Publishing data from electronic health records while preserving privacy: A survey of algorithms

Aris Gkoulalas-Divanis[a], Grigorios Loukides[b], Jimeng Sun[c]

[a]*IBM Research-Ireland, Damastown Industrial Estate, Mulhuddart, Dublin 15, Ireland.*
[b]*School of Computer Science & Informatics, Cardiff University, 5 The Parade, Roath, Cardiff, CF24 3AA, UK.*
[c]*IBM Thomas J. Watson Research Center, 17 Skyline Drive, Hawthorne, NY 10532, USA.*

Abstract

The dissemination of Electronic Health Records (EHRs) can be highly beneficial for a range of medical studies, spanning from clinical trials to epidemic control studies, but it must be performed in a way that preserves patients' privacy. This is not straightforward, because the disseminated data need to be protected against several privacy threats, while remaining useful for subsequent analysis tasks. In this work, we present a systematic review of algorithms that have been proposed for publishing structured patient data, in a privacy-preserving way. We review more than 45 popular algorithms, derive insights on their operation, and highlight their advantages and disadvantages. We also provide a discussion of some promising directions for future research in this area.

Keywords: privacy, electronic health records, anonymization, algorithms, survey

1. Introduction

Electronic Medical Record / Electronic Health Record (EMR/EHR) systems are increasingly adopted to collect and store various types of patient data, which contain information about patients' demographics, diagnosis codes, medication,

allergies, and laboratory test results [22, 90, 63]. For instance, the use of EMR/EHR systems, among office-based physicians, increased from 18% in 2001 to 72% in 2012 and is estimated to exceed 90% by the end of the decade [56].

Data from EMR/EHR systems are increasingly disseminated, for purposes beyond primary care, and this has been shown to be a promising avenue for improving research [63]. This is because it allows data recipients to perform large-scale, low-cost analytic tasks, which require applying statistical tests (e.g., to study correlations between BMI and diabetes), data mining tasks, such as classification (e.g., to predict domestic violence [107]) and clustering (e.g., to control epidemics [117]), or query answering. To facilitate the dissemination and reuse of patient-specific data and help the advancement of research, a number of repositories have been established, such as the the Database of Genotype and Phenotype (dbGaP) [89], in the U.S., and the U.K. Biobank [104], in the United Kingdom.

1.1. Motivation

While the dissemination of patient data is greatly beneficial, it must be performed in a way that preserves patients' privacy. Many approaches have been proposed to achieve this, by employing various techniques [43, 5], such as cryptography (e.g., [73, 55, 121, 11]) and access control (e.g., [110, 71]). However, these approaches are not able to offer patient anonymity (i.e., that patients' private and confidential information will not be disclosed) when data about patients are disseminated [39]. This is because the data need to be disseminated to a wide (and potentially unknown) set of recipients.

Towards preserving anonymity, policies that restrict the sharing of patient-specific medical data are emerging worldwide [91]. For example, in the U.S., the Privacy Rule of the Health Insurance Portability and Accountability Act (HIPAA)

[120] outlines two policies for protecting anonymity, namely *Safe Harbor*, and *Expert Determination*. The first of these policies enumerates eighteen direct identifiers that must be removed from data, prior to their dissemination, while, according to the Expert Determination policy, an expert needs to certify that the data to be disseminated pose a low privacy risk before the data can be shared with external parties. Similar policies are in place in countries, such as the U.K. [2] and Canada [3], as well as in the European Union [1]. These policies focus on preventing the privacy threat of *identity disclosure* (also referred to as *re-identification*), which involves the association of an identified individual with their record in the disseminated data. However, it is important to note that they do not provide *any computational guarantees for thwarting identity disclosure nor aim at preserving the usefulness of disseminated data in analytic tasks*.

To address re-identification, as well as other privacy threats, the computer science and health informatics communities have developed various techniques. Most of these techniques aim at publishing a dataset of patient records, while satisfying certain privacy and data usefulness objectives. Typically, privacy objectives are formulated using privacy models, and enforced by algorithms that transform a given dataset (to facilitate privacy protection) to the minimum necessary extent. The majority of the proposed algorithms are applicable to data containing demographics or diagnosis codes[1], focus on preventing the threats of *identity*, *attribute*, and/or *membership* disclosure (to be defined in subsequent sections), and operate by transforming the data using *generalization* and/or *suppression* techniques.

[1]These algorithms deal with either relational or transaction (set-valued) attributes. However, following [34, 75, 76, 87], we discuss them in the context of demographic and diagnosis information, which is modeled using relational and transaction attributes, respectively.

1.2. Contributions

In this work, we present a survey of algorithms for publishing patient-specific data in a privacy-preserving way. We begin by discussing the main privacy threats that publishing such data entails, and the privacy models that have been designed to prevent these threats. Subsequently, we provide a systematic review of algorithms, for each of these threats, which explains the strategies these algorithms employ for: (i) transforming data, (ii) preserving data usefulness, and (iii) searching the space of potential solutions. Based on these strategies, we then classify over 45 popular privacy algorithms. This allows deriving interesting insights on the operation of these algorithms, as well as on their advantages and limitations. In addition, we provide an overview of techniques for preserving privacy that are designed for different settings and types of data, and identify a number of important research directions for future work.

To the best of our knowledge, this is the first survey on algorithms for facilitating the privacy-preserving sharing of structured medical data. However, there are surveys in the computer science literature that do not focus on methods applicable to such data [39], as well as surveys that focus on privacy preservation methods for text data [94], privacy policies [91, 93], or system security [36] issues. In addition, we would like to note that the aim of this paper is to provide insights on the tasks and objectives of a wide range of algorithms. Thus, we have omitted the technical details and analysis of specific algorithms and refer the reader to the publications describing the algorithms for them.

1.2.1. Organization

The remainder of this work is organized as follows. Section 2 presents the privacy threats and models that have been proposed for preventing them. Sec-

tion 3 discusses the two scenarios for privacy-preserving data sharing. Section 4 surveys algorithms for publishing data, in the non-interactive scenario. Section 5 discusses other classes of related techniques. Section 6 presents possible directions for future research, and Section 7 concludes the paper.

2. Privacy threats and models

In this section, we first discuss the major privacy threats that are related to the disclosure of individuals' private and/or sensitive information. Then, we present privacy models that can be used to guard against each of these threats. The importance of discussing privacy models is twofold. First, privacy models can be used to evaluate how safe data are prior to their release. Second, privacy models can be incorporated into algorithms to ensure that the data can be transformed in a way that preserves privacy.

2.1. Privacy threats

Privacy threats relate to three different types of attributes, *direct identifiers*, *quasi-identifiers*, and *sensitive attributes*. Direct identifiers are attributes that can explicitly re-identify individuals, such as name, mailing address, phone number, social security number, other national IDs, and email address. On the other hand, quasi-identifiers are attributes which *in combination* can lead to identity disclosure, such as demographics (e.g., gender, date of birth, and zip code) [109, 128] and diagnosis codes [75]. Last, sensitive attributes are those that patients are not willing to be associated with. Examples of these attributes are specific diagnosis codes (e.g., psychiatric diseases, HIV, cancer, etc.) and genomic information. In Table 1, we present an example dataset, in which `Name` and `Phone Number`

are direct identifiers, `Date of birth`, `Zip Code`, and `Gender` are quasi-identifiers, and `DNA` is a sensitive attribute.

Direct identifiers		**Quasi-identifiers**			**Sensitive Attribute**
Name	Phone Number	Date of birth	Zip Code	Gender	DNA
Tom Green	6152541261	11.02.1980	55432	Male	AT. . .G
Johanna Marer	6152532126	17.01.1982	55454	Female	CG. . .A
Maria Durhame	6151531562	17.01.1982	55332	Female	TG. . .C
Helen Tulid	6153553230	10.07.1977	55454	Female	AA. . .G
Tim Lee	6155837612	15.04.1984	55332	Male	GC. . .T

Table 1: An example of different types of attributes in a relational table

Based on the above-mentioned types of attributes, we can consider the following classes of privacy threats:

- *Identity disclosure (or re-identification)* [112, 128]: This is arguably the most notorious threat in publishing medical data. It occurs when an attacker can associate a patient with their record in a published dataset. For example, an attacker may re-identify `Maria` in Table 1, even if the table is published deprived of the direct identifiers (i.e., `Name` and `Phone Number`). This is because `Maria` is the only person in the table who was born on `17.01.1982` and also lives in zip code `55332`.

- *Membership disclosure* [100]: This threat occurs when an attacker can infer with high probability that an individual's record is contained in the published data. For example, consider a dataset which contains information on only HIV-positive patients.The fact that a patient's record is contained in the dataset allows inferring that the patient is HIV-positive, and thus poses a threat to privacy. Note that membership disclosure may occur even when the data are protected from identity disclosure, and that there are several

real-world scenarios where protection against membership disclosure is required. Such interesting scenarios were discussed in detail in [100, 101].

- *Attribute disclosure (or sensitive information disclosure)* [88]: This threat occurs when an individual is associated with information about their sensitive attributes. This information can be, for example, the individual's value for the sensitive attribute (e.g., the value in *DNA* in Figure 1), or a range of values which contain an individual's sensitive value (e.g., if the sensitive attribute is *Hospitalization Cost*, then knowledge that a patient's value in this attribute lies in a narrow range, say $[5400, 5500]$, may be considered as sensitive, as it provides a near accurate estimate of the actual cost incurred, which may be considered to be high, rare, etc.).

There have been several incidents of patient data publishing, where identity disclosure has transpired. For instance, Sweeney [112] first demonstrated the problem in 2002, by linking a claims database, which contains information of about $135K$ patients and was disseminated by the Group Insurance Commission, to the voter list of Cambridge, Massachusetts. The linkage was performed, based on patient demographics (e.g., *Date of birth*, *Zip code*, and *Gender*) and led to the re-identification of, William Weld, then governor of Massachusetts. It was also suggested that more than 87% of U.S. citizens could be re-identified, based on such attacks. Many other identity disclosure incidents have been reported since [33]. These include attacks in which (i) students re-identified individuals in the Chicago homicide database by linking it with the social security death index, (ii) an expert witness re-identified most of the individuals represented in a neuroblastoma registry, and (iii) a national broadcaster re-identified a patient, who died

while taking a drug, by combining the adverse drug event database with public obituaries.

Membership and attribute disclosure have not led yet to documented privacy breaches in the healthcare domain. However, they have raised serious privacy concerns and were shown to be feasible in various domains. For example, individuals who were opposed to their potential association with sensitive movies (e.g., movies related to their sexual orientation) took legal action when it was shown that data published by Netflix may be susceptible to attribute disclosure attacks [99].

2.2. Privacy models

In this section, we present some well-established privacy models that guard against the aforementioned threats. These privacy models: (i) model what leads to one or more privacy threats, and (ii) describe a computational strategy to enforce protection against the threat. Privacy models are subsequently categorized according to the privacy threats they protect from, as also presented in Table 2.

2.2.1. Models against identity disclosure

A plethora of privacy models have been proposed to prevent identity disclosure in medical data publishing. These models can be grouped, based on they type of data to which they are applied, into two major categories: (i) models for demographics, and (ii) models for diagnosis codes.

Models for demographics. The most popular privacy model for protecting demographics is k-anonymity [109, 112]. k-anonymity requires each record in a dataset D to contain the same values in the set of Quasi-IDentifier attributes (QIDs) with at least $k-1$ other tuples in D. Recall that quasi-identifiers are typically innocuous

Attack Type	Privacy Models	
	Demographics	**Diagnosis codes**
Identity disclosure	k-anonymity [112] k-map [34] $(1,k)$-anonymity [45] $(k,1)$-anonymity [45] (k,k)-anonymity [45]	complete k-anonymity [52] k^m-anonymity [115] privacy-constrained anonymity [76]
Membership disclosure	δ-presence [100] *c-confident* δ*-presence* [103]	
Attribute disclosure	l-diversity [88, 69] *(a,k)-anonymity* [126] *p-sensitive-k-anonymity* [118] *t-closeness* [69] *range-based* [81, 60] *variance-based* [64] *Worst Group Protection* [84]	ρ*-uncertainty* [16] (h,k,p)*-coherence* [130] *PS-rule based anonymity* [80]

Table 2: Privacy models to guard against different attacks

attributes that can be used in combination to link external data sources with the published dataset. Satisfying k-anonymity offers protection against identity disclosure, because it limits the probability of linking an individual to their record, based on QIDs, to $1/k$. The parameter k controls the level of offered privacy and is set by data publishers, usually to 5 in the context of patient demographics [92].

