactive query model. If the attacker has access to all previous query results, the method denies the new query if it leaks any information beyond what the attacker already knows. Although this "detect and deny" approach is practical, Kenthapadi et al. [2005] pointed out that the denials themselves may implicitly disclose sensitive information, making the privacy protection problem even more complicated. This motivates the offline query auditing.

*Offline auditing*: In offline query auditing [Evfimievski et al. 2008], the data recipients submit their queries and receive their results. The auditor checks if a privacy requirement has been violated *after* the queries have been executed. The data recipients have no access to the audit results and, therefore, the audit results do not trigger extra privacy threats as in the online mode. The objective of offline query audition is to check for compliance of privacy requirements, not to prevent the attackers from accessing the sensitive information.

### 8.2. Privacy Threats Caused by Data Mining Results

The release of data mining results or patterns could pose privacy threats. There are two broad research directions in this family.

The first direction is to anonymize the data so that sensitive data mining patterns cannot be generated. Aggarwal et al. [2006] pointed out that simply suppressing the sensitive values chosen by individual record owners is insufficient because an attacker can use association rules learnt from the data to estimate the suppressed values. They proposed a heuristic algorithm to suppress a minimal set of values to combat such attacks. Verykios et al. [2004] proposed algorithms for hiding sensitive association rules in a transaction database. The general idea is to hide one rule at a time by either decreasing its support or its confidence, achieved by removing items from transactions. Rules satisfying a specified minimum support and minimum confidence are removed. However, in the notion of anonymity, a rule applying to a small group of individuals (i.e., low support) presents a more serious threat because record owners from a small group are more identifiable.

The second direction is to directly anonymize the data mining patterns. Atzori et al. [2008] proposed the insightful suggestion that if the goal is to release data mining results, such as frequent patterns, then it is sufficient to anonymize the patterns rather than the data. Their study suggested that anonymizing the patterns yields much better information utility than performing data mining on anonymous data. This opens up a new research direction for privacy-preserving patterns publishing. Kantarcioglu et al. [2004] defined an evaluation method to measure the loss of privacy due to releasing data mining results.

### 8.3. Privacy-Preserving Distributed Data Mining

Privacy-preserving distributed data mining (PPDDM) is a cousin to the research topic of privacy-preserving data publishing (PPDP). PPDDM assumes a scenario that multiple data holders want to collaboratively perform data mining on the union of their

data without revealing their sensitive information. PPDDM usually employs cryptographic solutions. Although the ultimate goal of both PPDDM and PPDP is to perform data mining, they have very different assumptions on data ownerships, attack models, privacy models, and solutions, so PPDDM is out of the scope of this survey. We refer readers interested in PPDDM to work by [Clifton et al. 2002; Kantarcioglu 2008; Pinkas 2002; Vaidya 2008; Wright et al. 2005].

## 9. SUMMARY AND FUTURE RESEARCH DIRECTIONS

Information sharing has become part of the routine activity of many individuals, companies, organizations, and government agencies. Privacy-preserving data publishing is a promising approach to information sharing, while preserving individual privacy and protecting sensitive information. In this survey, we reviewed the recent developments in the field. The general objective is to transform the original data into some anonymous form to prevent from inferring its record owners' sensitive information. We presented our views on the difference between privacy-preserving data publishing and privacy-preserving data mining, and gave a list of desirable properties of a privacy-preserving data publishing method. We reviewed and compared existing methods in terms of privacy models, anonymization operations, information metrics, and anonymization algorithms. Most of these approaches assumed a single release from a single publisher, and thus only protected the data up to the first release or the first recipient. We also reviewed several works on more challenging publishing scenarios, including multiple release publishing, sequential release publishing, continuous data publishing, and collaborative data publishing.

Privacy protection is a complex social issue, which involves policy-making, technology, psychology, and politics. Privacy protection research in computer science can provide only technical solutions to the problem. Successful application of privacy-preserving technology will rely on the cooperation of policy makers in governments and decision makers in companies and organizations. Unfortunately, while the deployment of privacy-threatening technology, such as RFID and social networks, grows quickly, the implementation of privacy-preserving technology in real-life applications is *very limited*. As the gap becomes larger, we foresee that the number of incidents and the scope of privacy breach will increase in the near future. Below, we identify a few potential research directions in privacy preservation, together with some desirable properties that could facilitate the general public, decision makers, and systems engineers to adopt privacy-preserving technology.

