International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems

Data Privacy: Definitions and Techniques

Sabrina De Capitani di Vimercati, Sara Foresti, Giovanni Livraga, Pierangela Samarati

Dipartimento di Informatica, Università degli Studi di Milano
Via Bramante 65, 26013 Crema, Italy
firstname.lastname*@unimi.it*

The proper protection of data privacy is a complex task that requires a careful analysis of what actually has to be kept private. Several definitions of privacy have been proposed over the years, from traditional *syntactic* privacy definitions, which capture the protection degree enjoyed by data respondents with a numerical value, to more recent *semantic* privacy definitions, which take into consideration the mechanism chosen for releasing the data.

In this paper, we illustrate the evolution of the definitions of privacy, and we survey some data protection techniques devised for enforcing such definitions. We also illustrate some well-known application scenarios in which the discussed data protection techniques have been successfully used, and present some open issues.

Keywords: Privacy, Anonymity, Data Protection, Data Publication.

1. Introduction

Over the last 15 years, the advancements in the Information Technology have radically changed our lives. We can access a variety of services and information anywhere anytime using our personal computers, mobile phones, tablets, or any device with an Internet connection. Although this situation has clearly brought enormous benefits to our society, the development of the Information Technology has also had a significant impact on users' privacy. As a matter of fact, more and more personal information is collected, processed, shared, and disseminated. This includes demographic data, medical data, tweets, emails, photos, videos, as well as location information. There are a variety of reasons for collecting, sharing, and disseminating personal information. For instance, public, private, and governmental organizations might disclose or share their data collections for research or statistical purposes, for providing services more efficiently and effectively, or because forced by laws and regulations. However, disseminating and sharing personal information may put individuals' privacy at risk: How should personal information be collected and processed? How should privacy be defined and enforced?

The problem of ensuring proper protection to users' privacy is far from trivial since privacy is a multi-faced concept that may have different forms: certain (sensitive) information about users should be kept private, the identity of users should be protected, or users' actions should not be traceable. Another complicating factor

is the presence of different information sources whose analysis and correlation can lead to improper leakage of information that was not intended for disclosure. A well-known example is related to the online DVD delivery service Netflix, which in 2006 started a competition for improving its movie recommendation system based on users' previous ratings. To this purpose, Netflix released 100 million records about movie ratings by 500,000 of its subscribers. The released records were de-identified substituting subscribers' personal identifying information (e.g., name and IP address) with numerical user IDs. However, by linking the movie recommendations available on the Internet Movie Database (IMDb) with the de-identified Netflix dataset, it was possible to re-identify individuals, thus revealing potentially sensitive information (e.g., a homosexual mother sought damages in a lawsuit for being outed by Netflix released data).[1]

The research community has dedicated many efforts in developing appropriate *definitions of privacy* along with *data protection techniques* specifically targeted to efficiently enforce them. These privacy definitions (and corresponding data protection techniques) can be broadly classified in the following two main categories.

- *Syntactic* privacy definitions capture the protection degree enjoyed by data respondents with a numerical value. Data protection techniques falling in this category are aimed at satisfying a syntactic privacy requirement (e.g., each release of data must be indistinguishably related to no less than a certain number of individuals in the population).
- *Semantic* privacy definitions are based on the satisfaction of a semantic privacy requirement. Data protection techniques falling in this category are aimed at satisfying a property that must be satisfied by the mechanism chosen for releasing the data (e.g., the result of an analysis carried out on a released dataset must be insensitive to the insertion or deletion of a tuple in the dataset).

The objective of this paper is to provide an overview of the main techniques proposed in the literature to protect users' privacy in data publishing. We organize the discussion according to the above categorization of privacy definitions and techniques. In the following, for the sake of readability, we refer to data protection techniques based on a syntactic (semantic, respectively) privacy definition as *syntactic* (*semantic*, respectively) *data protection techniques*.

The remainder of this paper is organized as follows. Section 2 presents some concepts and assumptions at the basis of the syntactic and semantic privacy definitions. Section 3 illustrates the most well-known syntactic privacy definitions and data protection techniques. Section 4 discusses more recent semantic privacy definitions and data protection techniques. Section 5 presents some examples of real-world application scenarios where syntactic and semantic data protection techniques have been concretely used. Section 6 discusses open issues that still need further investigation. Finally, Section 7 gives our concluding remarks.

2. Basic Concepts

We illustrate the basic concepts on which syntactic and semantic privacy definitions and data protection techniques are based.

2.1. *Syntactic data protection techniques*

Data to be protected are typically released in the form of a table (*microdata* table) defined on a set of attributes that can be classified as follows.

- *Identifiers*: attributes that uniquely identify a respondent (e.g., `SSN`).
- *Quasi-identifiers* (QI): attributes that, in combination, can be linked with external information to re-identify (all or some of) the respondents to whom information refers, or reduce the uncertainty over their identities (e.g., `DoB`, `Sex`, and `ZIP`).
- *Confidential attributes*: attributes that represent sensitive information (e.g., `Disease`).
- *Non-confidential attributes*: attributes that are not considered sensitive by the respondents and whose release is harmless (e.g., `FavoriteColor`).

Syntactic data protection techniques are based on the assumption that the release of a microdata table can put at risk only the privacy of those individuals contributing to the data collection. The first step for protecting their privacy consists in removing (or encrypting) explicit identifiers before releasing the table. However, a de-identified microdata table does not provide any guarantee of anonymity, since the quasi-identifier can still be linked to publicly available information to re-identify respondents. A study performed on 2000 U.S. Census data showed that 63% of the U.S. population can be *uniquely identified* combining their gender, ZIP code, and complete date of birth.[2] As an example, consider the de-identified table in Figure 1(a), including the medical information of a set of hospitalized patients, and the list of teachers in Sacramento made available by the local schools in Figure 1(b). Quasi-identifying attributes `DoB`, `Sex`, and `ZIP` can be exploited for linking the tuples in the medical table with the teachers' list, possibly re-identifying individuals and revealing their illnesses. In this example, the de-identified medical data include only one male patient, born on 1958/07/09 and living in 94232 area. This combination, if unique in the external world as well, uniquely identifies the corresponding tuple as pertaining to *John Doe*, 100 Park Ave., Sacramento, revealing that he suffers from diabetes.

Syntactic approaches are commonly based on the assumption that quasi-identifiers are the only attributes that can be exploited for linking sensitive data with publicly available respondents' identities. Therefore, these approaches protect the privacy of the respondents by applying microdata protection techniques on the quasi-identifier, typically guaranteeing data truthfulness,[3] while not modifying the sensitive attributes. Syntactic data protection techniques can be classified depending on whether they are aimed at protecting data against *identity disclosure* (i.e.,

SSN	Name	DoB	Sex	ZIP	Disease
*	*	1970/09/02	M	94152	Hepatitis
*	*	1970/09/20	F	94143	Cardiomyopathy
*	*	1970/09/12	F	94148	Eczema
*	*	1970/09/05	M	94155	Pneumonia
*	*	1960/08/01	F	94154	Stroke
*	*	1960/08/02	F	94153	Stroke
*	*	1960/08/10	M	94140	Stroke
*	*	1960/08/20	M	94141	Stroke
*	*	1970/08/07	F	94141	High Cholesterol
*	*	1970/08/05	F	94142	Erythema
*	*	*1958/07/09*	*M*	*94232*	Diabetes
*	*	1970/08/25	M	94153	High Cholesterol
*	*	1970/08/30	M	94156	Angina Pectoris
*	*	1960/09/02	M	94147	Hepatitis
*	*	1960/09/05	M	94145	Flu
*	*	1960/09/10	F	94158	Angina Pectoris
*	*	1960/09/30	F	94159	Cardiomyopathy

(a) De-identified medical data

Name	Address	City	ZIP	DoB	Sex	Course	School
...	...	...	...	...	...	...	...
John Doe	100 Park Ave.	Sacramento	*94232*	*58/07/09*	*male*	Maths	High School
...	...	...	...	...	...	...	...

