Computer Science Review 41 (2021) 100403

Contents lists available at ScienceDirect

Computer Science Review

journal homepage: www.elsevier.com/locate/cosrev

Review article

A survey of privacy-preserving mechanisms for heterogeneous data types

Mariana Cunha [a,*], Ricardo Mendes [b], João P. Vilela [a]

[a] CRACS/INESCTEC, CISUC, and Department of Computer Science, Faculty of Sciences, University of Porto, Porto, Portugal
[b] CISUC, Department of Informatics Engineering, University of Coimbra, Coimbra, Portugal

ARTICLE INFO

Article history:
Received 11 December 2020
Received in revised form 17 March 2021
Accepted 3 May 2021
Available online xxxx

Keywords:
Privacy
Privacy taxonomy
Privacy-preserving mechanisms
Heterogeneous data types
Privacy tools

ABSTRACT

Due to the pervasiveness of always connected devices, large amounts of heterogeneous data are continuously being collected. Beyond the benefits that accrue for the users, there are private and sensitive information that is exposed. Therefore, Privacy-Preserving Mechanisms (PPMs) are crucial to protect users' privacy. In this paper, we perform a thorough study of the state of the art on the following topics: heterogeneous data types, PPMs, and tools for privacy protection. Building from the achieved knowledge, we propose a privacy taxonomy that establishes a relation between different types of data and suitable PPMs for the characteristics of those data types. Moreover, we perform a systematic analysis of solutions for privacy protection, by presenting and comparing privacy tools. From the performed analysis, we identify open challenges and future directions, namely, in the development of novel PPMs.

Contents

1. Introduction

Data is continuously being collected due to the pervasiveness of always connected devices and the ubiquitousness of Internet of Things (IoT) technologies in people's lives. IoT provides the interconnection between multiple heterogeneous devices and sensors that are able to monitor and gather all types of data about machines and human social life [1]. Despite the benefits that can come from collecting data, users are exposing sensitive and private information with possibly untrustworthy entities. These entities can process, analyze and mine data in order to extract useful information, but also sell and/or share the collected data with third parties, using it maliciously. With the growing number of misuse of data and data breaches [2], privacy has been an emergent topic and serious privacy concerns have been

* Corresponding author.
E-mail addresses: mccunha@dei.uc.pt (M. Cunha), rscmendes@dei.uc.pt (R. Mendes), jvilela@fc.up.pt (J.P. Vilela).

https://doi.org/10.1016/j.cosrev.2021.100403

aroused. To address these issues, numerous Privacy-Preserving Mechanisms (PPMs) and tools have been proposed [3–5].

Although PPMs aim to preserve users' privacy, this can come at the expense of a degraded utility of data [6]. Therefore, the selection of a PPM should take into account not only the users' objective but also the trade-off between the privacy level and the utility of data, which are many times application-specific. Considering the heterogeneity of the collected data, selecting and configuring the proper PPM is quite challenging. To automatize this process and to give a logical and systematic structure of the main components and concepts of privacy, several tools were developed [7–10]. These tools were proposed to facilitate the configuration of PPMs and the analysis of results. However, selecting the proper PPM according to the characteristics of the data remains as a challenge.

To better understand how to identify PPMs according to the data characteristics, this survey presents an up-to-date and thorough review on heterogeneous data types and applicable PPMs. In recent years, several general surveys [3–5] have focused on PPMs for data mining and how they can be compared in terms of achieved privacy level, data utility, complexity, and/or application fields. Other more specific surveys discuss PPMs for a specific data type or a restrict group of data types [11–13], as well application of PPMs for specific domains [5,14]. Our survey differs from previous literature by proposing a privacy taxonomy for heterogeneous data types that establishes a relation between different data types and PPMs. In this survey, PPMs are classified according to the overall categories of data they can be applied to (structured, semi-structured and unstructured), as well as their suitability for real-time or offline application. The main contribution of this survey is the specification of a taxonomy of data types for each category of data that is amenable for the identification of corresponding PPMs, so as to allow the reader to properly understand the underlying principles of the addressed PPMs and their applicability to the data types in the taxonomy within. This survey further contributes by presenting and comparing existing privacy tools with respect to the data types and PPMs made available, as well the privacy and utility evaluation features of such tools.

The remainder of the survey is structured as follows. Section 2 provides a study and classification of heterogeneous data types. Section 3 presents the state-of-the-art PPMs. Section 4 proposes a privacy taxonomy for heterogeneous data types. Section 5 provides an overview of existing tools for privacy protection. Section 6 presents open challenges and future directions. Finally, Section 7 concludes the survey paper.

2. Heterogeneous data types

Everyday, various devices and services collect large amounts of heterogeneous data with different purposes. Although the collection purpose may vary, collected data may have similar characteristics. In the domain of IoT, considerable amounts of data are continuously collected by different sensors. According to [15], the top ten IoT sensors includes: temperature sensors, humidity sensors, pressure sensors, proximity sensors, level sensors, accelerometers, gyroscope, gas sensors, infrared sensors, and optical sensors. From these sensors, several services are provided and different data types are collected. This section gives an overview of existing types of data.

Commonly, data is classified according to its structure, that is, how the data is organized [12,13,16]. From this classification, we have structured data, semi-structured data and unstructured data. Structured data corresponds to data often stored in tables, such as relational databases or spreadsheets. Following the structure imposed by the database, we may have data types such as numbers, strings, booleans, dates, and others. Structured data is divided in categorical data, that is, data types that can be divided into groups, and numerical data, that corresponds to data types represented by numeric values of specific variables [17]. Categorical data is subdivided in nominal, which represents a set of possible values, and ordinal, which also represents a set of values but with a rank order. In its turn, numerical data is subdivided in interval and ratio, which represent variables that can be measured with an interval scale (e.g. Celsius scale) or a ratio scale (e.g. Kelvin temperature scale), respectively. Unstructured data consists in data that does not have a predefined data model or a specified organization. Examples of unstructured data are images, videos, streaming sensor data, and text documents. Within unstructured data, we may also have dates, numbers or facts. Semi-structured data is a type of structured data that does not have a rigid structure imposed by a data model. For example, emails are constituted by structured information (e.g. sender, recipient) and unstructured data that corresponds to the email message content and/or attachments. Semi-structured data are often represented as graphs, XML and other markup languages.

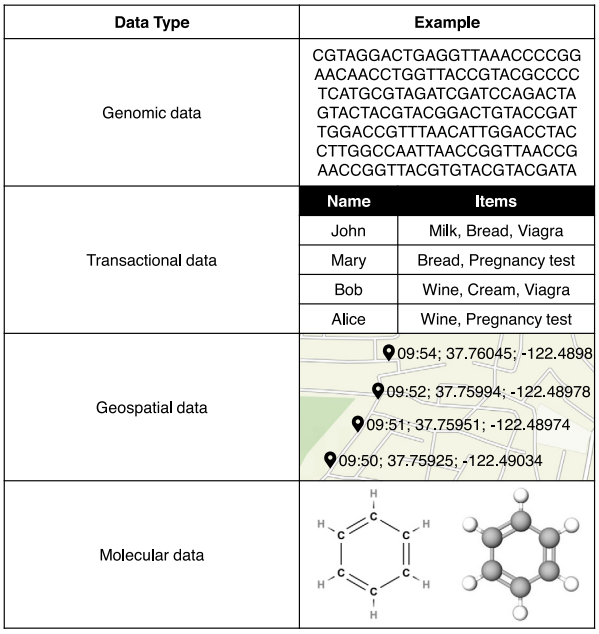

Fig. 1. Examples of data types.