Another privacy model that has been proposed for demographics is k-map [113]. This model is similar to k-anonymity but considers that the linking is performed based on larger datasets (called *population tables*), from which the published dataset has been derived. Thus, k-map is less restrictive than k-anonymity, typically allowing the publishing of more detailed patient information, which helps data utility preservation. On the negative side, however, the k-map privacy model is weaker (in terms of offered privacy protection) than k-anonymity because it assumes that: (i) attackers do not know whether a record is included in the published dataset, and (ii) data publishers have access to the population table.

El Emam et al. [34] provide a discussion of the k-anonymity and k-map mod-

els and propose risk-based measures, which approximate k-map and are more applicable in certain re-identification scenarios. Three privacy models, called $(1, k)$-*anonymity*, $(k, 1)$-*anonymity* and (k, k)-*anonymity*, which follow a similar concept to k-map, and are relaxations to k-anonymity, have been proposed by Gionis et al. [45]. These models differ in their assumptions about the capabilities of attackers and can offer higher data utility but weaker privacy than k-anonymity.

Models for diagnosis codes. Several privacy models have been proposed to protect identity disclosure attacks when sharing diagnosis codes. The work of He and Naughton [52] proposed *complete k-anonymity*, a model which assumes that any combination of diagnosis codes can lead to identity disclosure and requires at least k records, in the published dataset, to have the same diagnosis codes. Complete k-anonymity, however, may harm data utility unnecessarily because it is extremely difficult for attackers to know all the diagnoses in a patient record [75].

A more flexible privacy model, called k^m-*anonymity*, was proposed by Terrovitis et al. in [115]. k^m-anonymity uses a parameter m to control the maximum number of diagnosis codes that may be known to an attacker, and it requires *each* combination of m diagnosis codes to appear in at least k records of the released dataset. This privacy model is useful in scenarios in which data publishers are unable (or unwilling) to specify certain sets of diagnosis codes that may lead to identity disclosure attacks.

Recently, a privacy model, called *privacy-constrained anonymity*, was introduced by Loukides et al. in [76]. Privacy-constrained anonymity is based on the notion of privacy constraints. These are sets of diagnosis codes that may be known to an attacker and, collectively, they form the *privacy policy*. Given an owner-specified privacy policy, the privacy-constrained anonymity model limits

the probability of performing identity disclosure to at most $1/k$, by requiring the set of diagnoses in *each* privacy constraint to appear at least k times in the dataset (or not appear at all).

By definition, privacy-constrained anonymity assumes that attackers know whether a patient's record is contained in the released dataset. This assumption is made by most research in the field (e.g., [115, 130, 52, 78]), because such knowledge can be obtained by applying the procedure used to create the released data from a larger patient population, which is often described in the literature [75]. Relaxing this assumption, however, is straightforward by following an approach similar to that of the k-map model, and can potentially offer more utility at the expense of privacy. Privacy-constrained anonymity allows protecting only sets of diagnosis codes that may be used in identity disclosure attacks, as specified by the privacy policy. Thus, it addresses a significant limitation of both complete k-anonymity and k^m-anonymity which tend to overly protect the data (i.e., by protecting *all* or *all* m combinations of diagnosis codes), as well as preserving data utility significantly better.

2.2.2. Models against membership disclosure

The privacy models that have been discussed so far are not adequate for preventing membership disclosure, as explained in [100]. To address this shortcoming, two privacy models have been proposed by Nergiz et al. in [100] and [101]. The first of these models, called δ-presence [100], aims at limiting the attacker's ability to infer that an individual's record is contained in a relational dataset D, given a version $\tilde{D}$ of dataset D that is to be published and a public, population table P. The latter table is assumed to contain "all publicly known data" (i.e., the direct identifiers and quasi-identifiers of all individuals in the population, in-

cluding those in D). Satisfying δ-presence offers protection against membership disclosure, because the probability of inferring that an individual's record is contained in table D, using $\tilde{D}$ and P, will be within a range $(\delta_{min}, \delta_{max})$ of acceptable probabilities. A record that is inferred with a probability within this range is called δ-present, and the parameters δ_{min} and δ_{max} are set by data publishers, who also need to possess the population table P.

The fact that δ-presence requires data owners to have access to complete information about the population, in the form of table P, limits its applicability. To address this issue, Nergiz et al. [103] proposed the *c-confident δ-presence* privacy model. This model assumes a set of distribution functions for the population (i.e., attackers know the probability that an individual is associated with one or more values, over one or more attributes) instead of table P, and ensures that a record is δ-present with respect to the population with an owner-specified probability c.

2.2.3. Models against attribute disclosure

Privacy models against sensitive attribute disclosure can be classified into two groups, according to the type of attributes they are applied to: (i) models for patient demographics, and (ii) models for diagnosis codes. In what follows, we describe some representative privacy models from each group.

Models for demographics. The most popular privacy model that thwarts attribute disclosure attacks in patient demographics is l-diversity [88]. It requires each *anonymized group* in a dataset D to contain at least l "well represented" *sensitive attribute* (SA) values [88]. In most cases, an anonymized group is k-anonymous (i.e., it contains at least k records with the same values over the set of quasi-identifiers), although this is not a requirement of the definition of l-diversity. The

simplest interpretation of "well represented" is *distinct* and leads to *distinct* l-*diversity* [69], which requires each anonymized group to contain at least l distinct SA values. Another interpretation leads to *recursive* (c, l)-*diversity* [88], which requires each group in D to contain a large number of distinct SA values, none of which appears "too" often. Other principles that guard against value disclosure by limiting the number of distinct SA values in an anonymized group are (a,k)-*anonymity* [126] and p-*sensitive*-k-*anonymity* [118]. However, these privacy principles still allow attackers to infer that an individual is likely to have a certain SA value when that value appears much more frequently than other values in the group.

t-*closeness* [69] is another privacy model for protecting demographics from attribute disclosure attacks. This model aims at limiting the distance between the probability distribution of the SA values in an anonymized group and that of SA values in the entire dataset. This prevents an attacker from learning information about an individual's SA value that is not available from the dataset. Consider, for example, a dataset in which 60% of tuples have the value `Flu` in a SA `Disease`, and we form an anonymous group, which also has 60% of its disease values as `Flu`. Then, although an attacker can infer that an individual in the group suffers from `Flu` with relatively high probability (i.e. 60%), the group is protected according to t-closeness, since this fact can be inferred from the dataset itself.

Privacy models to guard against the disclosure of sensitive ranges of values in numerical attributes have also been proposed. Models that work by limiting the maximum range of SA values in a group of tuples have been proposed by Loukides et al. [81] and Koudas et al. [60], while LeFevre et al. [64] proposed limiting the variance of SA values instead. A privacy model, called *Worst Group*

Protection (WGP), which prevents range disclosure and can be enforced without generalization of SA values was introduced in [84]. WGP measures the probability of disclosing any range in the least protected group of a table, and captures the way SA values form ranges in a group, based on their frequency and similarity.

Models for diagnosis codes. Several privacy models have been proposed to protect attribute disclosure attacks when sharing diagnosis codes (e.g., the association of patients with sensitive diagnosis codes, such as those representing sexually transmitted diseases). One such model, proposed by Cao et al. [16], is *ρ-uncertainty*, which limits the probability of associating an individual with any (single) diagnosis code to less than ρ. This model makes the (stringent) assumption that each diagnosis code in a patient record can be sensitive, and all the remaining codes in the record may be used for its inference.

Another privacy model, called *(h, k, p)-coherence*, was proposed in [130] and guards against both identity and sensitive information disclosure. This model treats non-sensitive diagnosis codes similarly to k^m-anonymity and limits the probability of inferring sensitive diagnosis codes. In fact, parameters k and p have a similar role to k and m in k^m-anonymity, and h limits the probability of attribute disclosure.

The *PS-rule based anonymity* model (*PS-rule* stands for *Privacy Sensitive rule*), proposed by Loukides et al. in [80], also thwarts both identity and sensitive information disclosure. Similarly to *association rules* [7], PS-rules consist of two sets of diagnosis codes, the antecedent and consequent, which contain diagnosis codes that may be used in identity and sensitive information disclosure attacks, respectively. Given a PS-rule $A \rightarrow B$, where A and B is the antecedent and consequent of the rule, respectively, PS-rule based anonymity requires that the set of

diagnosis codes in A appears in at least k records of the published dataset, while at most $c \cdot 100\%$ of the records that contain the diagnosis codes in A, also contain the diagnosis codes in B. Thus, it protects against attackers who know whether a patient's record is contained in the published dataset. The parameter c is specified by data publishers, takes values between 0 and 1, and is analogous to the confidence threshold in association rule mining [7]. The PS-rule based anonymity model offers three significant benefits compared to previously discussed models for diagnosis codes: (i) it protects against both identity and sensitive information disclosure, (ii) it allows data publishers to specify detailed privacy requirements, and (iii) it is more general than these models (i.e., the models in [115, 130, 52] are special cases of PS-rule based anonymity).

3. Privacy scenarios

There are two popular scenarios for privacy-preserving data sharing, as illustrated in Figure 1. In this paper, we survey privacy models and algorithms that belong to the non-interactive data sharing scenario. This scenario has certain benefits: (i) it offers constant data availability (since the original dataset is published after being anonymized), (ii) it does not require any infrastructure costs, and (iii) it is good for hypothesis generation and testing (since patient records are published in a utility-aware, anonymized form). However, the non-interactive scenario suffers from two important shortcomings. First, data owners need to specify privacy and utility requirements prior to sharing their data, in order to ensure that the released dataset is adequately protected and highly useful. Second, data owners have no control over the released dataset. Thus, the released dataset may be susceptible to attacks that had not been discovered at the time of data release.

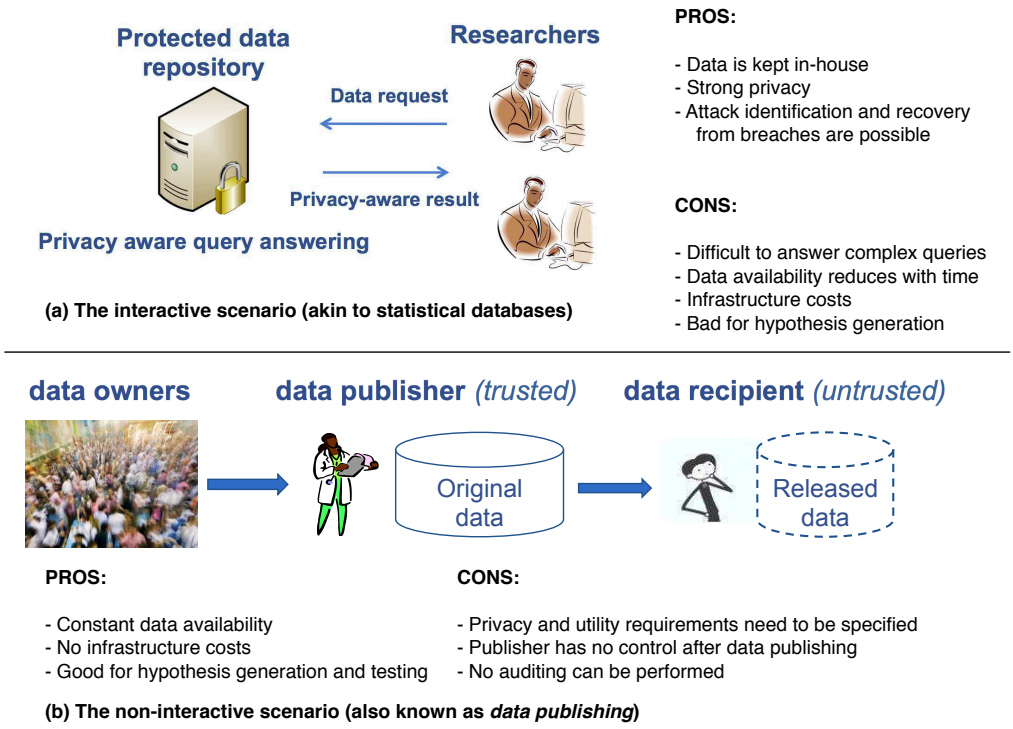

Figure 1: Privacy-preserving data sharing scenarios: (a) interactive vs. (b) non-interactive

Privacy-preserving data sharing can also be facilitated in the non-interactive scenario. This scenario assumes that the data are deposited into a (secure) repository and can be queried by external data users. Thus, the users receive protected answers to their queries, and not the entire dataset, as in the non-interactive scenario. The interactive scenario offers three main benefits, which stem from the fact that data are kept in-house to the hosting organization.