*Privacy-preserving tools for individuals*. Most previous privacy-preserving techniques were proposed for data publishers, but individual record owners should also have the right and responsibility to protect their own private information. There is an urgent need for personalized privacy-preserving tools, such as privacy-preserving web browsers and minimal information disclosure protocols for e-commerce activities. It is important that the privacy-preserving notions and tools developed are intuitive for novice users. Xiao and Tao [2006b]'s work on "personalized privacy preservation" provides a good start, but little work has been conducted on this direction since.

*Privacy protection in emerging technologies*. Emerging technologies, like location-based services [Atzori et al. 2007; Hengartner 2007; You et al. 2007], RFID [Wang et al. 2006], bioinformatics, and mashup web applications, enhance our quality of life. These new technologies allow corporations and individuals to have access to previously unavailable information and knowledge; however, such benefits also bring up many new privacy issues. Nowadays, once a new technology has been adopted by a small community, it can become very popular in a short period of time. A typical example is

the social network application called Facebook.[2] Since its deployment in 2004, it has acquired 70 million active users. Due to the massive number of users, the harm could be extensive if the new technology is misused. One research direction is to customize existing privacy-preserving models for emerging technologies.

*Incorporating privacy protection in engineering process*. The issue of privacy protection is often considered after the deployment of a new technology. Typical examples are the deployments of mobile devices with location-based services [Abul et al. 2008; Atzori et al. 2007; Hengartner 2007; You et al. 2007], sensor networks, and social networks. The privacy issue should be considered as a primary requirement in the engineering process for developing new technology. This involves formal specification of privacy requirements and formal verification tools to prove the correctness of a privacy-preserving system.

Finally, we emphasize that privacy-preserving technology solves only one side of the problem. It is equally important to identify and overcome the nontechnical difficulties faced by decision makers when they deploy a privacy-preserving technology. Their typical concerns include the degradation of data/service quality, loss of valuable information, increased costs, and increased complexity. We believe that cross-disciplinary research is the key to remove these obstacles, and urge computer scientists in the privacy protection field to conduct cross-disciplinary research with social scientists in sociology, psychology, and public policy studies. Having a better understanding of the privacy problem from different perspectives can help realize successful applications of privacy-preserving technology.

## ACKNOWLEDGMENTS

We sincerely thank the reviewers of this manuscript for greatly improving the quality of this survey.

## REFERENCES

ABUL, O., BONCHI, F., AND NANNI, M. 2008. Never walk alone: Uncertainty for anonymity in moving objects databases. In *Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE)*. 376–385.

ADAM, N. R. AND WORTMAN, J. C. 1989. Security control methods for statistical databases. *ACM Comput. Surv. 21,* 4, 515–556.

AGGARWAL, C. C. AND YU, P. S. 2008a. A framework for condensation-based anonymization of string data. *Data Min. Knowl. Discov. 13*, 3, 251–275.

AGGARWAL, C. C. AND YU, P. S. 2008b. On static and dynamic methods for condensation-based privacy-preserving data mining. *ACM Trans. Datab. Syst. 33*, 1.

AGGARWAL, C. C. AND YU, P. S. 2008c. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, Berlin.

AGGARWAL, C. C. AND YU, P. S. 2007. On privacy-preservation of text and sparse binary data with sketches. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*.

AGGARWAL, C. C., PEI, J., AND ZHANG, B. 2006. On privacy preservation against adversarial data mining. In *Proceedings of the 12th ACM SIGKDD*. ACM, New York.

AGGARWAL, C. C. 2005. On *k*-anonymity and the curse of dimensionality. In *Proceedings of the 31st Conference on Very Large Data Bases (VLDB)*. 901–909.

AGGARWAL, G., FEDER, T., KENTHAPADI, K., MOTWANI, R., PANIGRAHY, R., THOMAS, D., AND ZHU, A. 2006. Achieving anonymity via clustering. In *Proceedings of the 25th ACM SIGMOD-SIGACT-SIGART PODS Conference*. ACM, New York.

AGGARWAL, G., FEDER, T., KENTHAPADI, K., MOTWANI, R., PANIGRAHY, R., THOMAS, D., AND ZHU, A. 2005. Anonymizing tables. In *Proceedings of the 10th International Conference on Database Theory (ICDT)*. 246–258.