Beyond the aforementioned, unstructured data can be further divided in several categories such as [16]: time series data, streaming data, sequence data, multimedia data, and spatial data. While time series data consists in sequences of values/events repeatedly collected over time (e.g. stock market data), sequence data corresponds to sequences of ordered values/events that are recorded with or without a certain timestamp (e.g. genomic data, see Fig. 1). Streaming data consists in data continuously arriving (e.g. sensor data). Multimedia data includes data such as images, videos or audios. The last category is spatial data that corresponds to space-related data, such as maps. Although the terms spatial and geospatial data are often used as equivalents, geospatial data corresponds to a type of spatial data that is related to Earth and that contains geographic components, such as location coordinates (see Fig. 1). Finally, textual and transactional data can also be unstructured data types, whereby textual data refers to unstructured text (e.g. documents) and transactional data, a canonical example of set-valued data, corresponds to data in which each record contains a set of arbitrary items (e.g. online shopping, see Fig. 1).

Datasets can also be divided in three categories [18]: record data, graph-based data, and ordered data. Record data is usually stored in relational databases or flat files and each record is described with the same set of attributes. Graph-based data is typically used to represent data objects that can be mapped as nodes of a graph, while their relationship is mapped as a link (e.g. social network data and molecules, see Fig. 1). The ordered data category pertains data that is ordered in time or space, such as, sequential data, time series data, or spatial data.

A relevant matter for processing heterogeneous data types is the amount of data to be considered. The integration and analysis of heterogeneous data types is quite challenging, specially, due to the increase of data collection, that results in big data issues. Rob Thomas,[1] general manager for IBM Analytics,[2] defined big data as "diverse datasets that include structured, semi-structured and unstructured data, from different sources and in different volumes, from terabytes to zettabytes. It is about datasets so large and diverse that it is difficult, if not impossible, for traditional relational databases to capture, manage, and process them with low-latency". To deal with the processing, integration, and analysis of heterogeneous data and big data, some methods have been developed and presented in [19].

The focus of this survey is on heterogeneous data types and corresponding PPMs. Big data aspects have been the subject of other surveys, where, for example, a well-defined taxonomy is presented [16] according to six dimensions: data, compute infrastructure, storage infrastructure, analytics, visualization, and security and privacy. In the dimension of data, the authors divided data according to different characteristics, such as the structure of data, as mentioned before. Similarly, the survey [20] presents a rich taxonomy of big data on the following domains: semantic, compute infrastructure, storage system, big data management, data mining and machine learning, and security and privacy. With respect to the semantic of big data, the authors consider diverse characteristics, such as volume, velocity, variety, and others. Within variety, there is a data classification that also divides data according to its structure (i.e. structured, semi-structured, and unstructured data). In the data taxonomy proposed by [13], big data was presented as a category that was likewise divided according to the data structure and included streaming data as a subcategory.

To summarize this section, Fig. 2 presents a data taxonomy according to the structure of data, where data is first divided into structured, semi-structured and unstructured. Within each category, structured data is divided into categorical and numerical data, semi-structured data is divided into graph data, XML, and key-valued data, and unstructured data is divided into textual, multimedia, time series, streaming, sequence, spatial, and transactional data. This data taxonomy will be instrumental so as to identify PPMs suitable to the identified data categories, as we will now address.

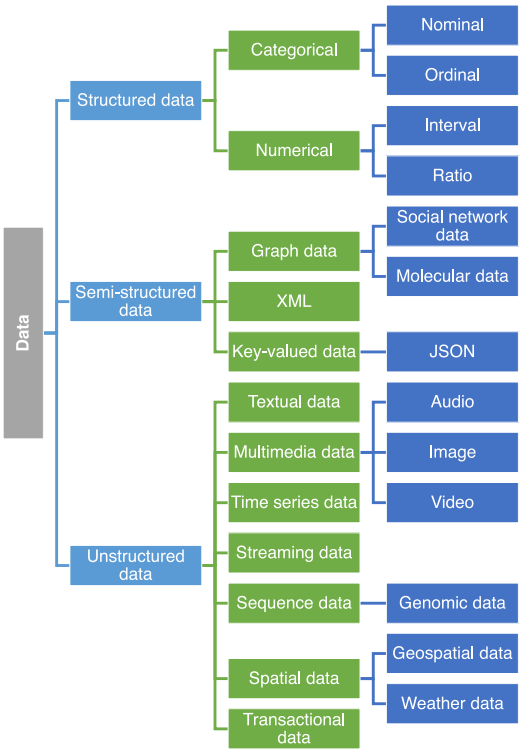

Fig. 2. Data taxonomy according to the structure of data.

3. Privacy-preserving mechanisms

This section gives an overview of existing PPMs over different domains. Before presenting the PPMs, some concepts are briefly presented as background knowledge. PPMs are applied to protect user's sensitive and private information. In general, we consider a sensitive attribute (SA) when we have user-specific private data that can be shared for research/statistical analysis purposes, but should not be linkable to the individual user. A quasi-identifier (QID) consists in a non-sensitive attribute (or a set of attributes) that can be combined or linked with external/background information to re-identify the individual to whom data refers. Finally, a key attribute consists in a explicit/uniquely identifier (ID) of an individual, or in other words, personally identifiable information (PII).

To preserve users' privacy, PPMs often apply one or a combination of data sanitizing operations, such as generalization, suppression, perturbation, anatomization, permutation and/or slicing [5, 13]. The sanitization goal is to protect sensitive information by removing or modifying attributes of data. Generalization corresponds to the replacement of a value with a broader one. For instance, the replacement of numerical data with intervals (e.g. an age of 33 may be specified as the interval [30, 35]), and the definition of a hierarchy for categorical attributes (i.e. generalize specific values of an attribute with a value/category that includes those values). Suppression consists in removing some values of an attribute to prevent the disclosure of information. Typically, this operation is used in tables by removing all values of an attribute in a column or by removing an entry row. Perturbation corresponds to the replacement of the original data with values with identical statistical information. This operation is commonly achieved with the addition of noise. Anatomization (or anatomy) consists in the de-association of quasi-identifiers (QIDs) and sensitive attributes (SAs) in two separated tables in order to prevent the linkage of QIDs to SAs [21]. Permutation corresponds to the rearrangement of values after their partitioning into group of values. Although this operation alone is not suitable for real-world data, it is often combined with slicing [13]. Slicing partitions the data both vertically and horizontally, which makes this technique able to handle high-dimensional data and data without a clear separation between QIDs and SAs [22]. Briefly, vertical partitioning consists in having each attribute or subset of attributes contained in each column and horizontal partitioning consists in randomly permute the values within columns, thus breaking the linkage among different columns. Since the presented list

[1] https://www.robdthomas.com/
[2] https://www.ibm.com/analytics

ID	QID			SA
Name	Age	Sex	Zip Code	Disease
Mary	25	F	46909	Hepatitis C
John	25	M	46909	HIV
Bob	35	M	46900	HIV
Ian	54	M	46900	Gastritis
Alice	57	F	46761	HIV
Helen	60	F	46761	Hepatitis C
Cindy	60	F	46760	Hepatitis C
Alex	63	M	46760	Diabetes

(a) Original table

ID	QID			SA
Name	Age	Sex	Zip Code	Disease
*	[20-55]	*	4690*	Hepatitis C
*	[20-55]	*	4690*	HIV
*	[20-55]	*	4690*	HIV
*	[20-55]	*	4690*	Gastritis
*	[56-65]	*	4676*	HIV
*	[56-65]	*	4676*	Hepatitis C
*	[56-65]	*	4676*	Hepatitis C
*	[56-65]	*	4676*	Diabetes

(b) Anonymized table

Fig. 3. Example of suppression of the 'Name' and 'Sex' attributes and generalization of the 'Age' and 'Zip Code' attributes in a table, where 'Name' is the identifier (ID) attribute, 'Age', 'Zip Code' and 'Sex' are quasi-identifiers (QID), and 'Disease' is the sensitive attribute (SA).