First, data owners can audit the use of their data and apply access control policies. This ensures that attackers can be identified and held accountable, a capability that is not offered by techniques that are designed for the non-interactive scenario. Furthermore, the enforced protection mechanism for the repository can

be improved at any time based on new privacy threats that are identified, thus data owners can provide state-of-the-art protection of the sensitive data in the repository. Second, the interactive scenario allows the enforcement of strong, semantic privacy models that will be discussed later. Third, the fact that the types of posed queries are known a-priori to data owners helps deciding on an appropriate level of privacy that should be offered when answering the queries.

On the other hand, complex queries are difficult to support in the interactive setting, while there are often restrictions on the number of queries that can be answered. Additionally, several analytic tasks (e.g., visualization) require individual records, as opposed to aggregate results or models. These tasks are difficult to be supported in the interactive scenario. In general, it is interesting to observe that the advantages of the interactive scenario are disadvantages of the non-interactive scenario, and vice versa. Consequently, data publishers need to carefully select the appropriate privacy-preserving data sharing scenario based on their needs.

A popular class of algorithms that are designed for the interactive scenario enforce privacy by adding noise to each query answer, thereby offering *output privacy*. The goal of these algorithms is to tune the magnitude of the added noise so that privacy is preserved, while accurate, high-level statistics can still be computed using queries. For instance, several algorithms that enforce *differential privacy* [29], a strong privacy model to be discussed later, in the interactive setting, are surveyed in [30]. In addition to constructing protected query answers, these algorithms monitor the number of queries posed to the system and stop answering queries, when the maximum number of queries that can be answered, while satisfying differential privacy, is reached. The release of statistics in a privacy-preserving way has also been thoroughly investigated by the statistical disclosure

Attack Type	Privacy Models	
	Demographics	**Diagnosis codes**
Identity disclosure	k-Minimal Generalization [109] OLA [32] Incognito [65] Genetic [58] Mondrian [66, 67] TDS [40] NNG [28] Greedy [129] k-Member [15] KACA [68] Agglomerative [45] (k, k)-anonymizer [45] Hilb [44] iDist [44] MDAV [25] CBFS [62]	UGACLIP [76] CBA [87] UAR [86] Apriori [115] LRA [116] VPA [116] mHgHs [74] Recursive Partition [52]
Membership disclosure	SPALM [100] MPALM [100] SFALM [101]	
Attribute disclosure	Incognito with l-diversity [88] Incognito with t-closeness [69] Incognito with (a, k)-anonymity [126] p-sensitive k-anonymity [118] Mondrian with l-diversity [127] Mondrian with t-closeness [70] Top Down [126] Greedy algorithm [81] Hilb with l-diversity [44] iDist with l-diversity [44] Anatomize [127]	Greedy [130] SuppressControl [16] TDControl [16] RBAT [79] Tree-based [80] Sample-based [80]

Table 3: Algorithms to prevent against different attacks

control community (see [4] for a survey). However, the techniques in [4] do not guarantee privacy preservation using a rigorous privacy model [29].

4. Privacy techniques

In this section, we provide a classification of algorithms that employ the privacy models in Section 2 and have been designed for the non-interactive scenario. These algorithms are summarized in Table 3. For each class of algorithms, we also discuss techniques that are employed in their operation.

4.1. Algorithms against identity disclosure

The prevention of identity disclosure requires transforming quasi-identifiers to enforce a privacy model in a way that preserves data utility. Since transforming the data to achieve privacy and optimal utility is computationally infeasible (see for example [109]), most algorithms adopt heuristic strategies to explore the space of possible solutions. That is, they consider different ways of transforming quasi-identifiers in order to find a "good" solution that satisfies privacy and the utility objective. After discussing approaches to transform quasi-identifiers, we survey utility objectives and heuristic strategies. Based on this, we subsequently present a classification of popular algorithms.

4.1.1. Transforming quasi-identifiers

There are three main techniques to transform quasi-identifiers in order to prevent identity disclosure: (i) microaggregation [24], (ii) generalization [109], and (iii) suppression [109]. Microaggregation involves replacing a group of values in a QID using a summary statistic (e.g., centroid or median for numerical and categorical QIDs, respectively). This technique has been applied to demographics but not to diagnosis codes. Generalization, on the other hand, suggests replacing QID values by more general, but semantically consistent, values. Two generalization models, called *global* and *local* recoding, have been proposed in the literature (see [102] for an excellent survey of generalization models). Global recoding involves mapping the domain of QIDs into generalized values. These values correspond to aggregate concepts (e.g., *British* instead of *English*, for Ethnicity) or collections of values (e.g., *English or Welsh*, for Ethnicity, or *18 to 30*, for Age). Thus, all occurrences of a certain value (e.g., *English*) in a dataset will be generalized to the same value (e.g., *European*). On the other hand, local recoding involves map-

ping QID values of individual records into generalized ones on a group-by-group basis. Therefore, the value *English* in two different records may be replaced by *British* in one record, and by *European*, in another. Similarly, diagnosis codes can be replaced either by aggregate concepts (e.g., *Diseases of Other Endocrine Glands* instead of *Diabetes melitus type I*) or by sets of diagnosis codes (e.g., {*Diabetes melitus type I*, *Diabetes melitus type II*}), which are interpreted as any (non-empty) subset of diagnosis codes contained in the set. Last, suppression involves the deletion of specific QID values from the data.

Although each technique has its benefits, generalization is typically preferred over microaggregation and suppression. This is because microaggregation may harm data truthfulness (i.e., the centroid may not appear in the data), while suppression incurs high information loss. Interestingly, there are techniques that employ more than one of these operations. For example, the work of [76] employs suppression when it is not possible to apply generalization while satisfying some utility requirements.

4.1.2. Utility objectives

Preventing identity disclosure may lower the utility of data, as it involves data transformation. Thus, existing methods aim at preserving data utility by following one of the following general strategies: (i) they quantify information loss using an optimization measure, which they attempt to minimize, (ii) they assume that data will be used in a specific data analysis task and attempt to preserve the accuracy of performing this task using the published data, and (iii) they take into account utility requirements, specified by data owners, and aim at generating data that satisfy these requirements. In what follows, we discuss each of these strategies.

One way to capture data utility is by measuring the level of information loss

incurred by data transformation. The measures that have been proposed are based on (i) the size of anonymization groups, or (ii) the characteristics of generalized values. Measures of the first category are based on the intuition that all records in an anonymization group are indistinguishable from one another, as they have the same value over QIDs. Thus, larger groups incur more information loss. Examples of these measures are *Discernability Metric* (DM) [9] and *Normalized Average Equivalence Class Size* [66], which differ from one another in the way they penalize groups. The main drawback of these measures is that they neglect the way values are transformed within an anonymized group. These measures, for example, would assign the same penalty to a group of records with values $\{14, 15, 16\}$ in a QID *Age* that are generalized to *14 to 16* or *Underage*. However, using the generalized value *14 to 16* incurs lower information loss, as this is more specific than *Underage*.

The above-mentioned limitation is addressed by the second category of measures, which take into account the way values are generalized. Examples of these measures are *Generalization Cost* (GC) [6], *Normalized Certainty Penalty* (NCP) [129], and *Loss Metric* (LM) [58]. All of these measures are applicable to demographics and penalize less specific generalized values (i.e., they favor *British* over *European*) but the latter two (i.e., NCP and LM) are more flexible, as they can be applied to both numerical and categorical attributes. A recently proposed information-loss measure for diagnosis codes is *Information Loss Metric* (ILM) [87]. ILM quantifies information loss of a generalized diagnosis code by imposing a large penalty on generalized terms that contain many diagnosis codes and appear in many records of the dataset.

Another way to capture data utility is based on measuring the accuracy of a

specific task performed on anonymized data. Iyengar [58], for example, observed that generalization can make it difficult to build an accurate classification model. This is because records with different class labels become indistinguishable from one another, when they fall into the same anonymization group. For example, assume that all records, whose value in Ethnicity is *Welsh*, have a classification label *Yes*, whereas all records with *English* have a label *No*. Generalizing the values *Welsh* and *English* to *British* does not allow to distinguish between records that have different classification labels. To capture data utility, Iyengar introduced the *Classification Metric* (CM), which is expressed as the number of records whose class labels are different from that of the majority of records in their anonymized group, normalized by the dataset size.

LeFevre et al. [66] considered measuring the utility of anonymized data when used for aggregate query answering purposes and proposed a measure, called *Average Relative Error* (ARE). ARE quantifies data utility by measuring the difference between the answers to a query using the anonymized and using the original data. This measure has been widely employed, as it is applicable to different types of data (e.g., both demographics and diagnosis codes) and is independent of the way data are anonymized. Fung et al. [41], on the other hand, considered clustering and proposed comparing the cluster structures of the original and anonymized data, using the F-measure [122] and Match point. Although these measures are also general, currently they have only been applied to demographics.

Several publishing scenarios involve the release of an anonymized dataset to support a specific medical study, or to data recipients having certain data analysis requirements. In such scenarios, knowledge of how the dataset will be analyzed can be exploited during anonymization to better preserve data utility. For example,

consider a dataset which contains `Age`, `Gender`, `Ethnicity`, and `Marital Status`, as quasi-identifiers, and needs to be released for performing a study on the age of female patients. Intuitively, distorting the values of the first two attributes should be avoided, as the result of the study depends on their values. Samarati proposed modeling data analysis requirements based on the minimum number of suppressed tuples, or on the height of hierarchies for categorical QID values [109]. However, such requirements are difficult to be specified by data publishers, as they require knowledge of how the dataset will be anonymized.

Xu et al. [129] prioritized the anonymization of certain quasi-identifier attributes by using data-owner specified weights. The proposed approach, however, cannot guarantee that some attributes will not be overdistorted (e.g., gender information can be lost, even when the generalization of `Ethnicity` is preferred to that of `Gender`). To guarantee that the anonymized data will remain useful for the specified analysis requirements, Loukides et al. [85] proposed a model for expressing data utility requirements and an algorithm for anonymizing data, based on this model. Utility requirements can be expressed at an attribute level (e.g., imposing the length of range, or the size of set that anonymized groups may have in a given quasi-identifier attribute), or at a value level (e.g., imposing ranges or sets allowed for specified values). The approach of [85] can be applied to patient demographics but not to diagnosis codes.

Anonymizing diagnosis codes in a way that satisfies data utility requirements has been considered in [76]. The proposed approach models data utility requirements using sets of diagnosis codes, referred to as *utility constraints*. A utility constraint represents the ways the codes, contained in it, can be generalized in order to preserve data utility. Thus, utility constraints specify the information that

the anonymized data should retain in order to be useful in intended medical analysis tasks. For example, assume that the disseminated data must support a study that requires counting the number of patients with *Diabetes*. To achieve this, a utility constraint for *Diabetes*, which is comprised of all different types of diabetes, must be specified. By anonymizing data according to this utility constraint, we can ensure that the number of patients with *Diabetes* in the anonymized data will be the same as in the original data. Thus, the anonymized dataset will be as useful as the original one, for the medical study on diabetes.

4.1.3. Heuristic strategies

Optimally anonymizing data with respect to the aforementioned utility criteria is computationally infeasible (see for example [66, 129, 78]). Consequently, many anonymization methods employ heuristic search strategies to form anonymous groups. In what follows, we discuss search strategies that have been applied to demographics and diagnosis codes.