[2]http://www.facebook.com

AGRAWAL, D. AND AGGARWAL, C. C. 2001. On the design and quantification of privacy preserving data-mining algorithms. In *Proceedings of the 20th ACM Symposium on Principles of Database Systems (PODS)*. ACM, New York, 247–255.

AGRAWAL, R. AND SRIKANT, R. 2000. Privacy preserving data mining. In *Proceedings of the ACM SIGMOD*. ACM, New York, 439–450.

AGRAWAL, S. AND HARITSA, J. R. 2005. A framework for high-accuracy privacy-preserving mining. In *Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE)*. 193–204.

ALON, N., MATIAS, Y., AND SZEGEDY, M. 1999. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci. 58*, 1, 137–147.

ATZORI, M., BONCHI, F., GIANNOTTI, F., AND PEDRESCHI, D. 2008. Anonymity preserving pattern discovery. *Int. J. Very Large Data Bases 17*, 4, 703–727.

ATZORI, M., BONCHI, F., GIANNOTTI, F., PEDRESCHI, D., AND ABUL, O. 2007. Privacy-aware knowledge discovery from location data. In *Proceedings of the International Workshop on Privacy-Aware Location-based Mobile Services (PALMS)*. 283–287.

BARAK, B., CHAUDHURI, K., DWORK, C., KALE, S., MCSHERRY, F., AND TALWAR, K. 2007. Privacy, accuracy, and consistency too: A holistic solution to contingency table release. In *Proceedings of the 26th ACM Symposium on Principles of Database Systems (PODS)*. ACM, New York, 273–282.

BARBARO, M. AND ZELLER, T. 2006. A face is exposed for AOL searcher no. 4417749. *New York Times* (Aug. 9).

BAYARDO, R. J. AND AGRAWAL, R. 2005. Data privacy through optimal k-anonymization. In *Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE)*. 217–228.

BEINAT, E. 2001. Privacy and location-based: Stating the policies clearly. *GeoInformatics*.

BLUM, A., LIGETT, K., AND ROTH, A. 2008. A learning theory approach to non-interactive database privacy. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC)*. ACM, New York, 609–618.

BLUM, A., DWORK, C., MCSHERRY, F., AND NISSIM, K. 2005. Practical privacy: The sulq framework. In *Proceedings of the 24th ACM Symposium on Principles of Database Systems (PODS)*. ACM, New York, 128–138.

BRAND, R. 2002. Microdata protection through noise addition. In *Inference Control in Statistical Databases, From Theory to Practice*, London, 97–116.

BU, Y., FU, A. W. C., WONG, R. C. W., CHEN, L., AND LI, J. 2008. Privacy preserving serial data publishing by role composition. *Proc. VLDB Endowment 1*, 1, 845–856.

BURNETT, L., BARLOW-STEWART, K., PROS, A., AND AIZENBERG, H. 2003. The gene trustee: A universal identification system that ensures privacy and confidentiality for human genetic databases. *J. Law and Medicine 10*, 506–513.

BYUN, J.-W., SOHN, Y., BERTINO, E., AND LI, N. 2006. Secure anonymization for incremental datasets. In *Proceedings of the VLDB Workshop on Secure Data Management (SDM)*.

CARLISLE, D. M., RODRIAN, M. L., AND DIAMOND, C. L. 2007. California inpatient data reporting manual, medical information reporting for California (5th Ed), Tech. rep., Office of Statewide Health Planning and Development.

CHAKARAVARTHY, V. T., GUPTA, H., ROY, P., AND MOHANIA, M. 2008. Efficient techniques for documents sanitization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*. ACM, New York.

CHAUM, D. 1981. Untraceable electronic mail, return addresses, and digital pseudonyms. *Comm. ACM 24*, 2, 84–88.

CHAWLA, S., DWORK, C., MCSHERRY, F., SMITH, A., AND WEE, H. 2005. Toward privacy in public databases. In *Proceedings of the Theory of Cryptography Conference (TCC)*. 363–385.

CHAWLA, S., DWORK, C., MCSHERRY, F., AND TALWAR, K. 2005. On privacy-preserving histograms. In *Proceedings of the Uncertainty in Artificial Intelligence Coference (UAI)*.

CLIFTON, C., KANTARCIOGLU, M., VAIDYA, J., LIN, X., AND ZHU, M. Y. 2002. Tools for privacy preserving distributed data mining. *ACM SIGKDD Explor. Newsl. 4*, 2, 28–34.