Age	Sex	Zip Code	Disease
25	M	46909	HIV
25	F	46900	HIV
35	M	46909	Hepatitis C
54	M	46900	Gastritis
57	F	46761	Hepatitis C
60	M	46760	Diabetes
60	F	46761	Hepatitis C
63	F	46760	HIV

(a) Sliced table with one attribute per column

(Age, Sex)	(Zip Code, Disease)
(25, F)	(46900, HIV)
(25, M)	(46909, Hepatitis C)
(35, M)	(46900, Gastritis)
(54, M)	(46909, HIV)
(57, F)	(46760, Diabetes)
(60, F)	(46761, HIV)
(60, F)	(46760, Hepatitis C)
(63, M)	(46761, Hepatitis C)

(b) Sliced table with two attributes per column

Fig. 4. Example of slicing with one attribute per column (Fig. 4(a)) and two attributes per column (Fig. 4(b)), where data was partitioned both vertically and horizontally.

of sanitization operations is not an extensive list, please refer to [13,23] for a more thorough analysis.

Fig. 3 presents an example of suppression and generalization operations applied to a table. As shown in the anonymized table, suppression is achieved by removing all the values of the identifying 'Name' attribute and the QID 'Sex' attribute, and generalization is applied to the 'Age' and 'Zip Code' attributes, where numerical values were replaced with broader intervals. Although PPMs aim to preserve user's privacy, this can come at the expense of a degraded utility of data [6]. To measure the utility level, we have metrics such as the utility loss that evaluates the utility cost of applying a PPM. Considering Fig. 3, when we generalize the 'Age' attribute, there is an utility loss of information about the original ages. Thus, instead of possible insights about specific ages, our analysis will be performed on intervals, which may result in different conclusions.

Fig. 4 presents an example of slicing, where data was both vertically and horizontally partitioned. With respect to vertical partitioning, the table of Fig. 4(a) has one attribute per column, while the table of Fig. 4(b) has a set of two attributes per column. Within columns, the data was randomly permuted in both tables, thus preventing the linkage among the columns.

Aside from sanitization, PPMs can also rely on cryptography to preserve the privacy of the data. These mechanisms apply protocols to allow distributed processing, sharing and retrieval of data under privacy guarantees. Therefore, in the following subsections, PPMs are presented according to their methodologies. Anonymization mechanisms sanitize the data in order to protect private and sensitive information. Obfuscation mechanisms return obfuscated reports by perturbing the original data (e.g. adding noise to the original reports). Finally, we present mechanisms that do not apply data sanitization operations, but instead rely on cryptography to protect the data.

3.1. Anonymization mechanisms

The anonymization mechanisms are presented in this section and divided according to the data structure they are suitable for.

3.1.1. Structured data

One of the most known PPMs is k-anonymity that guarantees that in a set of k individuals, the identity of each one cannot be disclosed from at least $k-1$ individuals in the same set [24,25]. The set of k individuals is referred to as equivalence class. Moreover, the achieved privacy level can be measured by the value of k, such that a higher value of k corresponds to a higher privacy level (i.e. it is harder to de-anonymize). k-anonymity and its variants that are presented below (p-sensitive, l-diversity, and t-closeness) were designed for **structured data**, commonly represented in the form of tables. The p-sensitive mechanism [26] satisfies the k-anonymity property and guarantees that within a set of k individuals, for each group of confidential key attributes, the number of distinct values is at least p for each confidential attribute within the same group.

The l-diversity mechanism guarantees k-anonymity and expands it by requiring that each equivalence class is a set of entries such that at least l "well-represented" values exist for the sensitive attributes [27]. Thus, a table is considered conformant with l-diversity when all the equivalence classes of the table are l-diverse. However, l-diversity has a limitation in the assumptions of adversarial knowledge. This mechanism considers that if the distribution of the attribute is known, the adversaries will obtain knowledge on a sensitive attribute, which is a drawback of this approach [28]. To solve the issues created by l-diversity, t-closeness was proposed [29]. This mechanism is based on k-anonymity and l-diversity properties. An equivalence class is considered conformant with t-closeness when the distance between the distribution of a sensitive attribute in the class and the distribution of the attribute in the table is lower than a threshold

t. Thus, a table is in accordance with *t*-closeness when all the equivalence classes satisfy *t*-closeness.

Based on slicing and *l*-diversity, Li et al. proposed a PPM for **structured data** and **transactional data**, named *l*-diverse slicing [22]. This mechanism guarantees that an adversary cannot disclose sensitive information of any individual with a probability greater than 1/*l*. For that, the attributes are partitioned into columns, then the algorithm applies column generalization and partitions tuples into buckets. The highly correlated attributes are in the same column to preserve the correlation between those attributes, while the relations between uncorrelated attributes are broken. Thus, this mechanism prevents the linkage among different columns. The work in [30] proposed a mechanism for **structured data** that is suitable for multiple SAs. This mechanism is based on anatomization and slicing, while guaranteeing the *k*-anonymity and *l*-diversity principles.

On the other hand, building from slicing and *t-closeness*, Wang et al. proposed a mechanism named *t*-closeness slicing [31], whose objective is to better protect **transactional data** against existing attacks, in where an attacker is able to identify the owner of an individual (identity disclosure), infer information about an individual (attribute disclosure), or infer if an individual is in the dataset or not (membership disclosure). Similarly to *l*-diverse slicing, this mechanism uses slicing to partition transactional data both vertically and horizontally. Vertical partitioning groups highly correlated attributes into columns, while horizontal partitioning groups highly correlated transactions into buckets. Lastly, the algorithm randomly swaps pairs of rows to break the correlations among columns, thus protecting against the aforementioned privacy threats.

Differential privacy was introduced in the domain of statistical Databases (DBs) to protect **structured data**. Differential privacy guarantees that any finding obtained from the DB does not reveal the presence or absence of an item in a DB [32,33]. This mechanism aims to minimize the risk of an individual or a record entering in a DB, thereby encouraging the participation in data sharing. In particular, the objective of differential privacy is that a DB reveals low information about a certain individual/record, even if all the information about the others is known. That is, the response to a query to the DB must be indistinguishable, whether the individual/record is in the DB or not, with the goal of making individuals more confident about sharing their data. The most common mechanism of protection consists in adding noise to the data, in order to provide formal guarantees of privacy. For instance, the Laplace mechanism was proposed to protect **numerical data** and the Exponential mechanism was proposed to protect **categorical data**, following the respective Laplace and Exponential distributions [34,35]. In addition to being used in structured data, differential privacy can also be applied in **unstructured data**, such as **set-valued data** [36], **genomic data** [37] and **image data** [38].