(b) Public list of teachers in Sacramento

Fig. 1: An example of de-identified microdata table (a) and of publicly available non de-identified dataset (b)

they protect respondents' identities) or against *attribute disclosure* (i.e., they protect respondents' sensitive information).

2.2. *Semantic data protection techniques*

Semantic data protection techniques have recently been proposed to protect the privacy of both data respondents and individuals who are not included in data undergoing public release. To illustrate, consider the release of a dataset that can be used to compute the average amount of taxes annually paid by the citizens of Sacramento for each profession, and suppose that this information was not publicly available before the release. Assume that *Alice* knows that the taxes paid by *Bob* are 1,000$ less than the average taxes paid by *teachers* living in Sacramento. Although this piece of information alone does not permit *Alice* to gain any information about the taxes paid by *Bob*, if combined with the released dataset, it allows *Alice* to infer the taxes paid by *Bob*. Note that this information leakage does not depend on whether *Bob* is represented in the released dataset or not.

Semantic techniques operate in the following two scenarios.

- *Non-interactive* scenario consists in the release of a data collection. Protection techniques are therefore used to compute a privacy-preserving dataset, which is representative of the original data collection.

- *Interactive* scenario consists in evaluating queries over a private data collection managed by the data holder, without revealing to the requesting recipient any information that is not intended for disclosure. Protection techniques are used to guarantee that the query result (also when possibly combined with other results collected by data recipients) cannot be exploited to gain information that should be kept secret.

While syntactic techniques traditionally guarantee data protection preserving the truthfulness of the released information, semantic techniques typically add noise to the released data. Noise addition perturbs the original content of the dataset, thus achieving privacy at the price of truthfulness.

3. Syntactic Approaches

We describe the k-anonymity proposal,[4] one of the most popular syntactic privacy definitions developed for protecting a released dataset against identity disclosure. We then present solutions that protect released data against attribute disclosure, and also briefly overview some enhancements to traditional syntactic techniques introduced to remove assumptions that are at the basis of the original k-anonymity proposal.

3.1. *Protecting data against identity disclosure*

k-Anonymity [4] enforces the well-known protection requirement, typically applied by statistical agencies, demanding that any released information should be *indistinguishably related* to no less than a certain number of respondents. Since re-identification is assumed to occur exploiting quasi-identifying attributes only, this general requirement has been translated into the k-anonymity requirement: *Each release of data must be such that every combination of values of quasi-identifiers can be indistinctly matched to at least k respondents*.[4] As each respondent is assumed to be represented by at most one tuple in the released table and vice-versa (i.e. each tuple includes information related to one respondent only), a microdata table satisfies the k-anonymity requirement if and only if: *i)* each tuple in the released table cannot be related to less than k individuals in the population; and *ii)* each individual in the population cannot be related to less than k tuples in the table.

To verify whether a microdata table satisfies the k-anonymity requirement, the data holder should know in advance any possible external source of information that an observer could exploit for re-identification. Since this assumption is unfeasible in practice, the k-anonymity requirement is enforced by taking a safe approach and requiring each respondent to be indistinguishable from at least $k-1$ respondents of the table itself. A table is therefore said to be k-anonymous if each combination of values of the quasi-identifier appears with either zero or at least k occurrences in the released table. For instance, the table in Figure 1(a) is 1-anonymous if we assume the quasi-identifier to be composed of `DoB`, `Sex`, and `ZIP`, since different combinations of

values appear only once in the table. The definition of k-anonymous table represents a sufficient (but not necessary) condition for the k-anonymity requirement. In fact, since each combination of values of quasi-identifying attributes appears with at least k occurrences: *i)* each respondent cannot be associated with less than k tuples in the released table; and *ii)* each tuple in the released table cannot be related to less than k respondents in the population.

k-Anonymity is typically achieved by applying *generalization* and *suppression* over quasi-identifying attributes, while leaving sensitive and non-sensitive attributes unchanged. Generalization substitutes the original values with more general values. For instance, the date of birth can be generalized by removing the day, or the day and the month of birth. Suppression consists in removing information from the microdata table. The combination of generalization and suppression has the advantage of reducing the amount of generalization required to satisfy k-anonymity, thus releasing more precise (although non-complete) information. Intuitively, if a limited number of outliers (i.e., quasi-identifying values with less than k occurrences in the table) would force a large amount of generalization to satisfy k-anonymity, these outliers can be more conveniently removed from the table, improving the quality of released data. For instance, consider the table in Figure 1(a) and assume that the quasi-identifier is composed of attribute `ZIP` only. Since there is only one person living in 94232 area (11th tuple), attribute `ZIP` should be generalized removing the last three digits to guarantee 4-anonymity. However, if the 11th tuple in the table is suppressed, 4-anonymity can be achieved by generalizing the ZIP code removing only the last digit.

The approaches proposed in the literature to enforce k-anonymity can be classified on the basis of the granularity at which generalization and suppression operate.[5] More precisely, generalization can be applied at the *cell* level (substituting the cell value with a more general value) or at the *attribute* level (generalizing all the cells in the column). Suppression can be applied at the *cell*, *attribute*, or *tuple* level (removing a single cell, a column, or a row, respectively). Most of the solutions adopt attribute generalization and tuple suppression.[4,6,7] Figure 2 reports a 4-anonymous version of the table in Figure 1(a), obtained adopting attribute-level generalization (attributes `DoB`, `Sex`, and `ZIP` have been generalized by hiding the day of birth, the sex, and the last two digits of the ZIP code, respectively) and tuple-level suppression (the 11th tuple related to *John Doe* has been removed). Note that symbol $*$ represents any value in the attribute domain. Solutions adopting cell generalization have recently been investigated, since they cause a reduced information loss with respect to attribute generalization.[8] These approaches have however the drawback of producing tables where the values in the cells of the same column may be heterogeneous (e.g., some tuples report the complete date of birth, while other tuples only report the year of birth).

Regardless of the different level at which generalization and suppression are applied to enforce k-anonymity, information loss is inevitable due to the reduction in the details of the released data. To minimize the loss of information (and maximize

SSN	Name	DoB	Sex	ZIP	Disease
		1970/09/**	*	941**	Hepatitis
		1970/09/**	*	941**	Cardiomyopathy
		1970/09/**	*	941**	Eczema
		1970/09/**	*	941**	Pneumonia
		1960/08/**	*	941**	Stroke
		1960/08/**	*	941**	Stroke
		1960/08/**	*	941**	Stroke
		1960/08/**	*	941**	Stroke
		1970/08/**	*	941**	High Cholesterol
		1970/08/**	*	941**	Erythema
		1970/08/**	*	941**	High Cholesterol
		1970/08/**	*	941**	Angina Pectoris
		1960/09/**	*	941**	Hepatitis
		1960/09/**	*	941**	Flu
		1960/09/**	*	941**	Angina Pectoris
		1960/09/**	*	941**	Cardiomyopathy

Fig. 2: An example of 4-anonymous table

the utility of released data for final recipients), it is necessary to compute a k-anonymous table that minimizes generalization and suppression. The computation of an optimal k-anonymous table is however NP-hard. Therefore, both exact and heuristic algorithms have been proposed.[5]

3.2. *Protecting data against attribute disclosure*

k-Anonymity represents an effective solution for protecting respondents' identities in microdata release. However, protection against identity disclosure does not imply protection against attribute disclosure. As a consequence, a k-anonymous table could be exploited to infer (or reduce uncertainty on) the sensitive attribute values associated with respondents. The original definition of k-anonymity has been therefore extended to prevent attribute disclosure in k-anonymous tables. ℓ-Diversity and t-closeness are two well-known extensions that we describe in the following.