CLIFTON, C. 2000. Using sample size to limit exposure to data mining. *J. Comput. Security 8*, 4, 281–307.

COX, L. H. 1980. Suppression methodology and statistical disclosure control. *J. Am. Statistical Assoc. 75*, 370, 377–385.

DALENIUS, T. 1986. Finding a needle in a haystack - or identifying anonymous census record. *J. Official Statistics 2*, 3, 329–336.

DALENIUS, T. 1977. Towards a methodology for statistical disclosure control. *Statistik Tidskrift 15*, 429–444.

DENNING, D. E. 1985. Commutative filters for reducing inference threats in multilevel database systems. In *Proceedings of the IEEE Symposium on Security and Privacy*.

DEUTSCH, A. AND PAPAKONSTANTINOU, Y. 2005. Privacy in database publishing. In *Proceedings of the 10th International Conference on Database Theory (ICDT)*. 230–245.

DINUR, I. AND NISSIM, K. 2003. Revealing information while preserving privacy. In *Proceedings of the 22nd ACM Symposium on Principles of Database Systems (PODS)*. 202–210.

DOMINGO-FERRER, J. 2008. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, Berlin, 53–80.

DOMINGO-FERRER, J. AND TORRA, V. 2008. A critique of *k*-anonymity and some of its enhancements. In *Proceedings of the 3rd International Conference on Availability, Reliability and Security (ARES)*. 990–993.

DOMINGO-FERRER, J. AND TORRA, V. 2002. *Theory and Practical Applications for Statistical Agencies*. North-Holland, Amsterdam, 113–134.

DOMINGO-FERRER, J. 2001. *Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies*, 91–11.

DU, W. AND ZHAN, Z. 2003. Using randomized response techniques for privacy-preserving data mining. In *Proceedings of the 9th ACM SIGKDD*. ACM, New York.

DUNCAN, G. AND FIENBERG, S. 1998. Obtaining information while preserving privacy: A Markov perturbation method for tabular data. In *Statistical Data Protection*, 351–362.

DWORK, C. 2008. Differential privacy: A survey of results. In *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation (TAMC)*. 1–19.

DWORK, C. 2007. Ask a better question, get a better answer: A new approach to private data analysis. In *Proceedings of the International Conference on Database Theory (ICDT)*. 18–27.

DWORK, C. 2006. Differential privacy. In *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP)*. 1–12.

DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. 2006. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Theory of Cryptography Conference (TCC)*. 265–284.

DWORK, C. AND NISSIM, K. 2004. Privacy-preserving data mining on vertically partitioned databases. In *Proceedings of the 24th International Cryptology Conference (CRYPTO)*. 528–544.

EMAM, K. E. 2006. Data anonymization practices in clinical research: A descriptive study. Tech. rep. Access to Information and Privacy Division of Health in Canada.

EVFIMIEVSKI, A., FAGIN, R., AND WOODRUFF, D. P. 2008. Epistemic privacy. In *Proceedings of the 27th ACM Symposium on Principles of Database Systems (PODS)*. ACM, New York, 171–180.

EVFIMIEVSKI, A., SRIKANT, R., AGRAWAL, R., AND GEHRKE, J. 2002. Privacy preserving mining of association rules. In *Proceedings of the 8th ACM SIGKDD*. ACM, New York, 217–228.

FARKAS, C. AND JAJODIA, S. 2003. The inference problem: A survey. *ACM SIGKDD Explor. Newsl. 4*, 2, 6–11.

FULLER, W. A. 1993. Masking procedures for microdata disclosure limitation. *Official Statistics 9*, 2, 383–406.

FUNG, B. C. M., CAO, M., DESAI, B. C., AND XU, H. 2009. Privacy protection for RFID data. In *Proceedings of the 24th ACM SIGAPP Symposium on Applied Computing (SAC)*. ACM, New York.

FUNG, B. C. M., WANG, K., WANG, L., AND HUNG, P. C. K. 2009. Privacy-preserving data publishing for cluster analysis. *Data Knowl. Engin. 68*, 6, 552–575.

FUNG, B. C. M., WANG, K., FU, A. W. C., AND PEI, J. 2008. Anonymity for continuous data publishing. In *Proceedings of the 11th International Conference on Extending Database Technology (EDBT)*. ACM, New York, 264–275.