While the variant of centralized differential privacy requires users to have trust in a third party (the database owner) that will add noise to the database, in Local Differential Privacy (LDP) the noise is added by the user and, consequently, there is no need to trust in a centralized authority [39]. LDP was proposed due to the necessity of analyzing statistical data from users and inferring statistics about populations with privacy guarantees for individual users [40]. To achieve this, some techniques were proposed by well-known companies. Google proposed the Google RAPPOR [41], which is an open-source privacy technology used in Google Chrome to collect the common URLs, chosen homepages, settings and other web browsing behaviors. Apple uses LDP to collect usage statistics and commonly used emojis, new words added by the users, and to improve their behavior [42]. Microsoft uses LDP to collect the telemetry data [43]. The main difference between differential privacy and LDP is that differential privacy applies constant noise to all individuals in the dataset and LDP applies noise for each report individually (i.e. the dataset contains the aggregated result). Earlier LDP mechanisms have been developed for **numerical** [34,43], **categorical** [41,44], and **set-valued data** [45], whereas recently, mechanisms have been developed for different domains and data applications [46], such as **key–value data** [47] and **multidimensional data** (i.e. both numerical and categorical attributes) [48].

The LDP mechanism recalls the concept of personalized privacy proposed in the context of **structured data** by Xiao and Tao [49], where the users can define their privacy level. The goal of this mechanism is to perform the minimum generalization, while guaranteeing the maximum utility of data and the users' privacy preferences. For that, the algorithm starts by creating a subtree from a generalized taxonomy tree, allowing the users (record owners) to define a guarding node according to their privacy preferences. The guarding node indicates that the user does not want to be publicly associated with any leaf (sensitive value) in the subtree. Therefore, the breach probability is defined as the probability of an adversary to infer any sensitive value from the subtree of the guarding node. Beyond being used in structured data, personalized privacy can also be applied in **semi-structured data** and **unstructured data**, in **social network data** [50] and **geospatial data** [51,52], respectively.

3.1.2. Semi-structured data

Social network data is commonly represented as a graph, where nodes correspond to individuals and edges symbolize the relationships between those individuals. Privacy of **graph data** has received particular interest in research [53–55] due to the amounts of social network data that have been made publicly available. In this context, privacy breaches are divided in three categories [56]: identity disclosure, link disclosure, and content disclosure. Identity disclosure corresponds to the case when a node is revealed and, consequently, the identity of the individual represented by that node. Link disclosure occurs when a sensitive relationship (i.e. link/edge) is disclosed between two individuals (nodes). Finally, content disclosure is related to the privacy breach of the data associated to the nodes. To protect from identity disclosure, Liu and Terzi proposed a systematic framework for anonymization of identity on graphs [56]. From this work, a graph is *k*-degree anonymous if for every node *v*, there are at least *k-1* other nodes in the graph with the same degree as *v*. Thus, this mechanism prevents the re-identification of individuals (nodes) by adversaries with a priori knowledge about the degree of certain nodes. The main objective is to construct a *k*-degree anonymous graph from an input graph by performing the minimum number of graph-modification operations (e.g. edge additions or deletions). From the results, the utility of the anonymized graph and the efficiency of the proposed algorithms were guaranteed. To preserve the privacy of sensitive relationships in graph data, the authors of [57] proposed five different privacy-preserving techniques by varying the amount of data removed and the privacy preserved.

Yang and Li proposed a mechanism to protect sensitive information in **XML data** [58]. Since the existing dependencies in XML data can cause information leakage, the main objective of the proposed algorithm is to find the partial document of a given XML document that should be published to prevent information leakage. For that, the authors formulated the existing dependencies as XML constraints and protected sensitive information from data inference. Landberg et al. provided a new privacy notion *δ-dependency* and developed an extension of anatomy [21] for XML data [59]. The key idea of *δ-dependency* is to deal with hierarchical sensitive data that occurs when data values are taken

from a hierarchical tree structure, where the specificity of data values increases with moving down in the tree. For that, the developed mechanism supports the generalization of sensitive attributes. Furthermore, the algorithm based on the anatomy technique allows the de-association of quasi-identifiers and sensitive attributes in order to prevent data linkage.

3.1.3. Unstructured data

Textual data frequently includes personal text messages or documents. Since the content of the text may contain sensitive and private information, sanitization and/or anonymization of data is necessary to preserve users' privacy. Saygin et al. focused on preserving the privacy of text documents [60]. The proposed solution was divided in two phases: sanitization and anonymization. The first phase corresponds to the automatic identification and protection of sensitive contents of the text by modifying and hiding those private information. In the anonymization phase, the objective is to protect the privacy of the author/owner of the document. For that, a privacy technique based on the *k*-anonymity of authorship is used. In order to automatize the document sanitization, the authors of [61] proposed the ERASE (Efficient RedAction for Securing Entities) framework, which allows dynamic sanitization, whereby sensitive terms are identified and removed from the text, so as to enable distinct users to get different views of the document according to their authorization status. *t*-plausibility [62,63] was proposed for text sanitization, such that the sensitive terms are replaced with more general ones that are semantically related. A desensitized text is obtained by generalizing words without unnecessary degradation of the contained information. This theoretic approach is also used as a measure of quality of sanitized documents, according to the provided heuristics of text sanitization. Therefore, from this work, a sanitized text is *t*-plausible if at least *t* texts (including the original text) can be generalized to the sanitized text.

Transactional data, a canonical example of set-valued data, is generated from multiple sources, which is appreciated from the data mining point of view. However, since it may contain sensitive information, data privacy should be preserved before releasing the data. Xu et al. proposed a privacy notion *(h,k,p)-coherence* for transactional data and a mechanism that achieves coherence by using suppression [64]. The notion of *(h, k, p)-coherence* states that every subset of no more than *p* public items contains at least *k* transactions and no more than *h* percent of these transactions contains a common private item. If coherence is not satisfied by transactional data, the proposed mechanism uses suppression of public items to modify data and, consequently, achieve coherence. For that, the item is deleted from all transactions where the item is contained.

Terrovitis et al. proposed the concept of k^m-anonymity for **set-valued data** [65]. k^m-anonymity is based on *k*-anonymity but has the capacity to deal with data dimensionality. This concept states that for any set of m or less items in the database, there should be at least *k* transactions in the published database that contain the set. If k^m-anonymity is not met by the database, the authors follow a generalization approach, that is, precise items are replaced with more generalized ones. Alternatively, the authors in [66] proposed an algorithm to anonymize set-valued data based on *k*-anonymity considering that any item of the sets could be sensitive. However, while the former work [65] consists in a bottom-up approach and uses k^m-anonymity, the latter approach [66] follows a top-down approach and uses the original *k*-anonymity. In the context of set-valued data, *k*-anonymity states that for any transaction, there are at least *k-1* other identical transactions. Moreover, although both works use generalization, the authors of [66] proposed a top-down local generalization approach to achieve *k*-anonymity. The proposed algorithm is called "partition" and anonymizes set-valued data by recursively partitioning similar set-valued transactions into groups. Therefore, this method is linearly scalable with the input size taking into consideration the information loss. To improve the data quality and reduce the information loss, the authors of [67] proposed an approach that integrates generalization and suppression. While the previous works consider that any item of the sets could be sensitive, Ghinita et al. proposed an approach that takes into account the disclosure of the individuals' identity not only through the items but also what can be inferred from the non-sensitive information [68].