ℓ-Diversity. Two attacks that may lead to attribute disclosure in a k-anonymous table are the *homogeneity attack* [4,9] and the *external knowledge attack*.[9]

- *Homogeneity attack*. The homogeneity attack occurs when, in a k-anonymous table, all the tuples in an equivalence class (i.e., all the tuples with the same value for the quasi-identifier) assume also the same value for the sensitive attribute. If a data recipient knows the quasi-identifier value of an individual represented in the microdata table, she can identify the equivalence class representing the target respondent, and then infer the value of her sensitive attribute. For instance, consider the 4-anonymous table in Figure 2 and suppose that *Alice* knows that her friend *Gary* is a male, born on 1960/08/10 and living in 94140 area. Since all the tuples in the equivalence class with quasi-identifier ⟨1960/08/**,*,941**⟩ have *Stroke* as

SSN	Name	DoB	Sex	ZIP	Disease
		1970/**/**	M	9415*	High Cholesterol
		1970/**/**	M	9415*	Angina Pectoris
		1970/**/**	M	9415*	Hepatitis
		1970/**/**	M	9415*	Pneumonia
		1970/**/**	F	9414*	Cardiomyopathy
		1970/**/**	F	9414*	Eczema
		1970/**/**	F	9414*	High Cholesterol
		1970/**/**	F	9414*	Erythema
		1960/**/**	F	9415*	Stroke
		1960/**/**	F	9415*	Stroke
		1960/**/**	F	9415*	Angina Pectoris
		1960/**/**	F	9415*	Cardiomyopathy
		1960/**/**	M	9414*	Stroke
		1960/**/**	M	9414*	Stroke
		1960/**/**	M	9414*	Hepatitis
		1960/**/**	M	9414*	Flu

Fig. 3: An example of 4-anonymous and 3-diverse table

a value for attribute `Disease`, *Alice* can infer that *Gary* had a stroke.

- *External knowledge attack*. The external knowledge attack occurs when the data recipient can reduce her uncertainty about the value of the sensitive attribute of a target respondent, exploiting some additional (external) knowledge about the respondent. As an example, consider the 4-anonymous table in Figure 2 and suppose that *Alice* knows that her friend *Ilary* is a female, living in 94141 area and born on 1970/08/07. Observing the 4-anonymous table, *Alice* can infer that *Ilary* suffers from either *High Cholesterol*, *Erythema*, or *Angina Pectoris*. Suppose now that *Alice* sees *Ilary* running in the park every day. Since a person suffering from *Angina Pectoris* does not run every day, *Alice* can infer that *Ilary* suffers from *High Cholesterol* or *Erythema*.

The definition of ℓ-diversity counteracts homogeneity and external knowledge attacks by requiring the presence of at least ℓ *well-represented* values for the sensitive attribute in each equivalence class.[9] Several definitions for "well-represented" values have been proposed. A straightforward approach is to consider ℓ values well-represented if they are different. Therefore, the simplest formulation of ℓ-diversity requires that each equivalence class be associated with at least ℓ different values for the sensitive attribute. For instance, consider the 4-anonymous and 3-diverse table in Figure 3 and suppose that *Alice* knows that her neighbor *Ilary* is a female, living in 94141 area and born on 1970/08/07. Observing the table in Figure 3, *Alice* can infer that *Ilary* suffers from either *Cardiomyopathy*, *Eczema*, *High Cholesterol*, or *Erythema*. Since *Alice* knows that *Ilary* goes running every day, *Alice* can exclude the fact that *Ilary* suffers from *Cardiomyopathy*, but she cannot precisely determine whether *Ilary* suffers from *Eczema*, *High Cholesterol*, or *Erythema*.

The problem of computing an ℓ-diverse table minimizing the loss of information

caused by generalization and suppression is computationally hard. It is interesting to note that any algorithm proposed to compute a k-anonymous table that minimizes loss of information can be adapted to guarantee also ℓ-diversity, controlling if the condition on the diversity of the sensitive attribute values is satisfied by all the equivalence classes.[9]

***t*-Closeness.** Although ℓ-diversity represents a first step in counteracting attribute disclosure, this solution may still produce a table that is vulnerable to privacy breaches caused by *skewness* and *similarity attacks*.[10]

- *Skewness attack*. The skewness attack exploits the possible difference in the frequency distribution of the sensitive attribute values within an equivalence class, with respect to the frequency distribution of sensitive attribute values in the population (or in the released microdata table). In fact, differences in these distributions highlight changes in the probability with which a respondent in the equivalence class is associated with a specific sensitive value. As an example, consider the 3-diverse table in Figure 3 and suppose that *Alice* knows that her friend *Gary* is a male living in 94140 area and born on 1960/08/10. In the equivalence class with quasi-identifier ⟨1960/**/**,M,9414*⟩, two out of four tuples have value *Stroke* for attribute `Disease`. *Alice* can infer that *Gary* had a stroke with probability 50%, compared to a probability of 12.5% of the respondents of the released table.
- *Similarity attack*. The similarity attack occurs when, in an ℓ-diverse table, the values for the sensitive attribute associated with the tuples in an equivalence class are semantically similar, although syntactically different. For instance, consider the 3-diverse table in Figure 3 and suppose that *Alice* knows that her friend *Olivia* is a female, living in 94158 area, and born on 1960/09/10. In the equivalence class with quasi-identifier ⟨1960/**/**,F,9415*⟩, attribute `Disease` assumes values *Stroke*, *Angina Pectoris*, and *Cardiomyopathy*. As a consequence, *Alice* can discover that *Olivia* suffers from a cardiovascular disease.

The definition of t-closeness has been proposed to counteract skewness and similarity attacks,[10] and requires that the frequency distribution of the sensitive values in each equivalence class be close (i.e., with distance smaller than a fixed threshold t) to that in the released microdata table. In this way, the skewness attack has no effect since the knowledge of the quasi-identifier value for a target respondent does not change the probability for a malicious recipient of correctly guessing the sensitive value associated with the respondent. t-Closeness reduces also the effectiveness of the similarity attack, because the presence of semantically similar values in an equivalence class can only be due to the presence, with similar relative frequencies, of the same values in the microdata table.

The enforcement of t-closeness requires to evaluate the distance between the

Assumption	Available techniques
multiple tuples per respondent	(X,Y)-Privacy [12] k^m-anonymity [11]
multiple tables	(X,Y)-Privacy [12] MultiR k-anonymity [13]
microdata re-publication	m-Invariance [14]
data streams	correlation tracking [15] stream k-anonymity [16] ℓ-eligibility [17]
personalized privacy preferences	(α_i,β_i)-Closeness [18] Personalized Privacy [19]
multiple quasi-identifiers	Butterfly [20]
non-predefined quasi-identifiers	k^m-anonymity [11]
external knowledge	Privacy Skyline [21] ϵ-Privacy [22] (c,k)-Safety [23]

Fig. 4: Syntactic techniques removing traditional assumptions

frequency distribution of the sensitive attribute values in the released table and in each equivalence class. Such distance can be computed adopting different metrics, such as the Earth Mover Distance used by t-closeness.[10]

3.3. *Extensions of the syntactic approaches*

k-Anonymity, ℓ-diversity, and t-closeness are based on some assumptions that make them not always suitable for specific scenarios. Figure 4 summarizes some solution that extend the definitions of k-anonymity, ℓ-diversity, and t-closeness by removing (some of) the assumptions briefly discussed in the following.