FUNG, B. C. M., WANG, K., WANG, L., AND DEBBABI, M. 2008. A framework for privacy-preserving cluster analysis. In *Proceedings of the 2008 IEEE International Conference on Intelligence and Security Informatics (ISI)*. 46–51.

FUNG, B. C. M., WANG, K., AND YU, P. S. 2007. Anonymizing classification data for privacy preservation. *IEEE Trans. Knowl. Data Engin. 19*, 5, 711–725.

FUNG, B. C. M., WANG, K., AND YU, P. S. 2005. Top-down specialization for information and privacy preservation. In *Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE)*. 205–216.

GEHRKE, J. 2006. Models and methods for privacy-preserving data publishing and analysis. Tutorial at the 12th ACM SIGKDD.

GHINITA, G., TAO, Y., AND KALNIS, P. 2008. On the anonymization of sparse high-dimensional data. In *Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE)*. 715–724.

GOGUEN, J. AND MESEGUER, J. 1984. Unwinding and inference control. In *Proceedings of the IEEE Symposium on Security and Privacy*.

HEGLAND, M., MCINTOSH, I., AND TURLACH, B. A. 1999. A parallel solver for generalized additive models. *Comput. Statistics Data Anal. 31*, 4, 377–396.

HENGARTNER, U. 2007. Hiding location information from location-based services. In *Proceedings of the International Workshop on Privacy-Aware Location-based Mobile Services (PALMS)*. 268–272.

HINKE, T. 1988. Inference aggregation detection in database management systems. In *Proceedings of the IEEE Symposium on Security and Privacy*. 96–107.

HINKE, T., DEGULACH, H., AND CHANDRASEKHAR, A. 1995. A fast algorithm for detecting second paths in database inference analysis. *J. Comput. Security*.

HUANG, Z., DU, W., AND CHEN, B. 2005. Deriving private information from randomized data. In *Proceedings of the ACM SIGMOD*. ACM, New York, 37–48.

HUNDEPOOL, A. AND WILLENBORG, L. 1996. *1*- and ¿-argus: Software for statistical disclosure control. In *Proceedings of the 3rd International Seminar on Statistical Confidentiality*.

IYENGAR, V. S. 2002. Transforming data to satisfy privacy constraints. In *Proceedings of the 8th ACM SIGKDD*. ACM, New York, 279–288.

JAJODIA, S. AND MEADOWS, C. 1995. Inference problems in multilevel database management systems. In *IEEE Information Security: An Integrated Collection of Essays*. 570–584.

JAKOBSSON, M., JUELS, A., AND RIVEST, R. L. 2002. Making mix nets robust for electronic voting by randomized partial checking. In *Proceedings of the 11th USENIX Security Symposium*. 339–353.

JIANG, W. AND CLIFTON, C. 2005. Privacy-preserving distributed *k*-anonymity. In *Proceedings of the 19th Annual IFIP WG 11.3 Working Conference on Data and Applications Security*. 166–177.

JIANG, W. AND CLIFTON, C. 2006. A secure distributed framework for achieving *k*-anonymity. *Very Large Data Bases J. 15*, 4, 316–333.

KANTARCIOGLU, M. 2008. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, Berlin, 313–335.

KANTARCIOGLU, M., JIN, J., AND CLIFTON, C. 2004. When do data mining results violate privacy? In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, 599–604.

KARGUPTA, H., DATTA, S., WANG, Q., AND SIVAKUMAR, K. 2003. On the privacy preserving properties of random data perturbation techniques. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM)*. 99–106.

KENTHAPADI, K., MISHRA, N., AND NISSIM, K. 2005. Simulatable auditing. In *Proceedings of the 24th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. ACM, New York, 118–127.

KIFER, D. AND GEHRKE, J. 2006. Injecting utility into anonymized datasets. In *Proceedings of ACM SIGMOD*. ACM, New York.

KIM, J. AND WINKLER, W. 1995. Masking microdata files. In *Proceedings of the ASA Section on Survey Research Methods*. 114–119.

KOKKINAKIS, D. AND THURIN, A. 2007. Anonymization of Swedish clinical data. In *Proceedings of the 11th Conference on Artificial Intelligence in Medicine (AIME)*. 237–241.

KUMAR, R., NOVAK, J., PANG, B., AND TOMKINS, A. 2007. On anonymizing query logs via token-based hashing. In *Proceedings of the 16th World Wide Wed Conference*. 628–638.