With respect to **streaming data**, on pair with the aroused privacy concerns is the interest on the analysis of continuously data collection. To respond to the privacy issues, several anonymization mechanisms have been proposed as reviewed in [69]. Li et al. proposed the first algorithm based on *k*-anonymity for streaming data: Stream K-anonYmity (SKY) [70]. SKY uses a top-down specialization tree based on the attributes of the arriving tuples (i.e. piece of data of an individual in a stream). When a tuple arrives, the algorithm finds the most specific node in the tree that is able to anonymize the arriving tuple according to its attributes. The node that generalizes the arriving tuple is returned by the algorithm and, consequently, the privacy is preserved. Similarly, the Continuously Anonymizing STreaming data via adaptive cLustEring (CASTLE) mechanism was proposed for streaming data based on *k*-anonymity, however, CASTLE uses a cluster-based approach instead of a tree-based approach and is able to handle *l*-diversity [71]. Both SKY and CASTLE use a threshold/delay constraint to specify how long a tuple can wait before being published. Zhou et al. proposed a mechanism that takes into account the utility of data by considering not only the information loss but also the impact of the delay factor [72]. For that, the authors use the time delay as a factor of preference instead of a simple constraint. In contrast with these works, Kim et al. presented a delay-free anonymization mechanism [73], that does not generate an accumulation delay by immediately anonymizing the input streams with counterfeit values. In the domain of IoT, an unscented Kalman Filter based on differential privacy was proposed to protect user's privacy when sharing streaming data to cloud platform for real-time processing [74].

Al-Hussaeni et al. developed a mechanism for trajectory **streaming data** [75], named Incremental Trajectory Stream Anonymizer (ITSA). This mechanism incrementally anonymizes a sequence of sliding windows that are dynamically updated according to the trajectory stream. Building from the concept of sliding windows, Wang et al. proposed two dynamic algorithms for publishing transactional data streams that continuously anonymize a sliding window with generalization and suppression [76]. In the context of categorical data streams, Zhang et al. proposed a tuple-based anonymization mechanism that implements a two-phase approach [77]. This innovative approach first encodes the users sequences and then anonymizes the categorical information, thus preventing the disclosure of sensitive data. From the results, this approach achieves an efficient performance and low communication overhead. Besides preserving privacy of data streams, several works have been proposed in the context of data stream mining, such the ones in [78,79] that developed mechanisms based on perturbation.

Mix networks is a routing protocol that is used to provide hard-to-trace communications [80]. This protocol consists of using a chain of mix nodes that receive messages from different senders, shuffle the messages, and then send them in a random order to the next destination (likely another mix node). The link between the sender and the receiver is broken, which makes the trace of the end-to-end communication harder for possible eavesdroppers. To prevent the network from malicious mix nodes, each

mix node only has information about the previous node that sends the message and the next destination to send the mixed messages. Following the idea of mixing identities, Beresford and Stajano proposed the concept of mix-zones for privacy of **geospatial data** [81]. The proposed mechanism guarantees the privacy of the users by shuffling their identities when they enter in a mix-zone. As the users do not communicate with any apps within the mix-zone, applications cannot distinguish that user from any other who was in that mix-zone at the same time or even link users that enter in the mix-zone with those that leave it.

3.2. Obfuscation mechanisms

Obfuscation mechanisms are commonly used to protect users' privacy in the domain of **geospatial data**. Due to the characteristics of collecting this type of data, existing mechanisms were developed by considering the dependence or independence of reported locations, that is, *continuous* or *sporadic* scenarios, respectively. Geo-indistinguishability was proposed based on the notion of differential privacy for sporadic scenarios (i.e. considering independence between reports) [82]. This PPM guarantees a level of privacy within a radius, making any disclosed location indistinguishable from any other point within that radius. To achieve a desired privacy level, the mechanism adds random noise to the user's position, thus reporting an obfuscated location. The Planar Laplace (PL) mechanism was the first proposed geo-indistinguishable Location Privacy-Preserving Mechanism (LPPM). This mechanism adds 2-dimensional Laplacian noise centered at the exact user's location following a Laplacian distribution. In order to increase the utility of the data without decreasing the level of privacy, remapping techniques have been proposed [83] for geo-indistinguishability. Currently, the PL mechanism with optimal remapping is considered the state of the art of geo-indistinguishability in sporadic location privacy [84].

Based on the PL mechanism, LPPMs for the continuous scenario have been proposed. The adaptive geo-indistinguishability LPPM explores the effect of the correlation among the user's obfuscated locations [85]. This correlation can be used by an attacker to degrade the privacy of the user [86]. Therefore, the adaptive geo-indistinguishability mechanism applies the PL mechanism and dynamically adapts the privacy parameter ϵ considering the correlation of the previous obfuscated locations. To do so, the adaptive mechanism adjusts the amount of noise required to obfuscate the exact user location, in order to improve the privacy or the utility level. The obtained results show that the adaptive mechanism achieves better performance by adjusting the noise added according to the correlation of previous obfuscated locations.

Clustering geo-indistinguishability mechanism was proposed for both the sporadic and the continuous scenarios [87]. The clustering geo-indistinguishability creates obfuscation clusters to aggregate nearby locations, reporting the same obfuscated location for those points. To obfuscate the exact user locations, this mechanism uses the PL mechanism, which is considered the state-of-the-art mechanism for the sporadic scenarios. In particular, the authors explain how the proposed mechanism deals with the main issues of the geo-indistinguishability and with the frequency of reports. The assessment of the clustering geo-indistinguishability is performed in comparison with the PL mechanism and the adaptive geo-indistinguishability. The obtained results show that clustering geo-indistinguishability achieves a better trade-off between privacy and utility than the PL mechanism and the adaptive mechanism, by improving the privacy level with little to no loss of utility.

In addition to the aforementioned mechanisms and their applications, obfuscation mechanisms have been developed for **multimedia data**. In particular, **audio data** is continuously collected by microphones embedded in IoT devices, such as voice assistants that are designed to detect and to respond to voice commands. To protect user's privacy, the research community have been studied how to reduce speech intelligibility [88–90]. Chen et al. proposed an automatic method for reducing the speech intelligibility while preserving non-speech environmental sounds [90]. The intelligibility of the speech is related to the vowels and to the consonants, such that vowel-only sentences are more intelligible than consonant ones [88,89]. Based on this characteristic, the proposed method obfuscates the audio by identifying vocalic regions and replacing those regions with pre-recorded vowels, guaranteeing the independence between the identity of the replacement vowel and the identity of the spoken syllable. From the results, the proposed method significantly reduced the speech intelligibility, maintaining the recognizability of the environmental sounds. This algorithm is used as a filter of sensitive signals in the method developed by Liaqat et al. [91], whose goal is to continuously record audio while preserving privacy.

Another application of **audio data** is audio sensing, which is widely used for e-health applications (e.g. cough sensing). Larson et al. developed an algorithm that detects coughs from audio, while guaranteeing privacy [92]. This algorithm achieved privacy by disguising and suppressing speech sounds. Furthermore, Kumar et al. proposed two methods called *sound shredding* and *sound subsampling* to preserve the privacy in audio sensing [93]. *Sound shredding* consists in selecting an audio frame from the original audio and move it to a random location in the released audio, while *sound subsampling* corresponds to collect a part of the raw data instead of the all audio. Thus, there are sufficient information about the context (e.g. the gender of the speaker), but the content of the speech cannot be recognized. However, when the mechanisms are developed without a specific application/goal, relevant information may be lost.

Beyond audio privacy, several PPMs have been developed for **video data**. Boyle et al. started by developing blur and pixelize filters for videos and by studying the effect of those filters in privacy [94]. Nevertheless, the proposed mechanism had some limitations, namely, it only uses two filters and it was not applied to real world settings. The increasing number of video surveillance systems arouse the need of real-time PPMs. For instance, obfuscation techniques were proposed to mask video data [95], a smart camera, named *PrivacyCam*, was proposed to remove private information before producing the video stream [96], and others [97,98].