Multiple tuples per respondent. k-Anonymity assumes that, in a microdata table, each respondent is represented by a single tuple. However, a single individual might be associated with more than one tuple (e.g., in a medical dataset, each respondent may be associated with a tuple for each disease she suffers from). In this case, equivalence classes may contain tuples associated with the same respondent. Therefore, data protection techniques must require that each equivalence class, regardless of the number of tuples composing it, contains data related to at least k different individuals (e.g., k^m-anonymity,[11] (X,Y)-Privacy [12]).

Release of multiple tables. k-Anonymity assumes that all data to be released are stored in a unique table. In many real-world scenarios, however, data are organized in multiple relations, characterized by (functional) dependencies among them. In this case, to properly protect respondents' privacy, data protection techniques must guarantee that recipients cannot exploit dependencies or correlations among the

released tables to infer information not intended for disclosure (e.g., MultiR k-anonymity,[13] (X,Y)-privacy [12]).

Data republication. k-Anonymity assumes that, once released, data included in a microdata table are not further modified. However, a microdata table can be subject to frequent changes due to tuple insertions, deletions, or updates. As a consequence, it may be required to periodically re-publish the data collection. A malicious data recipient might then exploit subsequent releases to possibly correlate tuples in the different versions of the table, and gain information about respondents. For instance, assume that a value for the quasi-identifier appears in one of the releases only. This value will probably refer to a respondent that has been meanwhile removed. Data protection techniques then need to guarantee that the correlation of subsequent releases does not permit a malicious recipient to precisely re-identify data respondents (e.g., m-Invariance [14]).

Continuous data release. Traditional syntactic data protection techniques assume that all the data that need to be released are available to the data holder before their release. This may not be true in real-life scenarios where data are continuously generated and need to be timely released, since data utility decreases as time passes (e.g., credit card transactions). In this scenario, data protection techniques permit the release of a tuple only if, when combined with tuples already published, it does not violate respondents' privacy.[15,16,17]

Personalized privacy preferences. Traditional syntactic data protection techniques guarantee the same degree of privacy to all the respondents represented in the released microdata table. In fact, they define a unique privacy threshold value (e.g., the value k in k-anonymity) for the whole microdata table. Privacy requirements may however depend on respondents' preferences (different respondents may have different requirements about their own privacy), or on the sensitivity of the released values (some values may be considered "more sensitive" than others). In these scenarios, the released table must satisfy multiple privacy requirements, as the adoption of a unique privacy protection threshold to the whole dataset may result in over-protecting or under-protecting respondents/sensitive values (e.g., Personalized privacy [19] and (α_i, β_i)-closeness [18]).

Multiple quasi-identifiers. Traditional approaches assume that the released table is characterized by a unique quasi-identifier. However, different data recipients may be able to access different external data sources, which might be exploited for attacks. To limit the excessive loss of information that would be caused by considering a unique quasi-identifier composed of all the attributes that may possibly be externally available to at least a data recipient, privacy preserving techniques should be adapted to take multiple quasi-identifiers into consideration (e.g., Butterfly [20]).

Non-predefined quasi-identifiers. Syntactic anonymization approaches rely on the assumption that the attributes that can be exploited by data recipients for re-identification are known in advance. However, it may happen that the information exploited for re-identification cannot be determined a-priori and may not be represented by a set of attributes. For instance, in transactional data, a subset of the items composing a transaction may represent a quasi-identifier (e.g., the case of Netflix illustrated in Section 1). As an example, k^m-anonymity [11] has been specifically designed to protect transactional data.

External knowledge. Traditional syntactic approaches have been designed to protect microdata release against adversaries that are assumed to have specific types of knowledge. For instance, k-anonymity assumes that data recipients only know, besides the released table, publicly available datasets associating the identity of respondents with their quasi-identifier. However, data recipients may posses additional information (obtained, for example, from social networking sites) that could be exploited to infer sensitive information associated with a target respondent. For instance, consider the microdata table in Figure 3 and assume that *Alice* knows that her neighbor *John* is a male born on 1970/09/05 and living in 94155 area. Assuming that she does not have additional knowledge, *Alice* can infer from the table that *John* suffers from either *High Cholesterol*, *Angina Pectoris*, *Hepatitis*, or *Pneumonia*. Suppose that *Alice* knows that: *i) Bob*, who is in the same equivalence class as *John*'s, suffers from *High Cholesterol*; *ii)* the husband of her colleague *Carol*, whose tuple is in the same equivalence class as *John*'s, suffers from *Hepatitis*, since *Carol* took some days off to assist him; and *iii) John*'s wife does not suffer from *Pneumonia*. As a consequence, *Alice* can infer (with high probability) that *John* suffers from *Angina Pectoris*. Taking external knowledge into consideration when releasing a microdata table requires the definition of an adequate modeling of the external knowledge of the recipient. This task is complicated by the fact that it is not realistic to assume data holders to have complete knowledge of all the data available to recipients. Furthermore, information is collected and publicly released every day, and the external information that could be exploited for re-identification purposes changes continuously. Examples of attempts of modeling adversarial knowledge are represented by Privacy Skyline [21], ϵ-Privacy [22], and (c,k)-Safety [23].

4. Semantic Approaches

We now discuss recently proposed data protection techniques aimed at providing semantic privacy guarantees defined by the data holder prior to data publication. We first describe the original definition of differential privacy.[24] We then present solutions relaxing this (strict) definition to provide flexibility in its enforcement, and briefly overview solutions applying it to specific data release scenarios.

4.1. *Differential privacy*

One of the first definitions of privacy states that *anything that can be learned about a respondent from the statistical database should be learnable without access to the database*.[25] Although originally stated for statistical databases, this definition is also well suited for the microdata publishing scenario. Unfortunately, only an empty dataset can guarantee absolute protection against information leakage [24] since, besides exposing the privacy of data respondents, the release of a microdata table may also compromise the privacy of individuals who are *not* represented by a tuple in the released table (see Section 2.2).

Differential privacy is a novel privacy definition aimed at guaranteeing that the release of a microdata table does not disclose sensitive information about *any* individual who may or may not be represented by a tuple in the table.[24] Differential privacy aims at releasing a dataset that allows data recipients to learn properties about the population as a whole, while protecting the privacy of single individuals. The semantic privacy guarantee provided by differential privacy is that the probability that a malicious recipient correctly infers the sensitive attribute value associated with a target respondent is not affected by the presence/absence of the corresponding tuple in the released table. Formally, given two datasets T and T' differing only for one tuple, an arbitrary randomized function $\mathcal{K}$ (typically, the release function) satisfies ϵ-*differential privacy* if and only if $P(\mathcal{K}(T) \in S) \leq \exp(\epsilon) \cdot P(\mathcal{K}(T') \in S)$, where S is a subset of the possible outputs of function $\mathcal{K}$ and ϵ is a public privacy parameter. Intuitively, the released dataset satisfies ϵ-differential privacy if the removal (insertion, respectively) of one tuple from (into, respectively) the dataset does not significantly affect the result of the evaluation of function $\mathcal{K}$. As an example, consider an insurance company that consults a medical dataset to decide whether an individual is eligible for an insurance contract. If differential privacy is satisfied, the presence or absence of the tuple representing the individual in the dataset does not significantly affect the final decision taken by the insurance company. It is important to note that the external knowledge that an adversary may possess cannot be exploited for breaching the privacy of individuals. In fact, the knowledge that the recipient gains looking at the released dataset is bounded by the multiplicative factor $\exp(\epsilon)$, for any individual either represented or not in the released microdata table. In other words, the probability of observing a result in S for the evaluation of function $\mathcal{K}$ over T is close to the probability of observing a result in S for the evaluation of function $\mathcal{K}$ over T' (i.e., the difference between $P(\mathcal{K}(T) \in S)$ and $P(\mathcal{K}(T') \in S)$ is negligible). Note that the definition of ϵ-differential privacy does not depend on the computational resources of adversaries, and therefore it protects a data release against computationally-unbounded adversaries.