Algorithms for demographics. Algorithms for demographics typically employ: (i) binary search on the lattice of possible generalizations [109], (ii) a lattice search strategy similar in principle to the Apriori [7] used in association rule mining, (iii) genetic search on the lattice of possible generalizations [58], (iv) data partitioning [66, 57], (v) data clustering [102, 129, 81, 68], or (vi) space mapping [44].

The main idea behind strategies (i) to (iii) is to represent the possible ways to generalize a value in a quasi-identifier attribute, using a taxonomy, and then combine the taxonomies for all quasi-identifier attributes, to obtain a lattice. For instance, *English* and *Welsh* are the leaf-level nodes of a taxonomy for `Ethnicity` and their immediate ascendant is the generalized value *British*. Similarly, *Male*

and *Female* are the leaf-level nodes of a taxonomy for `Gender`, whose root value and immediate ascendant of the leaves is *Any*. Thus, we can combine these two taxonomies to get a lattice for `Ethnicity` and `Gender`. Each node in this lattice represents a different set of generalized values for `Ethnicity` and `Gender`, such as {*English*, *Male*}, {*English*, *Female*}, {*Welsh*, *Male*}, and {*British*, *Any*}. Thus, finding a way to generalize values can be performed by exploring the lattice using heuristics that avoid considering certain lattice nodes for efficiency reasons. The strategy (i) prunes the ascendants of lattice nodes that are sufficient to satisfy a privacy model, while the strategies (ii) and (iii) prune lattice nodes that are likely to incur high utility loss. The latter nodes are identified while considering nodes that represent incrementally larger sets of generalized values, for strategy (i), or while selecting nodes by combining their descendants, as specified by a genetic algorithm, in the case of strategy (ii).

Binary and Apriori-like lattice search strategies explore a small space of potential solutions and thus may fail to preserve data utility to the extent that genetic search strategies can do. However, genetic search is computationally intensive (e.g., the algorithm in [58] is orders of magnitude slower than the partitioning-based method of [66]) and may converge slowly. Consequently, more recent research has focused on developing methods that use strategies (iv) and (v), which are applied to the records of a dataset, and not to attribute values as strategies (i) to (iii) are. The objective of the former strategies is to organize records into carefully selected groups that help the preservation of privacy and the satisfaction of a utility objective. Both data partitioning and clustering-based strategies create groups iteratively, but they differ in the task they perform in an iteration. Specifically, partition-based strategies split records into groups, based on the value that these

records have in a single quasi-identifier attribute (i.e., an iteration creates two typically large groups of records that are similar with respect to a quasi-identifier), while clustering-based strategies merge two groups of records, based on the values of the records in all quasi-identifier attributes together. Therefore, partitioning-based methods tend to incur higher utility loss when compared to clustering-based methods [129, 81], and they are sensitive to the choice of the splitting attribute, performing poorly particularly when the dataset is skewed [102]. However, partitioning is faster than clustering by orders of magnitude, requiring $O(n \cdot log(n))$ time instead of $O(n^2)$, where n is the cardinality of the dataset.

A different heuristic search strategy relies on space mapping techniques [44]. These techniques create a ranking of records, such that records with similar values in quasi-identifiers have similar ranks. Based on this ranking, groups of records are subsequently formed by considering a number of records (e.g., at least k for k-anonymity) that have consecutive ranks. Space mapping techniques achieve good efficiency, as the ranking can be calculated in linear time, as well as being effective at preserving data utility.

Algorithms for diagnosis codes. Algorithms for diagnosis codes employ: (i) space partitioning in a bottom-up [76] or top-down [16] fashion, (ii) space clustering [46], or (iii) data partitioning in a top-down [52], vertical or horizontal [116] way. Clearly, lattice search cannot be used in the context of diagnosis codes, because there is a single, set-valued attribute to consider. Thus, one taxonomy which organizes diagnosis codes, and not a lattice of taxonomies, is used to model the ways these codes can be generalized. In addition, the space mapping techniques considered by Ghinita et al. in [44] are not applicable to diagnosis codes because there is a single, set-valued quasi-identifier attribute (i.e., a patient can be re-identified

using a set of their diagnosis codes) and not many quasi-identifier attributes, as in the case of patient demographics.

Both strategies (i) and (ii) attempt to find a set of generalized diagnosis codes that can be used to replace diagnosis codes in the original dataset (e.g., "diabetes" that replaces "diabetes mellitus type I" and "diabetes mellitus type II"). However, they differ in the way they operate. Specifically, space partitioning strategies require a taxonomy for diagnosis codes, which is provided by data owners (e.g., a healthcare institution), and dictate that the generalized diagnosis codes are part of the taxonomy. Space clustering strategies lift this requirement and are more effective in terms of preserving data utility. On the other hand, data partitioning strategies are applied to transactions (records) instead of diagnosis codes, and they aim to create groups of transactions that can be subsequently anonymized with low data utility loss. For example, assume that privacy is preserved by applying k^m-anonymity with $k = 2$ and $m = 2$. Two transactions with exactly the same diagnosis codes are already 2^2-anonymous, and thus they do not incur data utility loss as they can be released intact.

Space partitioning allows searching only a smaller space of possible solutions and typically results in incurring high information loss when compared to space clustering strategies. On the other hand, space clustering-based strategies are computationally intensive. It is important to note that the worst-case complexity of all strategies is exponential to the number of distinct diagnosis codes in a dataset, which can be in the order of several hundreds. This explains need for developing more effective and efficient strategies.

Algorithm	Privacy model	Transformation	Utility Objective	Heuristic strategy
k-Minimal Generalization [109]	k-anonymity	generalization and suppression	min. inf. loss	binary lattice search
OLA [32]	k-anonymity	generalization	min. inf. loss	binary lattice search
Incognito [65]	k-anonymity	generalization and suppression	min. inf. loss	apriori-like lattice search
Genetic [58]	k-anonymity	generalization	classification accuracy	genetic search
Mondrian [66]	k-anonymity	generalization	min. inf. loss	data partitioning
LSD Mondrian [67]	k-anonymity	generalization	regression accuracy	data partitioning
Infogain Mondrian [67]	k-anonymity	generalization	classification accuracy	data partitioning
TDS [40]	k-anonymity	generalization	classification accuracy	data partitioning
NNG [28]	k-anonymity	generalization	min. inf. loss	data partitioning
Greedy [129]	k-anonymity	generalization	min. inf. loss	data clustering
k-Member [15]	k-anonymity	generalization	min. inf. loss	data clustering
KACA [68]	k-anonymity	generalization	min. inf. loss	data clustering
Agglomerative [45]	k-anonymity	generalization	min. inf. loss	data clustering
(k,k)-anonymizer [45]	(k,k)-anonymity	generalization	min. inf. loss	data clustering
Hilb [44]	k-anonymity	generalization	min. inf. loss	space mapping
iDist [44]	k-anonymity	generalization	min. inf. loss	space mapping
MDAV [25]	k-anonymity	microaggregation	min. inf. loss	data clustering
CBFS [62]	k-anonymity	microaggregation	min. inf. loss	data clustering

Table 4: Algorithms for preventing identity disclosure based on demographics

4.1.4. Classification of algorithms

We now present a classification of algorithms for preventing identity disclosure, based on the strategies they adopt for (i) transforming quasi-identifiers, (ii) preserving utility, and (iii) heuristically searching for a "good" solution.

Algorithms for demographics. Table 4 presents a classification of algorithms for demographics. As can be seen, these algorithms employ various data transformation and heuristic strategies, and aim at satisfying different utility objectives. All algorithms adopt k-anonymity, with the exception of (k, k)-anonymizer [45] which adopts the (k, k)-anonymity model, discussed in Section 2.2. The fact that (k, k)-anonymity is a relaxation of k-anonymity allows the algorithm in [45] to preserve more data utility than the Agglomerative algorithm, which is also pro-

posed in [45]. Furthermore, most algorithms use generalization to anonymize data, except (i) the algorithms in [109, 65], which use suppression in addition to generalization in order to deal with a typically small number of values that would incur excessive information loss if generalized, and (ii) the algorithms in [25, 62], which use microaggregation.

In addition, it can be observed that the majority of algorithms aim at minimizing information loss and that no algorithm takes into account specific utility requirements, such as limiting the set of allowable ways for generalizing a value in a quasi-identifier. At the same time, the Genetic [58], Infogain Mondrian [67], and TDS [40] algorithms aim at releasing data in a way that allows for building accurate classifiers. These algorithms were compared in terms of how well they can support classification tasks, using publicly available demographic datasets [53, 119]. The results are reported in [67] and [40], and demonstrate that Infogain Mondrian outperforms TDS which, in turn, outperforms the Genetic algorithm. The LSD Mondrian [67] algorithm is similar to Infogain Mondrian but uses a different utility objective measure, as its goal is to preserve the ability of using the released data for linear regression.

It is also interesting to observe that several algorithms implement data partitioning heuristic strategies. Specifically, the algorithms proposed in [66, 67] follow a top-down partitioning strategy inspired by kd-trees [38], while the TDS algorithm [40] employs a different strategy that takes into account the partition size and data utility in terms of classification accuracy. Interestingly, the partitioning strategy of NNG [28] is based on the distance of values and allows creating k-anonymous datasets, whose utility is no more than $6 \cdot q$ times worse than that of the optimal solution, where q is the number of quasi-identifiers in the dataset.

On the other hand, the algorithms that employ clustering [129, 15, 98, 45] follow a similar greedy, bottom-up procedure, which aims at building clusters of at least k records by iteratively merging together smaller clusters (of one or more records), in a way that helps data utility preservation. A detailed discussion and evaluation of clustering-based algorithms that employ generalization has been reported in [82], while the authors of [25] and [26] provide a rigorous analysis of clustering-based algorithms for microaggregation.

The use of space mapping techniques in algorithms iHilb and iDist, both of which were proposed in [44], enables them to preserve data utility equally well or even better than the Mondrian algorithm [66] and to anonymize data more efficiently. To map the space of quasi-identifiers, iHilb uses the Hilbert curve, which can preserve the locality of points (i.e., values in quasi-identifiers) fairly well [97]. The intuition behind using this curve is that, with high probability, two records with similar values in quasi-identifiers will also be similar with respect to their rank that is produced based on the curve. The iDist algorithm employs iDistance [131], a technique that measures similarity based on sampling and clustering of points, and is shown to be slightly inferior than iHilb in terms of data utility. Last, the algorithms in [109, 65, 32] use lattice-search strategies. An experimental evaluation using a publicly available dataset containing demographics [53], as well as 5 hospital discharge summaries, shows that the OLA algorithm [32] performs similarly to Incognito [65] and better than k-Minimal Generalization [109] in terms of preserving data utility. The authors of [32] also suggest that the way OLA generalizes data might help medical data analysts. Nevertheless, algorithms that use lattice-based search strategies typically explore a smaller number of generalizations than algorithms that employ data partitioning or clustering, and are

generally less effective at preserving data utility.

Algorithms for diagnosis codes. Algorithms for anonymizing diagnosis codes are summarized in Table 5. Observe that these algorithms adopt different privacy models, but they all use either a combination of generalization and suppression, or generalization alone in order to anonymize datasets. More specifically, the algorithms in [76, 87, 86] use suppression as a secondary operation and only when generalization alone cannot be used to satisfy the specified utility constraints. However, they differ in that CBA and UAR consider suppressing individual diagnosis codes, whereas UGACLIP suppresses sets of typically more than one diagnosis codes. Experiments using patient records derived from the Electronic Medical Record (EMR) system of Vanderbilt University Medical Center, which are reported in [87, 86], showed that the suppression strategy that is employed by CBA and UAR is more effective than that of UGACLIP.