Regarding the publication of **video data**, Wang et al. proposed a novel privacy notion ϵ-*Object Indistinguishability* for sensitive objects in video data, and a video sanitization technique, named *VERRO* [99]. The proposed privacy notion is based on LDP and guarantees that the objects in the video are indistinguishable. The *VERRO* technique consists in three steps: pre-processing, phase I, and phase II. Pre-processing uses computer vision techniques to identify and track all of the objects and to extract the background in each frame. In phase I, the presence or absence of each object is randomly generated for different frames of the video in order to be indistinguishable. In phase II, *VERRO* generates the synthetic video with the insertion of the synthetic objects into the video according to the presence/absence information randomly generated in phase I. The proposed technique was evaluated in real videos and the results showed its effectiveness and efficiency.

3.3. Cryptographic mechanisms

Cryptographic mechanisms are typically proposed to protect data independently of their type and/or structure. Nevertheless,

the mechanisms can later be developed according to a specific context and/or application (e.g. cloud [100]). Cryptographic mechanisms are often used for Privacy-Preserving Data Mining (PPDM) and distributed privacy-preserving (i.e. between two or more parties), being their goal to privately mine data without revealing individual data. To achieve this goal, several cryptographic mechanisms have been developed [101,102]. However, these mechanisms are usually associated with a high computational cost, which is a clear drawback when comparing with the anonymization/obfuscation mechanisms. The remainder of this section starts by presenting general mechanisms and protocols in the domain of cryptography followed by mechanisms that are suitable for specific data types.

Secure Multiparty Computation (SMC) [103] is a subfield of cryptography that consists in creating methods for different parties to jointly compute a function over their inputs, maintaining the privacy of those inputs. Thus, the privacy of each party is guaranteed from each other party and it is possible to compute different tasks over distance without requiring a trusted third party. Since only the data mining results are revealed to all involved parties, SMC has been extensively studied in the context of PPDM [104].

A basic building block for several SMC techniques is known as oblivious transfer [105]. Even et al. proposed the 1-out-of-2 oblivious transfer that is often used in PPDM [105]. This approach involves two parties, a *sender* and a *receiver*. The *sender* inputs a pair (x_0, x_1) and has no output (i.e. learns nothing), while the *receiver* inputs a bit $\sigma \in \{0, 1\}$ and outputs x_σ (i.e. only learns x_σ). Since inputs are encrypted, the *sender* learns nothing, while the *receiver* learns one out of the two possible inputs that were given by the *sender*.

Garbled circuit is an example of a cryptographic protocol that allows two-party secure computation where two mistrusting parties can jointly compute a function over their private inputs without a trusted third party [106,107]. Moreover, there are other methods that can be used for privacy-preserving computations, namely: secure sum, secure set union, secure size of intersection, scalar product and set intersection [5,101,108].

Homomorphic encryption is a technique that allows operations on encrypted data, generating encrypted results that match with the expected results when decrypted. However, the earlier homomorphic encryption techniques were limited to specific operations. To support various types of functions, fully homomorphic encryption was proposed [109]. This scheme allows to compute a broader number of operations over encrypted data without being able to decrypt. Homomorphic encryption can be used for privacy-preserving outsourced storage and computation [110].

In the context of **structured data**, Jiang and Clifton proposed a two-party framework D*k*A [111,112] that allows to integrate two private tables into a *k*-anonymous dataset, following the definition of SMC. For that, each party locally applies *k*-anonymity, generating a *k*-anonymous table. Then, the parties check if the resulting joint table would be *k*-anonymous, by calculating the intersection size. If the intersection size is at least *k*, the join of the two locally *k*-anonymous tables that is also globally *k*-anonymous is returned. Otherwise, each party further generalizes the data until it is sufficiently anonymized and a *k*-anonymous dataset is achieved. Building from this work, Mohammed et al. proposed two algorithms that, in contrast with the D*k*A, are scalable and allow to securely integrate private data from multiple parties [113].

The Private Information Retrieval (PIR) protocol was proposed in the context of **structured DBs** [114] and was then adapted for **streaming data** [115,116] and for **geospatial data** [117]. In the domain of **structured DB**, the PIR protocol allows users to retrieve an item from a DB without revealing which item is retrieved for the owner of the DB. With respect to **streaming data**, the authors of [115,116] adapted the PIR protocol to be executed in an online environment by considering the size of the query independent of the size of the stream. This work also extended the types of queries that can be performed, guaranteeing efficiency and multiple queries. In the domain of **geospatial data**, the PIR mechanism allows users to query the server of a Location-Based Service (LBS) through an encrypted query without revealing their location [117]. For instance, the user asks the server about the nearest Point of Interest (PoI) through an encrypted query and the server retrieves the nearest PoI according to the user location.

Due to the increasing number of genetic tests and the advancement of genomic research, several privacy concerns about the collection, storage and analysis of **genomic data** have aroused. In order to respond to this problem and to protect such sensitive human data, privacy-preserving techniques have been developed [118–120]. In particular, homomorphic encryption and garbled circuits are privacy-preserving techniques used in the context of genomic data [120]. This survey [120] covers the state-of-the-art PPMs of genomic data.

Ayday et al. proposed a system based on symmetric stream cipher, order-preserving encryption and data masking to preserve the privacy of storage, retrieval and processing of raw aligned **genomic data** (i.e. the aligned outputs of a DNA sequencer) [121]. The raw genomic data of an individual contains hundreds of millions of a sequences of nucleotides on DNA, also known as, short reads. The main objective of the proposed system is to retrieve short reads from the biobank to a certain Medical Unit (MU) without revealing the ambit of the test to the biobank. For that, the proposed system resorts to a certified institution to perform the encryption and sequencing of the short reads that will be stored in encrypted Sequence Alignment Map (SAM) files at a biobank. When the MU requests a certain range of short reads, the biobank privately retrieves the data according to what the MU is authorized to receive. To protect the disclosure of extra information by the MU, certain parts of the encrypted short reads are masked at the biobank, before being sent to the MU.

In order to preserve the privacy of **time series data**, Shi et al. proposed novel Private Stream Aggregation (PSA) methods based on cryptography and differential privacy [122]. The main objective of the proposed method is to guarantee the individual's privacy, while computing aggregate statistics from multiple individuals. Each participant periodically uploads encrypted noisy data to an untrusted data aggregator that is able to privately compute the aggregate statistics over multiple periods of time. Moreover, the proposed approach resorts to a data randomization technique to guarantee the differential privacy of the outcome statistic. Therefore, the data aggregator has the capability to decrypt the noisy sum of all individuals, but is unable to infer extra information about each individual.

In addition to the previous contexts, cryptographic mechanisms are also used in **multimedia data**, namely to protect sensitive contents in images. Content-based Image Retrieval (CBIR) is a well-studied problem in image processing that consists in analyzing and retrieving the information contained in **image data**. Since images can contain sensitive information, several encryption techniques have been proposed to preserve the data privacy, as reviewed in [123]. In [124], the authors proposed a mechanism that supports CBIR over encrypted data. Furthermore, the authors proposed a watermark-based protocol to prevent the illegal distribution of images with copy-deterrence by directly embedding a watermark into the encrypted images before sent to the user. Shen et al. proposed a CBIR mechanism that supports Multiple Image owners with Privacy Protection (MIPP) [125]. The proposed mechanism is based on SMC, where the owners of the images are able to encrypt their images with their own

keys. Thus, the mechanism allows an efficient image retrieval over images collected from multiple sources, while individual image privacy is guaranteed. Finally, from a practical point of view, the tool *iPrivacy* (image privacy) was developed to automatically recommend settings for image sharing [126]. *iPrivacy* detects privacy-sensitive objects in the images and then identify the privacy settings of those objects. Moreover, this tool is able to automatically blur those privacy-sensitive objects to preserve image privacy.