The techniques proposed to enforce the ϵ-differential privacy definition traditionally *add noise* to the released data. The magnitude of the noise is computed as a function of the difference that the insertion/removal of one respondent may cause on the result of the evaluation of function $\mathcal{K}$. Differential privacy can be

	Disease			
County	Cardiovascular Diseases	Cancer	Neurological Diseases	Cutaneous Conditions
A	35	12	25	10
B	27	20	12	20
C	0	16	10	75
D	10	40	90	15
E	38	88	22	31

Fig. 5: An example of frequency matrix representing, for each disease, the number of citizens of a given county suffering from it

enforced in both the interactive and non-interactive scenarios (see Section 2.2), possibly adopting different approaches for noise addition.[24] In the *interactive scenario*, ϵ-differential privacy is ensured by adding *random noise* to the query results evaluated on the original dataset.[26] The typical distribution considered for the random noise is *Laplace distribution* $Lap(\Delta(f)/\epsilon)$ with probability density function $P(x) = exp(-|x|/b)/2b$, where $b = \Delta(f)/\epsilon$ and $\Delta(f)$ is the maximum difference between the query result evaluated over T and over T' (which, for example, is equal to 1 for count queries, since T and T' differ for at most one tuple). In the *non-interactive scenario*, the data holder typically releases a *frequency matrix*, with a dimension for each attribute and an entry in each dimension for each value in the attribute domain. The value of a cell in the matrix is obtained counting the tuples in the table that assume, for each attribute, the value represented by the entry associated with the cell. Figure 5 illustrates an example of frequency matrix for a table with values A, B, C, D, and E for attribute `County`, and values *Cardiovascular Diseases*, *Cancer*, *Neurological Diseases*, and *Cutaneous Conditions* for attribute `Disease`. Since each cell in the frequency matrix is the result of the evaluation of a count query on the original dataset, the techniques proposed to guarantee ϵ-differential privacy in the interactive scenario can also be adopted to protect the entries of the released frequency matrix (i.e., to protect the result of the count queries).

4.2. *Relaxing differential privacy*

The original definition of ϵ-differential privacy is strict and imposes very tight constraints on the data that can be released. However, there are different scenarios where an increased flexibility, to be achieved at the price of a relaxed privacy requirement, may be accepted by the data holder to provide data recipients with information of higher interest. In the following, we briefly discuss some solutions that relax the original definition of ϵ-differential privacy.

(ϵ,δ)-Differential privacy. (ϵ,δ)-Differential privacy relaxes the original definition of ϵ-differential privacy by introducing an additive factor δ in the difference between $P(\mathcal{K}(T) \in S)$ and $P(\mathcal{K}(T') \in S)$.[27] More formally, given two datasets T

and T' differing only for one tuple, an arbitrary randomized function $\mathcal{K}$ satisfies (ϵ,δ)-*differential privacy* if and only if $P(\mathcal{K}(T) \in S) \leq \exp(\epsilon) \cdot P(\mathcal{K}(T') \in S) + \delta$, where S is a subset of the possible outputs of function $\mathcal{K}$, ϵ is a public privacy parameter, and δ is a negligible function in the size of the dataset (i.e., δ grows more slowly than the inverse of any polynomial in the size of the released table). On one hand, δ increases the threshold of the difference between the results computed over T and T', thus possibly causing a higher privacy risk. On the other hand, δ reduces noise addition and therefore permits to provide better accuracy in data release.

Computational differential privacy. The original definition of ϵ-differential privacy provides privacy guarantees against computationally unbounded adversaries. However, this worst case assumption does not hold in real-life scenarios, where adversaries have limited computational resources. The definition of ϵ-differential privacy has then be relaxed to consider *realistic* adversaries (i.e., with polynomial time computational bounds).[28] This relaxed condition permits to achieve weaker privacy guarantees, with the advantage of limiting noise addition. The solutions that consider adversaries with polynomial computational bounds can be classified in the following two categories.

- *Indistinguishability-based* approach. Given two datasets T and T' differing only for one tuple, an arbitrary randomized function $\mathcal{K}$ satisfies differential privacy if a realistic adversary is not able to distinguish (with non negligible probability) the result of the evaluation of $\mathcal{K}$ over T from the result of the evaluation of $\mathcal{K}$ over T'.
- *Simulation-based* approach. This approach first simulates the view that an adversary could gain by accessing a dataset through an arbitrary randomized function $\mathcal{K}'$ that satisfies differential privacy. If the result computed by the real releasing function $\mathcal{K}$ is computationally indistinguishable from the result of $\mathcal{K}'$, then $\mathcal{K}$ satisfies (computational) differential privacy. Indeed, a computationally bounded adversary would not be able to distinguish the result computed by $\mathcal{K}$ from the one computed by $\mathcal{K}'$.

Both these definitions of (computational) differential privacy can also be adopted to satisfy the relaxed requirement of (ϵ,δ)-differential privacy.[28]

4.3. *Differential privacy for specific problems*

The definition of differential privacy (and its relaxed formulations) can be adopted in any data release scenario, independently from the function characterizing data release. Figure 6 summarizes some of the recent refinements of differential privacy, which have been proposed for managing the release of: the result of count queries, synthetic data, and sparse frequency matrices. In the figure, the considered refinements have been classified according to the scenario in which they operate (i.e., interactive, non-interactive, or both), and the goal they achieve in data release.

Solution	Objective	Scenario: interactive	Scenario: non interactive
matrix mechanism [29]	minimize noise addition, consistent query answers	×	
Privlet [30]	reduce error in the result of range-count queries	×	×
universal histogram [31]	satisfy consistency constraints in different query results	×	
diff. private synthetic data [32]	preserve statistical characteristics of synthetic datasets		×
data summaries [33]	reduce time in computing frequency matrices		×

Fig. 6: Objective and scenarios of the semantic solutions described in Section 4.3

Count queries. Count queries are functions often used for analyzing data, and can either be directly evaluated by data recipients on the published dataset or by the data holder on her private data collection. These queries may possibly include conditions restricting the subset of tuples of interest (e.g., "determine the number of U.S. male patients suffering from hypertension hospitalized in Sacramento"). The original technique proposed to achieve differential privacy might fail to provide useful results for count queries. In fact, differential privacy guarantees sufficient accuracy in the evaluation of queries that involve a limited number of respondents. Also, the addition of random noise drawn from a Laplace distribution does not take into account correlated queries, that is, queries operating on overlapping subsets of respondents. However, the results of correlated queries should not be in conflict (e.g., two evaluations of the same query should provide the same result to avoid unintended information leakage). Recently, some specific approaches have been developed to overcome the above limitations, such as the *matrix mechanism* [29] and *Privelet* [30] to improve the quality of count query results, and *universal histograms* [31] to avoid conflicts in different query results.