Algorithm	**Privacy model**	**Transformation**	**Utility Objective**	**Heuristic strategy**
UGACLIP [76]	privacy-constrained anonymity	generalization and suppression	utility requirements	bottom-up space partitioning
CBA [87]	privacy-constrained anonymity	generalization and suppression	utility requirements	space clustering
UAR [86]	privacy-constrained anonymity	generalization and suppression	utility requirements	space clustering
Apriori [115]	k^m-anonymity	generalization	min. inf. loss	top-down space partitioning
LRA [116]	k^m-anonymity k^m-anonymity	generalization	min. inf. loss	horizontal data partitioning
VPA [116]	k^m-anonymity k^m-anonymity	generalization	min. inf. loss	vertical data partitioning
mHgHs [74]	k^m-anonymity	generalization and suppression	min. inf. loss	top-down space partitioning
Recursive partition [52]	complete k-anonymity	generalization	min. inf. loss	data partitioning

Table 5: Algorithms for preventing identity disclosure based on diagnosis codes

Furthermore, the algorithms in Table 5 aim at either satisfying utility requirements, or at minimizing information loss. The UGACLIP, CBA, and UAR al-

gorithms adopt *utility constraints* to formulate utility requirements and attempt to satisfy them. However, these algorithms still favor solutions with low information loss, among those that satisfy the specified utility constraints. All other algorithms attempt to minimize information loss, which they quantify using two different measures; a variation of the *Normalized Certainty Penalty* (NCP) measure [129] for the algorithms in [115, 116, 52], or the *Loss Metric* (LM) [58] for the mHgHs algorithm [74]. However, to our knowledge, there are no algorithms for diagnosis codes that aim at preserving data utility for intended mining tasks, such as classification. Given the extensive use of diagnosis codes in these tasks, we believe that the development of such algorithms merits further investigation.

It is also interesting to observe that several algorithms implement data partitioning heuristic strategies. Specifically, the algorithms proposed in [66, 67] follow a top-down partitioning strategy inspired by kd-trees [38], while the TDS algorithm [40] employs a different strategy that takes into account the partition size and data utility in terms of classification accuracy. Interestingly, the partitioning strategy of NNG [28] is based on the distance of values and allows creating a k-anonymous dataset, whose utility is no more than $6 \cdot q$ times worse than that of the optimal solution, where q is the number of quasi-identifiers in the dataset. On the other hand, the algorithms that employ clustering [129, 15, 98, 45] follow a similar greedy, bottom-up procedure, which aims at building clusters of at least k records by iteratively merging together smaller clusters of records, in a way that helps data utility preservation. A detailed discussion and evaluation of clustering-based algorithms that employ generalization has been reported in [82].

Moreover, it can be seen that all algorithms in Table 5 operate on either the space of diagnosis codes, or on that of the records of the dataset to be published.

Specifically, UGACLIP [76] partitions the space of diagnosis codes in a bottom-up manner, whereas Apriori [115] and mHgHs [74] employ top-down partitioning strategies. The strategy of UGACLIP considers a significantly larger number of ways to generalize diagnosis codes than that of Apriori, which allows for better data utility preservation. In addition, the space clustering strategies of CBA and UAR are far more effective than the bottom-up space partitioning strategy of UGACLIP, but they are also more computationally demanding.

Data partitioning strategies are employed by the Recursive partition [52], LRA [116] and VPA [116] algorithms. The first of these algorithms employs a top-down partitioning strategy, which is applied recursively. That is, it starts by a dataset which contains (i) all transactions of the dataset to be published, and (ii) a single generalized diagnosis code *Any*, which replaces all diagnosis codes. This dataset is split into subpartitions of at least k transactions, which contain progressively less general diagnosis codes (e.g., *Any* is replaced by *Diabetes* and then by *Diabetes mellitus type I*). The strategy employed by the Recursive partition algorithm enforces complete k-anonymity with lower utility loss than that of Apriori [52]. On the other hand, LRA and VPA use horizontal and vertical data partitioning strategies, respectively. Specifically, LRA attempts to create subpartitions of transactions with "similar" items that can be generalized with low information loss. To achieve this, it sorts the transactions in the dataset to be published based on Gray ordering [106] and then groups these transactions into subpartitions of approximately equal size. VPA partitions data records vertically in order to create sub-records (i.e., parts of transactions) with "similar" items. The Apriori algorithm, discussed above, is then used by the LRA and VPA algorithms for anonymizing each of the created subpartitions, separately.

4.2. Techniques against membership disclosure

The fact that membership disclosure cannot be forestalled by simply preventing identity disclosure, calls for specialized algorithms. However, as can be seen in Table 6, the proposed algorithms for membership disclosure share the same main components (i.e., quasi-identifier transformation strategy, utility objective, and heuristic strategy) with the algorithms that protect from identity disclosure. Furthermore, these algorithms are all applied to demographics.

Algorithm	**Data type**	**Privacy model**	**Transformation**	**Utility Objective**	**Heuristic strategy**
SPALM [100]	demographics	δ-presence	generalization	min. inf. loss	top-down lattice search
MPALM [100]	demographics	δ-presence	generalization	min. inf. loss	top-down lattice search
SFALM [101]	demographics	c-confident δ-presence	generalization	min. inf. loss	top-down lattice search

Table 6: Algorithms for preventing membership disclosure

All existing algorithms against membership disclosure have been proposed by Nergiz et al. [100, 101], to the best of our knowledge. In [100], they proposed two algorithms, called SPALM and MPALM, which transform quasi-identifiers, using generalization, and aim at finding a solution that satisfies δ-presence with low information loss. Both algorithms adopt a top-down search on the lattice of all possible generalizations, but they differ in their generalization model. Specifically, the SPALM algorithm generalizes the values of each quasi-identifier separately, requiring all values in a quasi-identifier to be generalized in the same way (e.g., all values *English*, in *Ethnicity*, are generalized to *British*). On the contrary, the MPALM algorithm drops this requirement, allowing two records with the same value in a quasi-identifier to be generalized differently (e.g., one value *English* to be generalized to *British* and another to *European*). In a subsequent work [101], Nergiz et al. proposed an algorithm called SFALM, which is similar to SPALM

but employs c-confident δ-presence. The fact that the latter privacy model does not require complete information about the population, as discussed above, greatly improves the applicability of SFALM in practice.

The aforementioned algorithms against membership disclosure are limited in their choice of data transformation strategies and utility objectives, since they all employ generalization and aim at minimizing information loss. We believe that developing algorithms that adopt different data transformation strategies (e.g., microaggregation) and utility objectives (e.g., utility requirements) is worthwhile. At the same time, the algorithms in [100, 101] are not applicable to diagnosis codes, because diagnosis codes have different semantics than demographics. However, it is easy to see that membership disclosure attacks based on diagnosis codes are possible, because diagnosis codes can be used to reveal the fact that a patient's record is contained in the published dataset. This calls for developing algorithms for sharing diagnosis codes in a way that forestalls membership disclosure.

4.2.1. Techniques against attribute disclosure

In what follows, we discuss privacy considerations that are specific for algorithms that aim at thwarting attribute disclosure. Subsequently, we present a classification of these algorithms.

Algorithms for preventing attribute disclosure enforce privacy principles that govern the associations between quasi-identifier and sensitive values (e.g., *Income* in a demographics dataset or *Schizophrenia* in a dataset containing diagnosis codes). To enforce these principles, they create anonymous groups and then merge them iteratively, until the associations between these attributes and sensitive values become protected, according to a certain privacy model (e.g., l-diversity) [88, 118, 126, 69, 67]. While this can be achieved using generalization

and / or suppression, a technique called *bucketization* has been proposed in [127] as a viable alternative. Bucketization works by releasing: (i) a projection D_q of the dataset D on the set of quasi-identifiers, and another projection, D_s, on the sensitive attribute, and (ii) a group membership attribute that specifies the associations between records in D_q and D_s. By carefully constructing D_q and D_s, it is possible to enforce l-diversity with low information loss [127], as values in quasi-identifiers are released intact. However, the algorithm in [127] does not guarantee that identity disclosure will be prevented.

Many of the algorithms considered in this section follow the same data transformation strategies and utility objectives, with the algorithms examined in Section 4, but they additionally ensure that sensitive values are protected within each anonymized group. This approach helps data publishers construct data that are no more distorted than necessary to thwart attribute disclosure, and the algorithms following this approach are termed *protection constrained*. Alternatively, data publishers may want to produce data with a desired trade-off between data utility and privacy protection against identity disclosure. This is possible using *trade-off constrained* algorithms [81, 83, 84]. These algorithms quantify and aim at optimizing the trade-off between the distortion caused by generalization and the level of data protection against attribute disclosure.

Algorithms for demographics. A classification of algorithms for demographics is presented in Table 7. As can be seen, the majority of these algorithms follow the protection-constrained approach and are based on algorithms for identity disclosure, such as Incognito [65], Mondrian [66], iHilb [44], or iDist [44]. Furthermore, most of these algorithms employ generalization, or a combination of generalization and suppression, and they enforce l-diversity, t-closeness, p-sensitive

Algorithm	Privacy model	Transformation	Approach	Heuristic strategy
Incognito with l-diversity [88]	l-diversity	generalization and suppression	protection constrained	apriori-like lattice search
Incognito with t-closeness [69]	t-closeness	generalization and suppression	protection constrained	apriori-like lattice search
Incognito with (a, k)-anonymity [126]	(a, k)-anonymity	generalization and suppression	protection constrained	apriori-like lattice search
p-sens k-anon [118]	p-sensitive k-anonymity	generalization	protection constrained	apriori-like lattice search
Mondrian with l-diversity [127]	l-diversity	generalization	protection constrained	data partitioning
Mondrian with t-closeness [70]	t-closeness	generalization	protection constrained	data partitioning
Top Down [126]	(a, k)-anonymity	generalization	protection constrained	data partitioning
Greedy algorithm [81]	tuple diversity	generalization and suppression	trade-off constrained	data clustering
Hilb with l-diversity [44]	l-diversity	generalization	protection constrained	space mapping
iDist with l-diversity [44]	l-diversity	generalization	protection constrained	space mapping
Anatomize [127]	l-diversity	bucketization	protection constrained	quasi-identifiers are released intact

Table 7: Algorithms for preventing attribute disclosure based on demographics

k-anonymity, (a, k)-anonymity, or tuple-diversity. An exception is the Anatomize algorithm [127], which was specifically developed for enforcing l-diversity using bucketization. This algorithm works by creating buckets with the records that have the same value in the sensitive attribute and then constructing groups with at least l different values in the sensitive attribute. The construction of groups is performed by selecting records from appropriate buckets, in a round-robin fashion. Interestingly, the Anatomize algorithm requires an amount of memory that is linear to the number of distinct values of the sensitive attribute and creates anonymized data with bounded *reconstruction error*, which quantifies how well correlations among values in quasi-identifiers and the sensitive attribute are preserved. In fact, the authors of [127] demonstrated experimentally that the Anatomize algorithm outperforms an adaption of the Mondrian algorithm that enforces l-diversity in terms of preserving data utility. Moreover, it is worth noting that the algorithms in [118, 126, 81], which employ p-sensitive k-anonymity, (a, k)-anonymity, or tuple

Algorithm	Privacy model	Transformation	Approach	Heuristic strategy
Greedy [130]	(h, k, p)-coherence	suppression	protection constrained	greedy search
SuppressControl [16]	ρ-uncertainty	suppression	protection constrained	greedy search
TDControl [16]	ρ-uncertainty	generalization and suppression	protection constrained	top-down space partitioning
RBAT [79]	PS-rule based anonymity	generalization	protection constrained	top-down space partitioning
Tree-based [80]	PS-rule based anonymity	generalization	protection constrained	top-down space partitioing
Sample-based [80]	PS-rule based anonymity	generalization	protection constrained	top-down and bottom-up space partitioning

Table 8: Algorithms for preventing attribute disclosure based on diagnosis codes

diversity, are applied to both quasi-identifiers and sensitive attributes and provide protection from identity and attribute disclosure together. On the other hand, the Anatomize algorithm does not provide protection guarantees against identity disclosure, as all values in quasi-identifiers are released intact.

Algorithms for diagnosis codes. Algorithms for anonymizing diagnosis codes against attribute disclosure are summarized in Table 8. As can be seen, the algorithms adopt different privacy models, namely (h, k, p)-coherence, ρ-uncertainty, or PS-rule based anonymity, and they use suppression, generalization, or a combination of suppression and generalization. Specifically, the authors in [16] propose an algorithm, called TDControl, which applies suppression when generalization alone cannot enforce ρ-uncertainty, and a second algorithm, called SuppressControl, which only employs suppression. Through experiments, they demonstrate that combining suppression with generalization is beneficial for both data utility preservation and efficiency.