LEFEVRE, K., DEWITT, D. J., AND RAMAKRISHNAN, R. 2006a. Mondrian multidimensional k-anonymity. In *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE)*.

LEFEVRE, K., DEWITT, D. J., AND RAMAKRISHNAN, R. 2006b. Workload-aware anonymization. In *Proceedings of the 12th ACM SIGKDD*. ACM, New York.

LEFEVRE, K., DEWITT, D. J., AND RAMAKRISHNAN, R. 2005. Incognito: Efficient full-domain k-anonymity. In *Proceedings of ACM SIGMOD*. ACM, New York, 49–60.

LI, J., TAO, Y., AND XIAO, X. 2008. Preservation of proximity privacy in publishing numerical sensitive data. In *Proceedings of the ACM Conference on Management of Data (SIGMOD)*. 437–486.

LI, N., LI, T., AND VENKATASUBRAMANIAN, S. 2007. *t*-closeness: Privacy beyond *k*-anonymity and *l*-diversity. In *Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE)*.

MACHANAVAJJHALA, A., KIFER, D., ABOWD, J. M., GEHRKE, J., AND VILHUBER, L. 2008. Privacy: Theory meets practice on the map. In *Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE)*. 277–286.

MACHANAVAJJHALA, A., KIFER, D., GEHRKE, J., AND VENKITASUBRAMANIAM, M. 2007. *l*-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data 1*, 1.

MACHANAVAJJHALA, A., GEHRKE, J., KIFER, D., AND VENKITASUBRAMANIAM, M. 2006. *l*-diversity: Privacy beyond k-anonymity. In *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE)*.

MALIN, B. AND AIROLDI, E. 2006. The effects of location access behavior on re-identification risk in a distributed environment. In *Proceedings of the 6th Workshop on Privacy Enhancing Technologies (PET)*. 413–429.

MARTIN, D., KIFER, D., MACHANAVAJJHALA, A., GEHRKE, J., AND HALPERN, J. 2007. Worst-case background knowledge in privacy-preserving data publishing. In *Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE)*.

MATLOFF, N. S. 1988. Inference control via query restriction vs. data modification: A perspective. In *Database Security: Status and Prospects*. 159–166.

MEYERSON, A. AND WILLIAMS, R. 2004. On the complexity of optimal k-anonymity. In *Proceedings of the 23rd ACM SIGMOD-SIGACT-SIGART PODS*. ACM, New York, 223–228.

MIKLAU, G. AND SUCIU, D. 2004. A formal analysis of information disclosure in data exchange. In *Proceedings of the ACM SIGMOD*. ACM, New York, 575–586.

MOHAMMED, N., FUNG, B. C. M., WANG, K., AND HUNG, P. C. K. 2009. Privacy-preserving data mashup. In *Proceedings of the 12th International Conference on Extending Database Technology (EDBT)*.

MOORE, R. A., JR. 1996. 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, D.C.

MOTWANI, R. AND XU, Y. 2007. Efficient algorithms for masking and finding quasi-identifiers. In *Proceedings of the Conference on Very Large Data Bases (VLDB)*.

NERGIZ, M. E., ATZORI, M., AND CLIFTON, C. W. 2007. Hiding the presence of individuals from shared databases. In *Proceedings of ACM SIGMOD Conference*. ACM, New York, 665–676.

NERGIZ, M. E. AND CLIFTON, C. 2007. Thoughts on *k*-anonymization. *Data Knowl. Engin. 63*, 3, 622–645.

NERGIZ, M. E., CLIFTON, C., AND NERGIZ, A. E. 2007. Multirelational *k*-anonymity. In *Proceedings of the 23rd International Conference on Data Engineering (ICDE)*. 1417–1421.

OHRN, A. AND OHNO-MACHADO, L. 1999. Using Boolean reasoning to anonymize databases. *Artif. Intell. Medicine 15*, 235–254.

OZSOYOGLU, G. AND SU, T. 1990. On inference control in semantic data models for statistical databases. *J. Comput. Syst. Sci. 40*, 3, 405–443.

PAPADIMITRIOU, S., LI, F., KOLLIOS, G., AND YU, P. S. 2007. Time series compressibility and privacy. In *Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB)*, 459–470.

PINKAS, B. 2002. Cryptographic techniques for privacy-preserving data mining. *ACM SIGKDD Explor. Newsl. 4*, 2, 12–19.