Synthetic data. A traditional microdata protection technique consists in replacing the original dataset with a synthetic data collection that preserves some (key) statistical properties of the original microdata table.[3] The release of synthetic data does not put respondents' privacy at risk, since their real data are not released. Differential privacy has been recently used for computing a privacy-preserving synthetic dataset.[32]

Sparse frequency matrix. A frequency matrix is sparse when the number of non-zero entries represents a small fraction of the entries in the table. Traditional techniques proposed to guarantee ϵ-differential privacy do not make differences between zero and non-zero entries in the frequency matrix, and might generate a vast amount of *dummy* data. In fact, noise should be added to every cell, including those

with zero counts. Since noise has a low probability of being zero, the resulting differentially private matrix tends to become large and dense, and is computationally expensive to generate. The idea of releasing only a *summary* (i.e., a subset) of the original matrix has been put forward to the aim of providing differentially private results at a limited computational cost.[33]

5. Application Scenarios of Data Protection Techniques

The problem of protecting the respondents' privacy thought the application of appropriate techniques has been considered in several scenarios (e.g., [4,9,10,34,35,36]). In particular, the definitions of privacy and the corresponding protection techniques illustrated in previous sections have been adopted not only in a data publishing scenario but also in scenarios where the collected dataset might not need to be released. In the following, we illustrate how both syntactic and semantic privacy definitions have been used in scenarios of data mining, location data, and social networks.

5.1. *Privacy-preserving data mining*

Due to the growing amount of data being collected every day, data mining techniques are becoming more and more important for assisting decision making processes and extracting knowledge from huge data collections (e.g., frequent patterns, association rules, item classifications). Information extracted through data mining techniques, even if not explicitly including the original data, is built on them and can put the privacy of data respondents at risk. Data mining can therefore be adopted only with proper guarantees that the privacy of the underlying data is not compromised. *Privacy preserving data mining* has been proposed to counteract this privacy concern,[37,38,39] and its main goal is to provide a trade-off between sharing information for data mining analysis, on one side, and protecting information to preserve the privacy of data respondents on the other side.

Privacy preserving data mining techniques can be based on either syntactic or semantic privacy definitions, and can be classified in the following two categories.[37]

- *Protect-and-Mine*. These techniques first apply a privacy protection technique on the original dataset, and then perform mining on the obtained privacy-preserving dataset. These approaches rely on the observation that any computation performed on a dataset that preserves respondents' privacy satisfy the same privacy definition. The advantage of these approaches is that of decoupling data protection from mining, which can then be performed by parties different from the data holders. The disadvantage is that data mining algorithms do not operate on the original data, and then the usefulness and significance of the mining results can be compromised. Specific Protect-and-Mine strategies have been proposed to satisfy either syntactic or semantic privacy definitions. The basic idea of these solutions con-

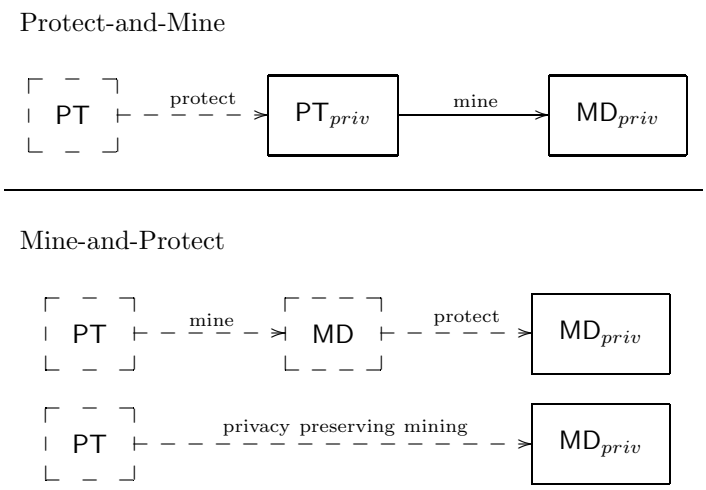

Fig. 7: Approaches for combining privacy-preserving techniques and data mining [37]

sists in applying data mining algorithms on k-anonymized tables,[40,41,42,43] or on differentially private datasets.[44,45]

- *Mine-and-Protect*. These techniques perform mining on the original dataset and apply specific techniques to guarantee that the mined results are privacy-preserving. This approach can be performed either executing the two steps in sequence, or combining them in a unique algorithm. Syntactic data protection techniques adopting the Mine-and-Protect approach protect the results of data mining guaranteeing that a malicious recipient cannot reconstruct (a portion of) the original dataset that violates the k-anonymity requirement.[46,47] Semantic approaches adopting the Mine-and-Protect strategy typically provide differential privacy within the mining process.[48,49,50,51,52] It is interesting to note that, while Protect-and-Mine strategy can only be adopted in the non-interactive scenario, Mine-and-Protect approach is also suited for the interactive scenario.

Figure 7 graphically illustrates this classification. In the figure, boxes represent data and edges represent processes producing data from data. The different data boxes are: PT, representing the original data collection; PT$_{priv}$, a privacy-preserving version of PT; MD, a result of a data mining process (without considering privacy constraints); and MD$_{priv}$, the result of a data mining process that satisfies the privacy constraints (e.g., the k-anonymity requirement) over PT. Dashed lines for boxes and edges denote data and processes, respectively, reserved to the data holder, while continuous lines denote data and processes that can be viewed and/or executed by other parties (as their visibility and execution does not violate privacy for the respondents of the original dataset).

5.2. *Protection of location data*

The diffusion of computing devices with location capabilities makes the location of users a new type of information, used by service providers to offer personalized Location-Based Services (LBSs). The knowledge of the position of users may however put their privacy at risk. Indeed, this information can allow the service provider to physically track users, and it could also be exploited for user re-identification (i.e., it can act as a quasi-identifier). For instance, suppose that *Alice* is hospitalized in a clinic specialized in treatments for cardiovascular diseases, and that while being hospitalized she uses a location-based service: the knowledge of her position can reveal that she suffers from a heart-related problem. Fostered by the growing demand for privacy protection in LBSs, in recent years the research community has addressed this problem and proposed several solutions for guaranteeing proper protection of users identities, locations, and personal information in the context of location data.

Anonymity-based solutions[53] enforce syntactic privacy requirements when location data can be exploited by a malicious data recipient as a quasi-identifier. These techniques provide respondents' privacy by enforcing (a possibly refined definition of) the k-anonymity requirement (i.e., requiring the presence of at least k different individuals in the same position).[53,54,55,56,57,58] To protect users' privacy when mining location data, recent privacy-preserving techniques guarantee that the results of the algorithm chosen for mining location data satisfy differential privacy.[59]

5.3. *Private analysis of social networks*

Social networks represent huge sources of (personal) information, as users share with each other personal information about themselves (e.g., friends, interests, and pictures). This huge amount of information is extremely valuable, as it is witnessed by the arrival of Facebook at the U.S. Nasdaq Stock Market. Indeed, the analysis of information collected by social networks can reveal (hidden) social patterns [60] that may put the privacy of the users at risk. The problem of protecting sensitive data in social networks environments is then becoming more and more important.

A social network can be conveniently modeled as a graph, where nodes represent users and edges represent their relationships (possibly of different types). The peculiarities of the graph representing the social network (e.g., the presence of a node with a certain number of incident edges) may however be exploited to re-identify users, since unique characteristics might make a user stand from others. Both syntactic [60,61,62,63] and, more recently, semantic [64,65,66] approaches have been proposed to protect the privacy of social network users.

6. Open Issues

The definition and the modeling of respondents' privacy is far from being a trivial task. The scientific community has provided several definitions of privacy, both in the syntactic and semantic scenarios, and has devoted many efforts in the design of

effective techniques to guarantee their satisfaction. Despite these efforts, protecting data privacy is still an open issue, which deserves further study and analysis.

Syntactic solutions, while conveniently capturing with a numerical value the protection degree enjoyed by data respondents, rely on restrictive assumptions that make them not easily applicable in many real-world scenarios. As already noted, these techniques assume that the external knowledge of a possible observer is known in advance to the data holder and that such a knowledge is limited and falls in predefined categories. As a consequence, syntactic solutions may fail in providing an adequate privacy level when the adversarial external knowledge does not fit one of the models proposed in the literature (see Section 3). Also, modeling any possible source of information that could be exploited by an observer is a difficult (if not impossible) task.