Another algorithm that uses suppression only is the Greedy algorithm, which was proposed by Xu et al. [130] to enforce (h, k, p)-coherence. This algorithm discovers all unprotected combinations of diagnosis codes with minimal size and

protects each identified combination, by iteratively suppressing the diagnosis code contained in the greatest number of these combinations. On the other hand, the RBAT [79], Tree-based [80], and Sample-based [80] algorithms employ generalization alone. All algorithms follow the protection-constrained approach, as they minimize information loss no more than necessary to prevent attribute disclosure.

In terms of heuristic search strategies, the algorithms in Table 8 employ a greedy search and operate on either the space of diagnosis codes, or on the transactions of the dataset to be published. Specifically, UGACLIP [76] partitions the space of diagnosis codes in a bottom-up manner, whereas Apriori [115] and mHgHs [74] employ top-down partitioning strategies. The strategy of UGACLIP considers a significantly larger number of ways to generalize diagnosis codes than that of Apriori, which allows better data utility preservation. In addition, the space clustering strategies of CBA and UAR are more effective than the bottom-up space partitioning strategy of UGACLIP, but they are more computationally demanding.

Moreover, all algorithms in Table 8 operate on the space of diagnosis codes and either perform greedy search to discover diagnosis codes that can be suppressed with low data utility loss, or they employ space partitioning strategies. Specifically, TDControl, RBAT, and the Tree-based algorithm all employ top-down partitioning, while Sample-based uses both top-down and bottom-up partitioning strategies. The main difference between the strategy of TDControl and that of RBAT is that the former is based on a taxonomy, which is used to organize diagnosis codes. This restricts the possible ways of partitioning diagnosis codes to those that can be expressed as *cuts* in the taxonomy[2], whereas the strategy of

[2]A cut is a set of generalized diagnosis codes that correspond to nodes of the taxonomy and replace (map) one or more diagnosis codes in the original dataset. Furthermore, the mapping

RBAT partitions the space in a more flexible way as it does not employ this restriction. This helps the preservation of data utility, as it allows exploring more ways to generalize data. The process of partitioning employed by RBAT can be thought of as "growing" a tree, where the nodes correspond to increasingly less generalized diagnosis codes. However, it was shown in [80] that the strategy employed in RBAT might fail to preserve data utility well, as the "growing" of the tree may stop "early". That is, a replacement of diagnosis codes with less general ones, which is beneficial for data utility, is possible but has not been considered by the strategy employed by RBAT.

To address this issue, a different strategy that examines such replacements, when partitioning the space of diagnosis codes, was proposed in [80]. This strategy examines certain branches of the tree that are not examined by the strategy of RBAT and its use allows the Tree-based algorithm to preserve data utility better than RBAT. Moreover, to further increase the number of ways to generalize data, the authors of [80] proposed a way to combine top-down with bottom-up partitioning strategies, by first growing the tree as long as identity disclosure is prevented, and then backtracking (i.e., traversing the tree in a bottom-up way) to ensure that attribute disclosure is guarded against.

5. Relevant techniques

This section provides a discussion of privacy-preserving techniques that are relevant, but not directly related, to those surveyed in this paper. These techniques are applied to different types of medical data, or aim at privately releasing aggre-

must be such that each diagnosis code in the original dataset is mapped to exactly one of these generalized codes.

gate information about the data.

5.1. Privacy-preserving sharing of genomic and text data

While many works investigate threats related to the publishing of demographics and diagnosis codes, there have been considerable efforts by the computer science and health informatics communities to preserve the privacy of other types of data, such as genomic and text. In the following, we briefly discuss techniques that have been proposed for the protection of each of the latter types of data.

5.1.1. Genomic privacy

It is worth noting that a patient's record may be distinguishable with respect to genomic data. Lin et al. [88], for example, estimated that an individual is unique with respect to a small number (approximately 100) of Single Nucleotide Polymorphisms (SNPs), i.e., DNA sequence variations occurring when a single nucleotide in the genome differs between paired chromosomes in an individual. In addition, the release of aggregate genomic information may threaten privacy, as genomic sequences contain sensitive information, such as the ancestral origin of an individual [108], and genetic information about the individual's family members [17]). For instance, Homer et al. [54] showed that such information may allow an attacker to infer whether an individual belongs to the case or control group of GWAS data (i.e., if the individual is diagnosed with a GWAS-related disease or not), while Wang et al. [123] presented two attacks; one that can statistically determine the presence of an individual in the case group, based upon a measure of the correlation between alleles, and another that allows the inference of the SNP sequences of many individuals that are present in the GWAS data, based on correlations between SNPs.

To protect the privacy of genomic information, there are several techniques that are based on cryptography (e.g., see [8] and the references therein) or on perturbation (e.g., see [37]). For instance, Wang et al. [124] proposed cryptographic techniques for the computation of edit distance on genomic data, while Baldi et al. [8] considered different operations, including paternity and genetic compatibility tests. On the other hand, Fienberg et al. [37] examined how to release aggregate statistics for GWAS while satisfying differential privacy through perturbation. In particular, the authors of [37] proposed two methods; one that focuses on the publication of the χ^2 statistic and p-values and works by adding Laplace noise to the original statistics, and a second method that allows releasing noisy versions of these statistics for the most relevant SNPs.

5.1.2. Text de-identification

A considerable amount of information about patients is contained in textual data, such as clinical notes, SOAP (Subjective, Objective, Assessment, Patient care plan) notes, radiology and pathology reports, and discharge summaries. Text data contain much confidential information about a patient, including their name, medical record identifier, and social security number, which must be protected before data release. This involves two steps: (i) detecting direct identifiers and (ii) transforming the detected identifiers, in a way that preserves the integrity of medical information. The latter step is called *de-identification*. In the following, we briefly discuss some techniques that have been proposed for both detecting and transforming direct identifiers. We refer the reader to the survey by Meystre et al. [94], for an extensive review.

Techniques for discovering direct identifiers are based on: (i) *Named Entity Recognition* (NER), (ii) *Grammars* (or *Rules*), and (iii) *Statistical learning*. NER-

based techniques work by locating direct identifiers in text and then classifying them into pre-defined categories. For instance, the atomic elements `Tom Green` and `6152541261` in a clinical note would be classified into the category for `Name` and `Phone Number`, respectively.

The second type of techniques use hand-coded rules and dictionaries to identify direct identifiers, or regular expressions for identifiers that follow certain syntactic patterns (e.g., a phone number must start with a valid area code), while the the last type of techniques are typically based on *classification*. That is, they aim at classifying the terms of previously unseen elements, contained in *test data*, as direct identifiers or as non-identifiers, based on knowledge of *training data*.

The main advantage of NER and grammar-based approaches is that they need little or no training data, and can be easily modified (e.g., by adding a new regular expression). However, their configuration typically requires significant domain expertise (e.g., to specify rules) and, in many cases, knowledge of the specific dataset (e.g., naming conventions). On the other hand, techniques that are based on statistical learning can "learn" the characteristics of data, using different methods, such as Support Vector Machines (SVM) [14] or Conditional Random Fields (CRF) [61]. However, they are limited in that they typically require large training datasets, such as manually annotated text data with pre-labeled identifiers, whose construction is challenging [13].

After the discovery of direct identifiers, there are several transformation strategies that can be applied to them. These include the replacement of direct identifiers with fake, but realistic-looking, elements [111, 12, 50], suppression [18], and generalization [59]. Most of the works on protecting text data aim at transforming direct identifiers without offering specific privacy guarantees. On the contrary, the

works of Chakaravarthy et al. [18] and Jiang et al. [59] offer such guarantees by employing privacy models. The first of these works proposes the K-safety model, which prevents the matching of documents to entities, based on terms that co-occur in a document. This is achieved by lower-bounding the number of entities that these terms correspond to. The work of Jiang et al. [59] proposes a different privacy model, called t-plausibility, which, given word ontologies and a threshold t, requires the sanitized text to be associated with at least t plausible texts, any of which could be the original text.

5.1.3. Aggregate information release

There are certain applications in which data recipients are interested in learning aggregate information from the data, instead of detailed information about individual records. Such aggregate information can range from simple statistics that are directly computed from the data to complex patterns that are discovered through the application of data mining techniques. The interest for supporting these applications has been fueled by recent advances in the development of *semantic* privacy models. These models dictate that the mechanism chosen for releasing the aggregate information (e.g., in the form of a noisy summary of the data), must satisfy certain properties.

One of the most established semantic models is *differential privacy* [49], which requires the outcome of a calculation to be insensitive to any particular record in the dataset. More formally, given an arbitrary, randomized function K and a subset S of its possible outputs, a dataset D is differentially private if

$$P(K(D) \in S) \leq e^{\epsilon} \cdot P(K(D') \in S) \tag{1}$$

where D' is a dataset that differs from D in only one record, and $P(K(D) \in S)$ (respectively, $P(K(D') \in S)$) is the probability that the result of applying K to D (respectively, D'), is contained in the subset S. For instance, the result of statistical analysis carried out on a differentially private data summary must be insensitive to the insertion (or deletion) of a record in (from) the original dataset from which the summary is produced. This offers privacy, because the inferences an attacker can make about an individual will be approximately independent of whether any individual's record is included in the original dataset or not. On the negative side, the enforcement of differential privacy only allows the release of noisy summary statistics[3], and it does not guarantee the prevention of all attacks. Cormode, for example, showed that an attacker can infer the sensitive value of an individual fairly accurately, by applying a classification algorithm on differentially private data [21].

Differential privacy has led to the development of several other semantic models, which are surveyed in [23]. These models relax the (strong) privacy requirements posed by differential privacy by: (i) introducing an additive factor δ to the right part of Equation (1) [31], or (ii) considering attackers with limited computational resources (i.e., attackers with polynomial time computation bounds) [95]. This offers the advantage of limiting noise addition at the expense of weaker privacy guarantees than those offered by differential privacy.

At the same time, there are algorithms for enforcing differential privacy which are applicable to demographics or diagnosis codes. For example, Mohammed et al. [96] proposed a method to release a noisy summary of a dataset containing

[3]This is similar to knowing the queries posed to a technique for enforcing differential private in the interactive setting and releasing the noisy answers to these queries in the form of a summary.

demographics that aims at preserving classification accuracy, while Chen et al. [20] showed how to release noisy answers to certain count queries involving sets of diagnosis codes. Interestingly, both techniques apply partitioning strategies similar in principle to those used by the TDS algorithm [40] and the Recursive partition [52] algorithm, respectively. In addition, systems that allow the differentially private release of aggregate information from electronic health records are emerging. For instance, SHARE [42] is a recently proposed system for releasing multidimensional histograms and longitudinal patterns.

6. Future research directions

Disseminating person-specific data from electronic health records offers the potential for allowing large-scale, low-cost medical studies, in areas including epidemic detection and post-marketing safety evaluation. At the same time, preserving patient privacy is necessary and, in many cases, this can be achieved based on the techniques presented in this survey. However, there are several directions that warrant further research.

First, it is important to study privacy threats posed when releasing patient data, from both a theoretical and practical perspective. This requires the identification and modeling of privacy attacks, beyond those discussed in the paper, and an evaluation of their feasibility on large cohorts of patient data. In fact, it is currently difficult to automatically detect threats for many types of medical data (e.g., for data containing diagnosis codes, or for genomic data), despite some interesting work [10, 35], on demographics, towards this direction. Furthermore, knowledge of: (i) dependencies between quasi-identifiers and sensitive values (e.g., the fact that male patients are less likely to be diagnosed with breast cancer than female

ones) [72, 27], (ii) quasi-identifier values of particular individuals [114] and/or their family members [19], and (iii) the operations of anonymization algorithms [125], may pose privacy risks. However, none of these threats has been investigated in the context of medical data, and it is not clear whether or not the solutions proposed by the computer science community to tackle them are suitable for use in healthcare settings.