POHLIG, S. AND HELLMAN, M. 1978. An improved algorithm for computing logarithms over gf(p) and its cryptographic significance. *IEEE Trans. Inform. Theory IT-24*, 106–110.

PRESIDENT INFORMATION TECHNOLOGY ADVISORY COMMITTEE. 2004. Revolutionizing health care through information technology. Tech. rep., Executive Office of the President of the United States.

QUINLAN, J. R. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

RASTOGI, V., SUCIU, D., AND HONG, S. 2007. The boundary between privacy and utility in data publishing. In *Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB)*. 531–542.

REISS, S. P. 1984. Practical data-swapping: The first steps. *ACM Trans. Datab. Syst. 9*, 1, 20–37.

REISS, S. P., POST, M. J., AND DALENIUS, T. 1982. Non-reversible privacy transformations. In *Proceedings of the 1st ACM Symposium on Principles of Database Systems (PODS)*. 139–146.

ROSEN, B. E., GOODWIN, J. M., AND VIDAL, J. J. 1992. Process control with adaptive range coding. *Biological Cyber. 67*, 419–428.

RUBIN, D. B. Discussion statistical disclosure limitation. *J. Official Statistics 9*, 2.

SAMARATI, P. 2001. Protecting respondents' identities in microdata release. *IEEE Trans. Knowl. Data Engin. 13*, 6, 1010–1027.

SAMARATI, P. AND SWEENEY, L. 1998a. Generalizing data to provide anonymity when disclosing information. In *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART (PODS)*. ACM, New York, 188.

SAMARATI, P. AND SWEENEY, L. 1998b. Protecting privacy when disclosing information: *k*-anonymity and its enforcement through generalization and suppression. Tech. rep., SRI International.

SAYGIN, Y., HAKKANI-TUR, D., AND TUR, G. 2006. *Web and Information Security*. IRM Press, 133–148.

SHANNON, C. E. 1948. A mathematical theory of communication. *The Bell Syst. Tech. J. 27*, 379 and 623.

SKOWRON, A. AND RAUSZER, C. 1992. *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Set Theory*.

SWEENEY, L. 2002a. Achieving k-anonymity privacy protection using generalization and suppression. *Int. J. Uncertainty, Fuzziness, Knowl.-Based Syst. 10*, 5, 571–588.

SWEENEY, L. 2002b. *k*-Anonymity: A model for protecting privacy. *Int. J. Uncertainty, Fuzziness, Knowl.-Based Syst. 10*, 557–570.

SWEENEY, L. 1998. Datafly: A system for providing anonymity in medical data. In *Proceedings of the IFIP TC11 WG11.3 11th International Conference on Database Securty XI: Status and Prospects*. 356–381.

TERROVITIS, M. AND MAMOULIS, N. 2008. Privacy preservation in the publication of trajectories. In *Proceedings of the 9th International Conference on Mobile Data Management (MDM)*. 65–72.

TERROVITIS, M., MAMOULIS, N., AND KALNIS, P. 2008. Privacy-preserving anonymization of set-valued data. *Proc. VLDB Endowment 1*, 1, 115–125.

THURAISINGHAM, B. M. 1987. Security checking in relational database management systems augmented with inference engines. *Comput. Security 6*, 479–492.

TRUTA, T. M. AND BINDU, V. 2006. Privacy protection: *p*-sensitive *k*-anonymity property. In *Proceedings of the Workshop on Privacy Data Management (PDM)*. 94.

VAIDYA, J. 2008. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, Berlin, 337–358.

VERYKIOS, V. S., ELMAGARMID, A. K., BERTINO, E., SAYGIN, Y., AND DASSENI, E. 2004. Association rule hiding. *IEEE Trans. Knowl. Data Engin. 16*, 4, 434–447.

VINTERBO, S. A. 2004. Privacy: A machine learning view. *IEEE Trans. Knowl. Data Engin. 16*, 8, 939–948.

WANG, K., XU, Y., FU, A. W. C., AND WONG, R. C. W. 2009. *ff*-anonymity: When quasi-identifiers are missing. In *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE)*.

WANG, K., FUNG, B. C. M., AND YU, P. S. 2007. Handicapping attacker's confidence: An alternative to k-anonymization. *Knowl. Inform. Syst. 11*, 3, 345–368.