Semantic privacy definitions, first introduced to overcome the limitations of syntactic techniques, suffer from other limitations. Indeed, semantic solutions are aimed at providing a (semantic) privacy guarantee that prevents malicious observers from drawing a specific kind of inference from the released data collection. However, "one size does not fit all", and the inference channel blocked by a semantic definition of privacy may not be suited for all the possible data publishing scenarios. As a consequence, even a dataset satisfying differential privacy is vulnerable to privacy breaches, as briefly illustrated in the following.

- Since differential privacy does not make *any* assumption on how data have been collected and generated, individuals cannot be protected against malicious observers that are interested in determining whether an individual took part in the data generation process. As a matter of fact, the removal of a tuple from the released dataset does not hide all the traces that an observer could exploit to infer whether the individual participated in the computation of the released table. For instance, consider a social network where users *Alice* and *Carol* do not know each other, but are both friends with *Bob*, who is their unique common friend. *Bob* may introduce *Alice* to *Carol*, who then become friends. Suppose now that *Bob* unsubscribes from the social network before the list of users is publicly released: even though *Bob* is not included in the list, the friendship between *Alice* and *Carol* leaves a trace testifying the fact that he participated in the social network.
- Differential privacy does not take into consideration precise query answers on the original data that may have been disclosed prior to the privacy-preserving publication of data. As an example, to periodically monitor the risk of an epidemic disease, a hospital might need to release the exact number of patients suffering from a rare disease, and this exact information might be known to observers. Such deterministic statistics over a data collection can possibly be exploited by a recipient to infer sensitive information, since they can reduce (or even nullify) the noise added by the data

holder to limit the risk of privacy breaches.

- Differential privacy techniques assume independence among the records of the data collection to be released. Dependencies among tuples can therefore put individuals' privacy at risk. Even if an attacker neither knows nor can infer that a respondent is represented in the released dataset, the knowledge that she participated in the data generation process might leak sensitive information about other individuals that are somewhat related to her (e.g., her relatives). As an example, suppose that a data recipient can infer that *Bob*'s data has been used for the computation of the privacy-preserving table representing individuals suffering from flu. Even if the observer does not know whether *Bob*'s tuple belongs to the released table, she can easily infer that (with high probability) her wife also suffers from flu.

Taking into consideration these issues, Kifer and Machanavajjhala propose a novel definition of privacy, based on the concept of *evidence of participation*.[67] This privacy notion considers the inferences that can be drawn from a differentially private dataset exploiting the knowledge about the participation of an individual in the generation of the dataset.[67] Starting from this novel definition of privacy, and acknowledging that there exists many different inference channels that can put respondents' privacy at risk, a novel privacy framework has been proposed.[68] Goal of this framework is to model inference risks in advance to provide a customized definition of privacy, tailored for meeting the privacy needs of the data holder in counteracting specific inference channels.

7. Conclusions

Privacy is a multi-faceted concept that has been the subject of several definitions and refinements. Different data protection techniques have been proposed to meet these privacy definitions, to ensure that no individuals' identities or sensitive information be improperly disclosed. In this paper, we first illustrated traditional techniques adopting a syntactic definition of privacy, designed for preventing identity and attribute disclosure in microdata publishing. We then discussed more recent proposals adopting a semantic approach. We also presented an overview of some applications of the data protection techniques discussed in the paper. Finally, we highlighted open issues.

Acknowledgments

This work was partially supported by the Italian Ministry of Research within the PRIN 2008 project "PEPPER" (2008SY2PH4), and by the Università degli Studi di Milano within the "UNIMI per il Futuro - 5 per Mille" project "PREVIOUS".

References

1. A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *Proc. of IEEE S&P 2008*, Berkeley/Oakland, CA, USA, May 2008.
2. P. Golle. Revisiting the uniqueness of simple demographics in the US population. In *Proc. of WPES 2006*, Alexandria, VA, USA, October 2006.
3. Federal Committee on Statistical Methodology. *Statistical policy working paper 22 (Second Version)*. USA, December 2005. Report on Statistical Disclosure Limitation Methodology.
4. P. Samarati. Protecting respondents' identities in microdata release. *IEEE TKDE*, 13(6):1010–1027, November/December 2001.
5. V. Ciriani, S. De Capitani di Vimercati, S. Foresti, and P. Samarati. k-Anonymity. In T. Yu and S. Jajodia, editors, *Secure Data Management in Decentralized Systems*. Springer-Verlag, 2007.
6. R. J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *Proc. of ICDE 2005*, Tokyo, Japan, April 2005.
7. K. LeFevre, D.J. DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *Proc. of SIGMOD 2005*, Baltimore, MD, USA, June 2005.
8. K. LeFevre, D.J. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. In *Proc. of ICDE 2006*, Atlanta, GA, USA, April 2006.
9. A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. ℓ-diversity: Privacy beyond k-anonymity. *ACM TKDD*, 1(1):3:1–3:52, March 2007.
10. N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and ℓ-diversity. In *Proc. of ICDE 2007*, Istanbul, Turkey, 2007.
11. M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving anonymization of set-valued data. *PVLDB*, 1(1):115–125, August 2008.
12. K. Wang and B.C.M. Fung. Anonymizing sequential releases. In *Proc. of KDD 2006*, Philadelphia, PA, USA, August 2006.
13. M.E. Nergiz, C. Clifton, and A.E. Nergiz. Multirelational k-anonymity. In *Proc. of ICDE 2007*, Istanbul, Turkey, April 2007.
14. X. Xiao and Y. Tao. m-invariance: Towards privacy preserving re-publication of dynamic datasets. In *Proc. of SIGMOD 2007*, Beijing, China, June 2007.
15. F. Li, J. Sun, S. Papadimitriou, G.A. Mihaila, and I. Stanoi. Hiding in the crowd: Privacy preservation on evolving streams through correlation tracking. In *Proc. of ICDE 2007*, Istanbul, Turkey, April 2007.
16. B. Zhou, Y. Han, J. Pei, B. Jiang, Y. Tao, and Y. Jia. Continuous privacy preserving publishing of data streams. In *Proc. of the EDBT 2009*, Saint Petersburg, Russia, March 2009.
17. K. Wang, Y. Xu, R. Wong, and A. Fu. Anonymizing temporal data. In *Proc. of ICDM 2010*, Sydney, Australia, December 2010.
18. K. B. Frikken and Y. Zhang. Yet another privacy metric for publishing micro-data. In *Proc. of WPES 2008*, Alexandria, VA, USA, October 2008.
19. X. Xiao and Y. Tao. Personalized privacy preservation. In *Proc. of SIGMOD 2006*, Chicago, IL, USA, June 2006.
20. J. Pei, Y. Tao, J. Li, and X. Xiao. Privacy preserving publishing on multiple quasi-identifiers. In *Proc. of ICDE 2009*, Shanghai, China, March - April 2009.
21. B.-C. Chen, K. LeFevre, and R. Ramakrishnan. Privacy skyline: Privacy with multidimensional adversarial knowledge. In *Proc. of the VLDB 2007*, Vienna, Austria, 2007.
22. A. Machanavajjhala, J. Gehrke, and M. Götz. Data publishing against realistic ad-