Upon presenting PPMs, based on either anonymization, obfuscation or cryptography, we will now propose a taxonomy of such mechanisms relating them with the heterogeneous data types identified in Section 2.

4. Privacy taxonomy for heterogeneous data types

This section starts by providing a literature review of existing data privacy taxonomies, that is, taxonomies that take into consideration privacy aspects, and ends with the proposal of a novel privacy taxonomy for heterogeneous data types, presenting PPMs that fit to the characteristics of different data types. Moreover, the PPMs are classified according to their application mode in real world, i.e. weather they are suitable for real-time or offline application.

To better understand data privacy, Barker et al. created a taxonomy based on a 3D graph [127]. This graph contains three contributors of data privacy: visibility, granularity, and purpose. Each one of these categories has specific values. For instance, visibility means if the data is visible to all world, third party, house, owner or none. This taxonomy allows us to select the privacy-preserving mechanism according to the values of the categories. Although the authors present a table of the privacy taxonomy with mechanisms from the literature, they only present this analysis for three mechanisms, exclusively according to the axes of the 3D graph, and lacks important PPMs proposed since its publication in 2009.

Sharma et al. presents a comparative study of privacy-preserving techniques [11]. The privacy-preserving techniques are compared according to different characteristics, such as the dataset type, the data type, the information loss, and others. However, the provided comparison considers techniques instead of specific examples of PPMs. Moreover, regarding data types, the authors compare techniques only according to the following three data types: numerical, categorical, and boolean data. On the other hand, Puri et al. only focus on relational and transaction data [12]. For these data types, the authors present existing techniques that ensure privacy while publishing data and, in particular, they present a case study concerning algorithms to anonymize patient data.

Due to the diversity of privacy techniques, Kanwal et al. presents a comparison between different techniques considering their merits, demerits, and their data applications [13]. The authors present a data taxonomy and possible techniques for structured data, semi-structured data, unstructured data, and big data. However, the presented analysis is mainly focused on privacy techniques that can be applied in e-health. In addition, the analysis is performed according to the privacy techniques (e.g. suppression and generalization) and, then, in which PPMs are those techniques applied.

Data privacy is also considered in the analysis of big data [16, 20], where the main challenge of applying privacy models is the computational cost. Both [16] and [20] present taxonomies of big data according to different domains and include security and privacy as one of the aspects. In the domain of security and privacy, the work [16] discusses some existing issues and possible solutions for the following five types of data: streaming data, graph data, scientific, web, retail and financial data. However, the presented solutions are related to both security and privacy issues and, in some cases, correspond only to recommendations/best practices and not to PPMs. With respect to data privacy, the survey [20] only mentions existing mechanisms to preserve privacy without specifying how those mechanisms work or for what types of data they are suitable.

Since the realm of big data contains structured and unstructured data, finding the suitable PPM remains as an open issue. Although there are mechanisms for structured data, extracting the sensitive information from unstructured data is not trivial [128]. In the domain of big data, this is harder due to the amount of data and the associated computational cost. Victor et al. provide a survey on privacy models for big data [129]. In particular, several privacy models are studied, starting with the traditional mechanisms and, then, presenting mechanisms that can be extended for big data. In contrast to our focus that is identifying PPMs according to the data characteristics, the goal of the authors of [129] consists in distinguishing which big data issue is addressed by the mechanisms.

While some existing taxonomies focus on privacy aspects, heterogeneous data types might share common aspects. This results in a challenge when choosing an efficient PPM for each specific and heterogeneous data type. In this paper, we propose a privacy taxonomy that maps data types and their common characteristics with appropriate PPMs, thus serving as a guideline to assess which PPMs are available for specific data types and their underlying characteristics.

Fig. 5 presents the proposed taxonomy, associating the PPMs described in Section 3 and their methodology (anonymization, obfuscation, cryptographic) with suitable data types, as classified in Section 2. For the data types presented in the data taxonomy of Fig. 2, we identified suitable mechanisms based on the data characteristics. In some cases, PPMs were primarily developed for a data type and/or specific for an application and then were extended and adapted for other data types. For instance, in the context of structured data, in general the mechanisms were primarily developed for categorical or numerical data and then expanded for both types. The proposed taxonomy facilitates the selection of PPMs according to the data type and its structure/characteristics. While previous works presented the PPMs and their data applications, our taxonomy starts from the data types according to their structure to identify appropriate PPMs.

On the other hand, a factor that also influences the selection of PPMs is related to how the mechanisms are applied in real world scenarios, that is, if they can be applied in real-time (online) or offline. This application mode depends on several aspects, such as the data type, the complexity of the mechanism, or even the objective of the service and its time constraints. Thus, some of the mechanisms are developed to be executed during data collection, while others are developed for data publishing and, therefore, need data to be complete. This is the case of textual data, for example, in where a text must be completely collected before applying the PPM. In real-time contexts, due to the run time requirements, the complexity and efficiency of the mechanisms have a huge impact and should be considered during the implementation. Nevertheless, the mechanisms developed for real-time scenarios can also be applied offline. For example, considering streaming data, we can apply PPMs at collection time or after the data collection, when data is complete and needs to be privately published. Similarly, in geospatial data, we can consider PPMs at collection time (e.g. to protect location points) or afterwards (e.g. to protect a trajectory or distinct location coordinates).

Fig. 6 schematizes the PPMs studied in Section 3 according to the data types presented in Section 2 and the application mode