Second, the mining of published data may reveal privacy-intrusive inferences about individuals [47, 48, 51], which cannot be eliminated by applying the privacy models discussed so far. Intuitively, this is because mining reveals knowledge patterns that apply to a large number of individuals, and these patterns are not considered as sensitive by the aforementioned privacy models. Consider, for example, that an insurance company applies classification to the data obtained from a healthcare institution to discover that patients over 40 years old, who live in an area with Zip Code 55413, are likely to be diagnosed with diseases that have a very high hospitalization cost. Based on this (sensitive) knowledge, the insurance company may decide to offer more expensive insurance coverage to these patients. To avoid such inferences, sensitive knowledge patterns need to be identified prior to data publishing and be concealed, so that they cannot be discovered when the data are shared.

Third, the large growth in the complexity and size of medical datasets that are being disseminated poses significant challenges to existing privacy-preserving algorithms. As an example of a complex data type, consider a set of records that contain both demographics and diagnosis codes. Despite the need for analyzing demographics and diagnosis codes together, in the context of medical tasks (e.g., for predictive modeling), preserving the privacy of such datasets is very

challenging. This is because, it is not safe to protect demographics and diagnosis codes independently, using existing techniques (e.g., the Mondrian [66] algorithm for demographics and the UGACLIP [76] algorithm for diagnosis codes), while guarding against this threat and minimizing information loss is computationally infeasible [105]. In addition, the vast majority of existing techniques assume that the dataset to be protected is relatively small, so that it fits into the main memory. However, datasets with sizes of several GBs or even TBs may need to be disseminated in practice. Thus, it would be worthwhile to develop scalable techniques that potentially take advantage of parallel architectures to solve this problem.

Fourth, privacy approaches that apply to complex data sharing scenarios need to be proposed. As an example, consider the case of multiple healthcare providers and data recipients who wish to build a common data repository. Healthcare providers may, for example, contribute different parts of patients' EHR data to the repository, whereas data recipients may be querying these data, to obtain anonymized views (i.e., anonymized parts of one or more datasets in the repository), for different purposes [77]. This scenario presents several interesting challenges. First, data contributed by different providers need to be integrated in an efficient and privacy-preserving way. Second, user queries posed to the repository need to be audited and the anonymized views to be produced, so as to adhere to the imposed privacy requirements. Achieving privacy in this scenario is non-trivial, because malicious users may combine their obtained views to breach privacy, even when each query answer is safe when examined independently of others.

Last but not least, it is important to note that the overall assurance of health data privacy requires appropriate policy, in addition to technical means that are exceedingly important.

7. Conclusions

In this work, we presented a systematic review of privacy algorithms that have been proposed for publishing structured patient data. We reviewed more than 45 popular privacy algorithms, derived insights on their operation, and highlighted their advantages and disadvantages. Subsequently, we provided a discussion of some promising directions for future research in this area.

References

[1] EU Data Protection Directive 95/46/ECK, 1995.

[2] UK Data Protection Act, 1998.

[3] Personal Information Protection and Electronic Documents Act, 2000.

[4] N.R. Adam and J.C. Worthmann. Security-control methods for statistical databases: a comparative study. *ACM Comput. Surv.*, 21(4):515–556, 1989.

[5] C. C. Aggarwal and P. S. Yu. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, 2008.

[6] G. Aggarwal, F. Kenthapadi, K. Motwani, R. Panigrahy, and D. Thomasand A. Zhu. Approximation algorithms for k-anonymity. *Journal of Privacy Technology*, 2005.

[7] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB*, pages 487–499, 1994.

[8] P. Baldi, R. Baronio, E. De Cristofaro, P. Gasti, and G. Tsudik. Countering gattaca: efficient and secure testing of fully-sequenced human genomes. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 691–702, 2011.

[9] R.J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *21st ICDE*, pages 217–228, 2005.

[10] K. Benitez, G. Loukides, and B. Malin. Beyond safe harbor: automatic discovery of health information de-identification policy alternatives. In *ACM International Health Informatics Symposium*, pages 163–172, 2010.

[11] S. Berchtold, D. A. Keim, and H. Kriegel. The x-tree : An index structure for high-dimensional data. In *VLDB*, pages 28–39, 1996.

[12] J.J. Berman. Concept-match medical data scrubbing. *Archives of Pathology and Laboratory Medicine*, 127(6):680–686, 2003.

[13] V. Bhagwan, T. Grandison, and C. Maltzahn. Recommendation-based de-identification: A practical systems approach towards de-identification of unstructured text in healthcare. In *Proceedings of the 2012 IEEE Eighth World Congress on Services*, SERVICES '12, pages 155–162, 2012.

[14] C.J.C. Burges. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.*, 2(2):121–167, 1998.

[15] J. Byun, A. Kamra, E. Bertino, and N. Li. Efficient k-anonymization using clustering techniques. In *DASFAA*, pages 188–200, 2007.

[16] J. Cao, P. Karras, C. Raïssi, and K. Tan. rho-uncertainty: Inference-proof transaction anonymization. *PVLDB*, 3(1):1033–1044, 2010.

[17] C.A. Cassa, B. Schmidt, I.S. Kohane, and K. D. Mandl. My sister's keeper?: Genomic research and the identifiability of siblings. *BMC Medical Genomics*, 1:32, 2008.

[18] V.T. Chakaravarthy, H. Gupta, P. Roy, and M.K. Mohania. Efficient techniques for document sanitization. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 843–852, 2008.

[19] B. Chen, R. Ramakrishnan, and K. LeFevre. Privacy skyline: Privacy with multidimensional adversarial knowledge. In *VLDB*, pages 770–781, 2007.

[20] 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, 2011.

[21] G. Cormode. Personal privacy vs population privacy: learning to attack anonymization. In *KDD*, pages 1253–1261, 2011.

[22] B.B. Dean, J. Lam, J.L. Natoli, Q. Butler, D. Aguilar, and R.J. Nordyke. Use of electronic medical records for health outcomes research: A literature review. *Medical Care Reseach and Review*, 66(6):611?638, 2010.

[23] S. De Capitani di Vimercati, S. Foresti, G. Livraga, and P. Samarati. Data privacy: Definitions and techniques. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 20(6):793–818, 2012.

[24] J. Domingo-Ferrer and J. M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. *IEEE Trans. on Knowledge and Data Engineering*, 14(1):189–201, 2002.

[25] J. Domingo-Ferrer and V. Torra. Ordinal, continuous and heterogeneous k-anonymity through microaggregation. *DMKD*, 11(2):195–212, 2005.

[26] J. Domingo-Ferrer, A. Martínez-Ballesté, J. Mateo-Sanz, and F. Sebé. Efficient multivariate data-oriented microaggregation. *VLDB J.*, 15(4):355–369, 2006.

[27] W. Du, Z. Teng, and Z. Zhu. Privacy-maxent: integrating background knowledge in privacy quantification. In *SIGMOD*, pages 459–472, 2008.

[28] Y. Du, T. Xia, Y. Tao, D. Zhang, and F. Zhu. On multidimensional k-anonymity with local recoding generalization. In *ICDE '07*, pages 1422–1424, 2007.

[29] C. Dwork. Differential privacy. In *ICALP*, pages 1–12, 2006.

[30] C. Dwork. Differential privacy: A survey of results. In *TAMC*, pages 1–19, 2008.

[31] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: privacy via distributed noise generation. In *Proceedings of the 24th annual international conference on The Theory and Applications of Cryptographic Techniques*, EUROCRYPT'06, pages 486–503, 2006.

[32] K. El Emam, F. Dankar, R. Issa, E. Jonker et al. A globally optimal

k-anonymity method for the de-identification of health data. *Journal of the American Medical Informatics Association*, 16(5):670–682, 2009.

[33] K. El Emam, E. Jonker, L. Arbuckle, and B. Malin. A systematic review of re-identification attacks on health data. *PLoS ONE*, 6(12):e28071, 12 2011.

[34] K. El Emam and F. K. Dankar. Protecting privacy using k-anonymity. *Journal of the American Medical Informatics Association*, 15(5):627–637, 2008.

[35] K. El Emam, D. Buckeridge, R. Tamblyn, A. Neisa, E. Jonker, and A. Verma. The re-identification risk of canadians from longitudinal demographics. *BMC Medical Informatics and Decision Making*, 11(46), 2011.

[36] J. Fernandez-Aleman, I. Senor, P.A. Oliver Lozoya, and A. Toval. Security and privacy in electronic health records: A systematic literature review. *Journal of Biomedical Informatics*, 46(3):541 – 562, 2013.

[37] S.E. Fienberg, A. Slavkovic, and C. Uhler. Privacy preserving gwas data sharing. In *IEEE ICDM Worksops*, pages 628–635, 2011.

[38] Y.V. Filho. Optimal choice of discriminators in a balanced k-d binary search tree. *Information Processing Letters*, 13(2):67–70, 1981.

[39] B.C.M. Fung, K. Wang, R. Chen, and P.S. Yu. Privacy-preserving data publishing: A survey on recent developments. *ACM Comput. Surv.*, 42, 2010.

[40] B.C.M. Fung, K.Wang, and P.S. Yu. Top-down specialization for information and privacy preservation. In *ICDE*, pages 205–216, 2005.

[41] B.C.M. Fung, K. Wang, L. Wang, and P.C.K. Hung. Privacy-preserving data publishing for cluster analysis. *Data Knowledge Engineering*, 68(6):552–575, 2009.

[42] J.J. Gardner, L. Xiong, Y. Xiao, J. Gao, A.R. Post, X. Jiang, and L. Ohno-Machado. Share: system design and case studies for statistical health information release. *JAMIA*, 20(1):109–116, 2013.

[43] M. Gertz and S. Jajodia, editors. *Handbook of Database Security - Applications and Trends*. Springer, 2008.

[44] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. Fast data anonymization with low information loss. In *Proceedings of the 33rd international conference on Very large data bases*, VLDB '07, pages 758–769, 2007.

[45] A. Gionis, A. Mazza, and T. Tassa. k-anonymization revisited. In *ICDE*, pages 744–753, 2008.

[46] A. Gkoulalas-Divanis and G. Loukides. PCTA: Privacy-constrained Clustering-based Transaction Data Anonymization. In *EDBT PAIS*, page 5, 2011.

[47] A. Gkoulalas-Divanis and G. Loukides. Revisiting sequential pattern hiding to enhance utility. In *KDD*, pages 1316–1324, 2011.

[48] A. Gkoulalas-Divanis and V. S. Verykios. Hiding sensitive knowledge without side effects. *Knowledge and Information Systems*, 20(3):263–299, 2009.

[49] S. Guha, R. Rastogi, and K. Shim. Cure: an efficient clustering algorithm for large databases. In *SIGMOD*, pages 73–84, 1998.

[50] D. Gupta, M. Saul, and J. Gilbertson. Evaluation of a deidentification (De-Id) software engine to share pathology reports and clinical documents for research. *American Journal of Clinical Pathology*, 121(2):176–186, 2004.

[51] R. Gwadera, A. Gkoulalas-Divanis, and G. Loukides. Permutation-based Sequential Pattern Hiding. In *IEEE International Conference on Data Mining (ICDM)*, pages 241–250, 2013.

[52] Y. He and J. F. Naughton. Anonymization of set-valued data via top-down, local generalization. *PVLDB*, 2(1):934–945, 2009.

[53] S. Hettich and C.J. Merz. Uci repository of machine learning databases. 1998.

[54] N. Homer, S. Szelinger, M. Redman, et al. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS Genetics*, 4(8):e1000167, 2008.

[55] B. Hore, R.C. Jammalamadaka, and S. Mehrotra. Flexible anonymization for privacy preserving data publishing: A systematic search based approach. In *SDM*, 2007.

[56] C.J. Hsiao and E. Hing. Use and characteristics of electronic health record systems among office-based physician practices: United states, 2001-2012. *NCHS data brief*, pages 1–8, 2012.

[57] T. Iwuchukwu and J. F. Naughton. K-anonymization as spatial indexing: Toward scalable and incremental anonymization. In *VLDB*, pages 746–757, 2007.

[58] V. S. Iyengar. Transforming data to satisfy privacy constraints. In *KDD*, pages 279–288, 2002.

[59] W. Jiang, M. Murugesan, C. Clifton, and L. Si. t-plausibility: Semantic preserving text sanitization. In *Computational Science and Engineering, 2009. CSE '09. International Conference on*, volume 3, pages 68–75, 2009.