WANG, K. AND FUNG, B. C. M. 2006. Anonymizing sequential releases. In *Proceedings of the 12th ACM SIGKDD Conference*. ACM, New York.

WANG, K., FUNG, B. C. M., AND DONG, G. 2005. Integrating private databases for data analysis. In *Proceedings of the IEEE International Conference on Intelligence and Security Informatics (ISI)*. 171–182.

WANG, K., FUNG, B. C. M., AND YU, P. S. 2005. Template-based privacy preservation in classification problems. In *Proceedings of the 5th IEEE International Conference on Data Mining (ICDM)*. 466–473.

WANG, K., YU, P. S., AND CHAKRABORTY, S. 2004. Bottom-up generalization: A data mining solution to privacy protection. In *Proceedings of the 4th IEEE International Conference on Data Mining (ICDM)*.

WANG, S.-W., CHEN, W.-H., ONG, C.-S., LIU, L., AND CHUANG, Y. 2006. RFID applications in hospitals: A case study on a demonstration RFID project in a Taiwan hospital. In *Proceedings of the 39th Hawaii International Conference on System Sciences*.

WARNER, S. L. 1965. Randomized response: A survey technique for eliminating evasive answer bias. *J. Am. Statistical Assoc. 60*, 309, 63–69.

WONG, R. C. W., FU, A. W. C., WANG, K., AND PEI, J. 2007. Minimality attack in privacy preserving data publishing. In *Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB)*. 543–554.

WONG, R. C. W., LI., J., FU, A. W. C., AND WANG, K. 2006. (*a*,*k*)-anonymity: An enhanced *k*-anonymity model for privacy preserving data publishing. In *Proceedings of the 12th ACM SIGKDD*. ACM, New York, 754–759.

WRIGHT, R. N., YANG, Z., AND ZHONG, S. 2005. Distributed data mining protocols for privacy: A review of some recent results. In *Proceedings of the Secure Mobile Ad-Hoc Networks and Sensors Workshop (MADNES)*.

XIAO, X. AND TAO, Y. 2007. m-invariance: Towards privacy preserving re-publication of dynamic datasets. In *Proceedings of the ACM SIGMOD Conference*. ACM, New York.

XIAO, X. AND TAO, Y. 2006a. Anatomy: Simple and effective privacy preservation. In *Proceedings of the 32nd Conference on Very Large Data Bases (VLDB)*.

XIAO, X. AND TAO, Y. 2006b. Personalized privacy preservation. In *Proceedings of the ACM SIGMOD Conference*. ACM, New York.

XU, J., WANG, W., PEI, J., WANG, X., SHI, B., AND FU, A. W. C. 2006. Utility-based anonymization using local recoding. In *Proceedings of the 12th ACM SIGKDD Conference*. ACM, New York.

XU, Y., FUNG, B. C. M., WANG, K., FU, A. W. C., AND PEI, J. 2008. Publishing sensitive transactions for itemset utility. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM)*.

XU, Y., WANG, K., FU, A. W. C., AND YU, P. S. 2008. Anonymizing transaction databases for publication. In *Proceedings of the 14th ACM SIGKDD Conference*. ACM, New York.

YANG, Z., ZHONG, S., AND WRIGHT, R. N. 2005. Anonymity-preserving data collection. In *Proceedings of the 11th ACM SIGKDD Conference*. ACM, New York, 334–343.

YAO, C., WANG, X. S., AND JAJODIA, S. 2005. Checking for k-anonymity violation by views. In *Proceedings of the 31st Conference on Very Large Data Bases (VLDB)*. 910–921.

YOU, T.-H., PENG, W.-C., AND LEE, W.-C. 2007. Protect moving trajectories with dummies. In *Proceedings of the International Workshop on Privacy-Aware Location-Based Mobile Services (PALMS)*. 278–282.

ZAYATZ, L. 2007. Disclosure avoidance practices and research at the U.S. Census Bureau: An update. *J. Official Statistics 23,* 2, 253–265.

ZHANG, P., TONG, Y., TANG, S., AND YANG, D. 2005. Privacy-preserving naive Bayes classifier. Lecture Notes in Computer Science, vol. 3584.

ZHANG, Q., KOUDAS, N., SRIVASTAVA, D., AND YU, T. 2007. Aggregate query answering on anonymized tables. In *Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE)*.

Received April 2008; revised December 2008; accepted December 2008