versaries. *PVLDB*, 2(1):790–801, August 2009.
23. D. J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J.Y. Halpern. Worst-case background knowledge for privacy-preserving data publishing. In *Proc. of ICDE 2007*, Istanbul, Turkey, April 2007.
24. C. Dwork. Differential privacy. In *Proc. of ICALP 2006*, Venice, Italy, July 2006.
25. T. Dalenius. Towards a methodology for statistical disclosure control. *Statistik Tidskrift*, 15(429-444), 1977.
26. C. Dwork, F. Mcsherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Proc. of TCC 2006*, New York, NY, USA, March 2006.
27. C. Dwork and A. Smith. Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, 1(2):135–154, 2009.
28. I. Mironov, O. Pandey, O. Reingold, and S.P. Vadhan. Computational differential privacy. In *Proc. of CRYPTO 2009*, Santa Barbara, CA, USA, August 2009.
29. C. Li, M. Hay, V. Rastogi, G. Miklau, and A. McGregor. Optimizing linear counting queries under differential privacy. In *Proc. of PODS 2010*, Indianapolis, IN, USA, June 2010.
30. X. Xiao, G. Wang, and J. Gehrke. Differential privacy via wavelet transforms. *IEEE TKDE*, 23(8):1200–1214, August 2011.
31. M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially private histograms through consistency. *PVLDB*, 3(1-2):1021–1032, September 2010.
32. S. Zhou, K. Ligett, and L. Wasserman. Differential privacy with compression. In *Proc. of ISIT 2009*, Coex, Seoul, Korea, June-July 2009.
33. G. Cormode, M. Procopiuc, D. Srivastava, and T. Tran. Differentially private publication of sparse data. In *Proc. of EDBT/ICDT 2012*, Berlin, Germany, March 2012.
34. F. Dankar and K. El Emam. The application of differential privacy to health data. In *Proc. of PAIS 2012*, Berlin, Germany, March 2012.
35. G. Mathew and Z. Obradovic. A privacy-preserving framework for distributed clinical decision support. In *Proc. of ICCABS 2011*, Orlando, FL, USA, February 2011.
36. S. Cimato, M. Gamassi, V. Piuri, and F. Scotti. Privacy-aware biometrics: Design and implementation of a multimodal verification system. In *Proc. of ACSAC 2008*, Anaheim, CA, USA, December 2008.
37. V. Ciriani, S. De Capitani di Vimercati, S. Foresti, and P. Samarati. k-Anonymous data mining: A survey. In C.C. Aggarwal and P.S. Yu, editors, *Privacy-Preserving Data Mining: Models and Algorithms*. Springer-Verlag, 2008.
38. J. Domingo-Ferrer and V. Torra. Privacy in data mining. *Data Mining and Knowledge Discovery*, 11(2):117–119, September 2005.
39. V. Torra. Privacy in data mining. In *Data Mining and Knowledge Discovery Handbook*, pages 687–716. Springer-Verlag, 2010.
40. B.C.M. Fung, K. Wang, and P.S. Yu. Anonymizing classification data for privacy preservation. *IEEE TKDE*, 19(5):711–725, May 2007.
41. N. Matatov, L. Rokach, and O. Maimon. Privacy-preserving data mining: A feature set partitioning approach. *Information Sciences*, 180(14):2696 – 2720, July 2010.
42. G. Navarro-Arribas, V. Torra, A. Erola, and J. Castellà-Roca. User k-anonymity for privacy preserving data mining of query logs. *Information Processing & Management*, 48(3):476–487, 2012.
43. K. Wang, P.S. Yu, and S. Chakraborty. Bottom-up generalization: A data mining solution to privacy protection. In *Proc. of ICDM 2004*, Brighton, UK, November 2004.
44. R. Chen, N. Mohammed, B.C.M. Fung, B.C. Desai, and L. Xiong. Publishing set-valued data via differential privacy. *PVLDB*, 4(11):1087–1098, September 2011.

45. N. Mohammed, R. Chen, B.C.M. Fung, and P.S. Yu. Differentially private data release for data mining. In *Proc. of KDD 2011*, San Diego, CA, USA, August 2011.
46. M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi. Anonymity preserving pattern discovery. *The VLDB Journal*, 17(4):703–727, July 2008.
47. A. Friedman, R. Wolff, and A. Schuster. Providing k-anonymity in data mining. *The VLDB Journal*, 17(4):789–804, July 2008.
48. B. Ding, M. Winslett, J. Han, and Z. Li. Differentially private data cubes: Optimizing noise sources and consistency. In *Proc. of SIGMOD 2011*, Athens, Greece, June 2011.
49. A. Friedman and A. Schuster. Data mining with differential privacy. In *Proc. of KDD 2010*, Washington, DC, USA, July 2010.
50. G. Jagannathan, K. Pillaipakkamnatt, and R.N. Wright. A practical differentially private random decision tree classifier. In *Proc. of ICDMW 2009*, Miami, FL, USA, December 2009.
51. T. Wang and L. Liu. Output privacy in data mining. *ACM TODS*, 36(1):1–34, March 2011.
52. N. Zhang, M. Li, and W. Lou. Distributed data mining with differential privacy. In *Proc. of ICC 2011*, Kyoto, Japan, June 2011.
53. C. Bettini, S. Mascetti, X. S. Wang, D. Freni, and S. Jajodia. Anonymity and historical-anonymity in location-based services. In C. Bettini, S. Jajodia, P. Samarati, and X. S. Wang, editors, *Privacy in Location-Based Applications*. Springer-Verlag, 2009.
54. J. Domingo-Ferrer and R. Trujillo-Rasua. Microaggregation- and permutation-based anonymization of movement data. *Information Sciences*, 208:55–80, November 2012.
55. B. Gedik and L. Liu. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE TMC*, 7(1):1–18, January 2008.
56. S. Mascetti, C. Bettini, D. Freni, and X. S. Wang. Spatial generalisation algorithms for LBS privacy preservation. *Journal of Location Based Services*, 1(3):179–207, September 2007.
57. M.F. Mokbel, C.-Y. Chow, and W.G. Aref. The new Casper: Query processing for location services without compromising privacy. In *Proc. of VLDB 2006*, Seoul, Korea, September 2006.
58. K. Mouratidis and M.L. Yiu. Anonymous query processing in road networks. *IEEE TKDE*, 22(1):2–15, January 2010.
59. S.-S. Ho and S. Ruan. Differential privacy for location pattern mining. In *Proc. of SPRINGL 2011*, Chicago, IL, USA, November 2011.
60. B. Zhou, J. Pei, and W. Luk. A brief survey on anonymization techniques for privacy preserving publishing of social network data. *SIGKDD Explorations Newsletter*, 10(2):12–22, December 2008.
61. M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. *PVLDB*, 1(1):102–114, August 2008.
62. E. Zheleva and L. Getoor. Preserving the privacy of sensitive relationships in graph data. In *Proc. of PinKDD 2007*, San Jose, CA, USA, July 2008.
63. B. Zhou and J. Pei. The k-anonymity and ℓ-diversity approaches for privacy preservation in social networks against neighborhood attacks. *Knowledge and Information Systems*, 28(1):47–77, 2011.
64. M. Hay, C. Li, G. Miklau, and D. Jensen. Accurate estimation of the degree distribution of private networks. In *Proc. of ICDM 2009*, Miami, FL, USA, December 2009.
65. V. Rastogi, M. Hay, G. Miklau, and D. Suciu. Relationship privacy: Output perturbation for queries with joins. In *Proc. of PODS 2009*, Providence, RI, USA, June-July

2009.
66. D. J. Mir and R. N. Wright. A differentially private graph estimator. In *Proc. of ICDMW 2009*, Miami, FL, USA, December 2009.
67. D. Kifer and A. Machanavajjhala. No free lunch in data privacy. In *Proc. of SIGMOD 2011*, Athens, Greece, June 2011.
68. D. Kifer and A. Machanavajjhala. A rigorous and customizable framework for privacy. In *Proc. of PODS 2012*, Scottsdale, AZ, USA, May 2012.