[60] N. Koudas, Q. Zhang, D. Srivastava, and T. Yu. Aggregate query answering on anonymized tables. In *ICDE '07*, pages 116–125, 2007.

[61] J.D. Lafferty, A. McCallum, and F.C.N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, 2001.

[62] M. Laszlo and S. Mukherjee. Minimum spanning tree partitioning algorithm for microaggregation. *Knowledge and Data Engineering, IEEE Transactions on*, 17(7):902–911, 2005.

[63] E.C. Lau, F.S. Mowat, M.A. Kelsh, J.C. Legg, N.M. Engel-Nitz, H.N. Watson, H.L. Collins, R.J. Nordyke, and J.L. Whyte. Use of electronic

medical records (EMR) for oncology outcomes research: assessing the comparability of EMR information to patient registry and health claims data. *Clinical Epidemiology*, 3(1):259–272, 2011.

[64] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Workload-aware anonymization techniques for large-scale datasets. *TODS*, 33(3), 2008.

[65] K. LeFevre, D.J. DeWitt, and R. Ramakrishnan. Incognito: efficient full-domain k-anonymity. In *SIGMOD*, pages 49–60, 2005.

[66] K. LeFevre, D.J. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. In *ICDE*, page 25, 2006.

[67] K. LeFevre, D.J. DeWitt, and R. Ramakrishnan. Workload-aware anonymization. In *KDD*, pages 277–286, 2006.

[68] J. Li, R. Wong, A. Fu, and J. Pei. Achieving -anonymity by clustering in attribute hierarchical structures. In *DaWaK*, pages 405–416, 2006.

[69] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *ICDE*, pages 106–115, 2007.

[70] N. Li, T. Li, and S. Venkatasubramanian. Closeness: A new privacy measure for data publishing. *Knowledge and Data Engineering, IEEE Transactions on*, 22(7):943–956, 2010.

[71] N. Li and M.V. Tripunitara. Security analysis in role-based access control. *ACM Trans. Inf. Syst. Secur.*, 9(4):391–420, 2006.

[72] T. Li and N. Li. Injector: Mining background knowledge for data anonymization. In *ICDE*, pages 446–455, 2008.

[73] Y. Lindell and B. Pinkas. Privacy preserving data mining. page 36?54, 2000.

[74] J. Liu and K. Wang. Anonymizing transaction data by integrating suppression and generalization. In *Proceedings of the 14th Pasific-Asia conference on Advances in Knowledge Discovery and Data Mining*, PAKDD '10, pages 171–180, 2010.

[75] G. Loukides, J.C. Denny, and B. Malin. The disclosure of diagnosis codes can breach research participants' privacy. *Journal of the American Medical Informatics Association*, 17:322–327, 2010.

[76] G. Loukides, A. Gkoulalas-Divanis, and B. Malin. Anonymization of electronic medical records for validating genome-wide association studies. *Proceedings of the National Academy of Sciences*, 17(107):7898–7903, 2010.

[77] G. Loukides, A. Gkoulalas-Divanis, and B. Malin. *An Integrative Framework for Anonymizing Clinical and Genomic Data*, chapter 8, pages 65–89. Database Technology for Life Sciences and Medicine. World Scientific, 2010.

[78] G. Loukides, A. Gkoulalas-Divanis, and B. Malin. COAT: Constraint-based anonymization of transactions. *Knowledge and Information Systems*, 28(2):251–282, 2011.

[79] G. Loukides, A. Gkoulalas-Divanis, and J. Shao. Anonymizing transaction data to eliminate sensitive inferences. In *DEXA*, pages 400–415, 2010.

[80] G. Loukides, A. Gkoulalas-Divanis, and J. Shao. Efficient and flexible anonymization of transaction data. *Knowledge and Information Systems*, 36(1):153–210, 2013.

[81] G. Loukides and J. Shao. Capturing data usefulness and privacy protection in k-anonymisation. In *SAC*, pages 370–374, 2007.

[82] G. Loukides and J. Shao. Clustering-based k-anonymisation algorithms. In *DEXA*, pages 761–771, 2007.

[83] G. Loukides and J. Shao. An efficient clustering algorithm for -anonymisation. *J. Comput. Sci. Technol.*, 23(2):188–202, 2008.

[84] G. Loukides and J. Shao. Preventing range disclosure in k-anonymised data. *Expert Systems with Applications*, 38(4):4559–4574, 2011.

[85] G. Loukides, A. Tziatzios, and J. Shao. Towards preference-constrained -anonymisation. In *DASFAA International Workshop on Privacy-Preserving Data Analysis (PPDA)*, pages 231–245, 2009.

[86] G. Loukides and A. Gkoulalas-Divanis. Utility-preserving transaction data anonymization with low information loss. *Expert Systems with Applications*, 39(10):9764 – 9777, 2012.

[87] G. Loukides and A. Gkoulalas-Divanis. Utility-aware anonymization of diagnosis codes. *IEEE J. Biomedical and Health Informatics*, 17(1):60–70, 2013.

[88] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *ICDE*, page 24, 2006.

[89] M.D. Mailman, M. Feolo, Y. Jin, M. Kimura, K. Tryka, R. Bagoutdinov, et al. The ncbi dbgap database of genotypes and phenotypes. *Nature Genetics*, 39:1181–1186, 2007.

[90] G. Makoul, R. H. Curry, and P. C. Tang. The use of electronic medical records communication patterns in outpatient encounters. *Journal of the American Medical Informatics Association*, 8(6):610–615, 2001.

[91] B. Malin, D. Karp, and R.H. Scheuermann. Technical and policy approaches to balancing patient privacy and data sharing in clinical and translational research. *Journal of Investigative Medicine: the official publication of the American Federation for Clinical Research*, 58(1):11–18, 2010.

[92] B. Malin, G. Loukides, K. Benitez, and E. W. Clayton. Identifiability in biobanks: models, measures, and mitigation strategies. *Human Genetics*, 130(3):383–392, 2011.

[93] B. Malin, G. Loukides, K. Benitez, and E.W. Clayton. Identifiability in biobanks: models, measures, and mitigation strategies. *Human Genetics*, 130(3):383–392, 2011.

[94] S.M. Meystre, F.J. Friedlin, B.R. South, S. Shen, and M. H. Samore. Automatic de-identification of textual documents in the electronic health record: a review of recent research. *BMC Medical Rsearch Methodology*, 10(70), 2010.

[95] I. Mironov, O. Pandey, O. Reingold, and S. Vadhan. Computational differential privacy. In *Proceedings of the 29th Annual International*

Cryptology Conference on Advances in Cryptology, CRYPTO '09, pages 126–142, 2009.

[96] N. Mohammed, R. Chen, B.C.M. Fung, and P.S. Yu. Differentially private data release for data mining. In *KDD*, pages 493–501, 2011.

[97] B. Moon, H. v. Jagadish, C. Faloutsos, and J.H. Saltz. Analysis of the clustering properties of the hilbert space-filling curve. *IEEE Trans. on Knowl. and Data Eng.*, 13(1):124–141, 2001.

[98] M.Ruffolo, F. Angiulli, and C. Pizzuti. Descry: A density based clustering algorithm for very large dataset. In *5th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'04)*, pages 25–27, 2004.

[99] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *IEEE S&P*, pages 111–125, 2008.

[100] M. E. Nergiz, M. Atzori, and C. Clifton. Hiding the presence of individuals from shared databases. In *SIGMOD '07*, pages 665–676, 2007.

[101] M. E. Nergiz and C. Clifton. d-presence without complete world knowledge. *TKDE*, 22(6):868–883, 2010.

[102] M.E. Nergiz and C. Clifton. Thoughts on k-anonymization. *DKE*, 63(3):622–645, 2007.

[103] M.E. Nergiz and C. Clifton. δ-presence without complete world knowledge. *Knowledge and Data Engineering, IEEE Transactions on*, 22(6):868–883, 2010.

[104] W.E.R. Ollier, T. Sprosen, and T. Peakman. UK biobank: from concept to reality. *Pharmacogenomics*, 6(6):639–646, 2005.

[105] G. Poulis, G. Loukides, A. Gkoulalas-Divanis, and S. Skiadopoulos. Anonymizing data with relational and transaction attributes. In *Machine Learning and Knowledge Discovery in Databases - European Conference (ECML/PKDD) (3)*, pages 353–369, 2013.

[106] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical recipes in C (2nd ed.): the art of scientific computing*. Cambridge University Press, 1992.

[107] B.Y. Reis, I.S. Kohane, and K.D. Mandl. Longitudinal histories as predictors of future diagnoses of domestic abuse: modelling study. *BMJ*, 339(9), 2009.

[108] M. Rothstein and P. Epps. Ethical and legal implications of pharmacogenomics. *Nature Review Genetics*, 2:228–231, 2001.

[109] P. Samarati. Protecting respondents identities in microdata release. *TKDE*, 13(9):1010–1027, 2001.

[110] R.S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. *IEEE Computer*, 29(2):38–47, 1996.

[111] L. Sweeney. Replacing personally-identifying information in medical records, the scrub system. In *Proceedings of the AMIA Annual Fall Symposium*, pages 333–337, 1996.

[112] L. Sweeney. k-anonymity: a model for protecting privacy. *IJUFKS*, 10:557–570, 2002.

[113] L. Sweeney. *Computational disclosure control: a primer on data privacy protection*. PhD thesis, 2001. AAI0803469.

[114] Y. Tao, X. Xiao, J. Li, and D. Zhang. On anti-corruption privacy preserving publication. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, ICDE '08, pages 725–734, 2008.

[115] M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving anonymization of set-valued data. *PVLDB*, 1(1):115–125, 2008.

[116] M. Terrovitis, N. Mamoulis, and P. Kalnis. Local and global recoding methods for anonymizing set-valued data. *VLDB J*, 20(1):83–106, 2011.

[117] M. J. Tildesley, T. A. House, M.C. Bruhn, R.J. Curry, M. ONeil, J.L.E. Allpress, G. Smith, and M.J. Keeling. Impact of spatial clustering on disease transmission and optimal control. *Proceedings of the National Academy of Sciences*, 107(3):1041–1046, 2010.

[118] T.M. Truta and B. Vinay. Privacy protection: p-sensitive k-anonymity property. In *ICDE Workshops*, page 94, 2006.

[119] United States Census American Community Survey. 2003 Public Use Microdata.

[120] U.S. Department of Health and Human Services Office for Civil Rights. HIPAA administrative simplification regulation text, 2006.

[121] J. Vaidya and C. Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *KDD*, pages 206–215, 2003.

[122] C.J. Van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979.

[123] R. Wang, Y. F. Li, X. Wang, H. Tang, and X. Zhou. Learning your identity and disease from research papers: information leaks in genome wide association study. In *CCS*, pages 534–544, 2009.

[124] R. Wang, X. Wang, Z. Li, H. Tang, M. K. Reiter, and Z. Dong. Privacy-preserving genomic computation through program specialization. In *Proceedings of the 16th ACM conference on Computer and communications security*, CCS '09, pages 338–347, 2009.

[125] R.C. Wong, A. Fu, K. Wang, and J. Pei. Minimality attack in privacy preserving data publishing. In *VLDB*, pages 543–554, 2007.

[126] R.C. Wong, J. Li, A. Fu, and K.Wang. alpha-k-anonymity: An enhanced k-anonymity model for privacy-preserving data publishing. In *KDD*, pages 754–759, 2006.

[127] X. Xiao and Y. Tao. Anatomy: simple and effective privacy preservation. In *VLDB*, pages 139–150, 2006.

[128] X. Xiao and Y. Tao. Personalized privacy preservation. In *SIGMOD*, pages 229–240, 2006.

[129] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W-C. Fu. Utility-based anonymization using local recoding. In *KDD*, pages 785–790, 2006.

[130] Y. Xu, K. Wang, A. W-C. Fu, and P. S. Yu. Anonymizing transaction databases for publication. In *KDD*, pages 767–775, 2008.

[131] R. Zhang, P. Kalnis, B. Ooi, and K. Tan. Generalized multidimensional data mapping and query processing. *ACM Trans. Database Syst.*, 30(3):661–697, 2005.