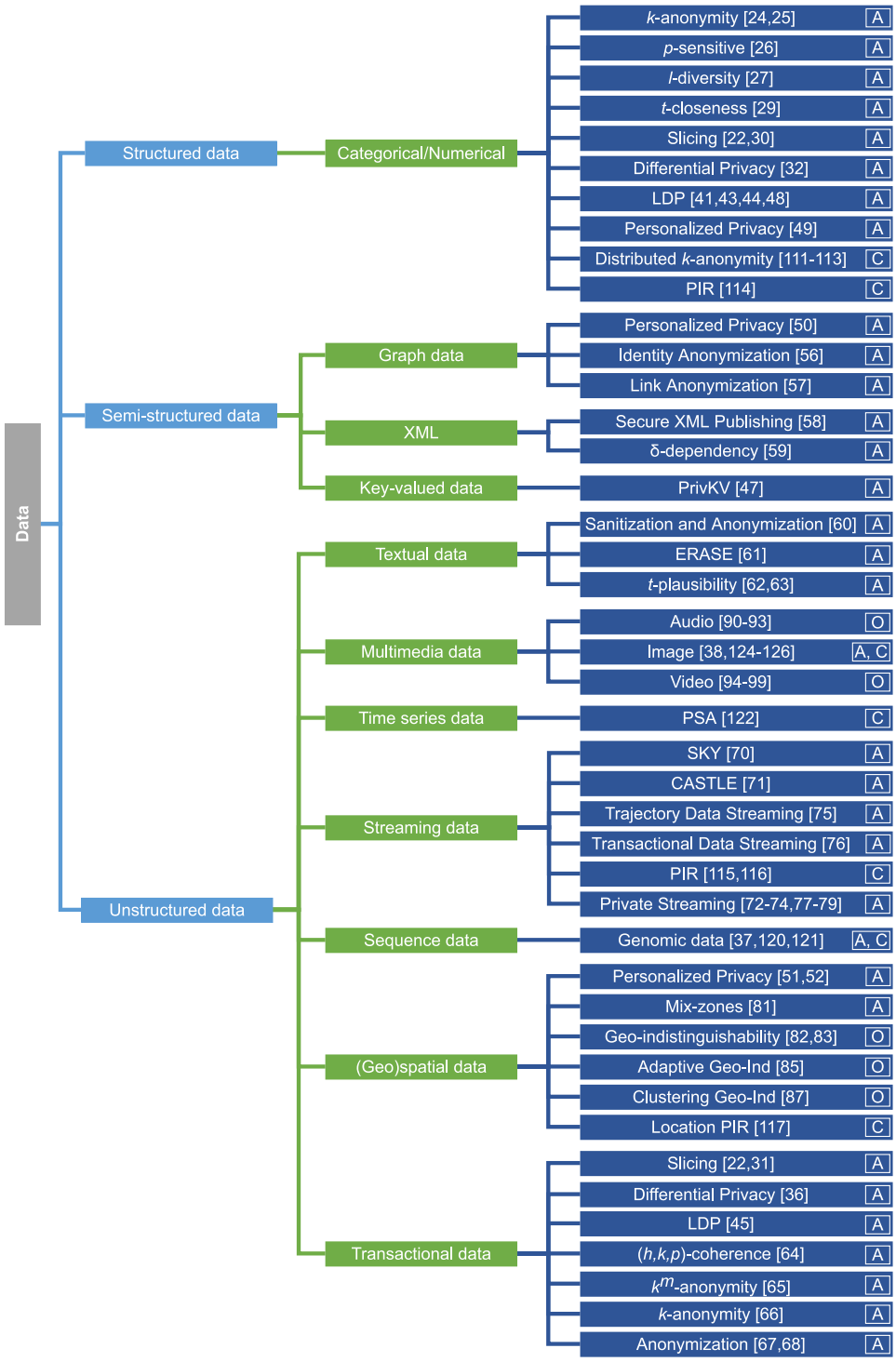

Fig. 5. Privacy taxonomy that establishes a relation between data types and PPMs, where the methodology of the mechanisms is represented as follows: Anonymization (A), Obfuscation (O) and Cryptographic (C).

of those mechanisms, that is, if the mechanisms are applicable in real-time and/or offline. For instance, location coordinates can be protected at collection/real-time, through geo-indistinguishable mechanisms [82,85,87], and/or offline. Moreover, there also exist mechanisms for real-time privacy protection of streaming data and time series data, such as SKY [70] and PSA [122], respectively. Privacy protection of XML data, instead lacks mechanisms that operate in real-time, with current proposals [58,59] being for offline processing and XML data publishing.

5. Privacy tools

This section covers existing privacy tools, namely, their objectives and implementation details. Beyond anonymizing data, some tools allow the assessment of different configurations of PPMs, which in turn enables the evaluation of the achieved privacy and utility level.

ARX Data Anonymization Tool is an open source tool for anonymizing sensitive personal data [7,130] that is available

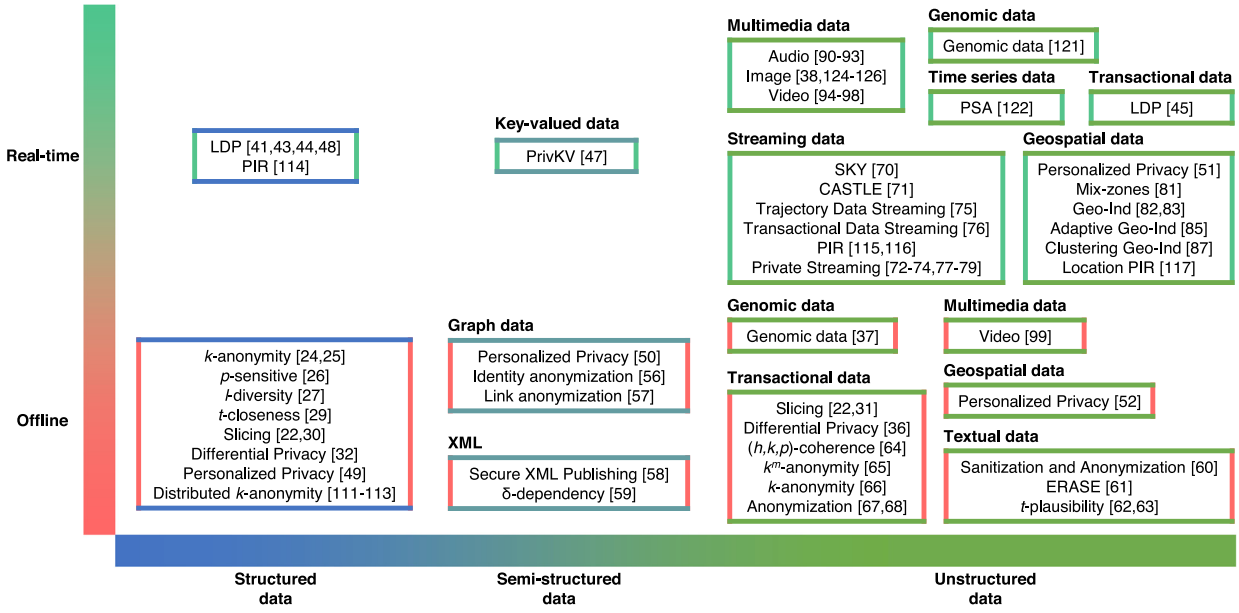

Fig. 6. Privacy schema that establishes a relation between data types, PPMs, and their application mode. The mechanisms classified as real-time can also be applied offline.

in [131]. This tool enables users to import data, configure, explore, analyze, and export data. In each step, the user is able to define a privacy model, to filter and analyze the solution space, and to evaluate the utility of the data. ARX is a complete tool that imports structured data only. This tool is written in Java and provides an API. Regarding the privacy models, it has already some mechanisms implemented, namely: syntactic privacy models (such as k-anonymity, l-diversity, t-closeness, and many others), statistical privacy models (e.g. population uniqueness), and semantic privacy models (e.g. differential privacy). ARX does not implement any attack/adversary model, but features the implementation of models for assessment of the risk of re-identification.

Similarly to ARX, *Amnesia* is a data anonymization tool [8] that is available as an online dashboard in [132]. The main objective of this tool is to transform relational and transactional databases into anonymized data by using generalization and suppression mechanisms. This tool focuses on Privacy-Preserving Data Publishing (PPDP) techniques and supports the following mechanisms: k-anonymity and k^m-anonymity. The goal of *Amnesia* is to remove sensitive information that can be used as identifying information from the published data. Moreover, this tool allows to remove not only the direct identifiers but also quasi-identifiers.

sdcTools consist in tools to provide Statistical Disclosure Control (SDC) [133]. The ARGUS software was developed in order to have a free software solution that guarantees the SDC. This software consists in two modules that implements protection mechanisms for microdata (such as census data that contains numerical and categorical information), μ-*ARGUS* [134] and τ-*ARGUS* [135]. *sdcMicro* [136] was also developed to anonymize microdata. This tool implements several anonymization techniques such as: k-anonymity, suppression, top and bottom coding, microaggregation, and others. Regarding the implementation, *sdcMicro* is available as open-source and consists in an R-package.

Anonimatron is an open source project to anonymize data from structured databases and files [137]. This tool is written in Java and runs on Windows, Mac OSX, and Linux derivatives. Moreover, it supports data from multiple databases. The main goal of *Anonimatron* is to anonymize or de-personalize data. To achieve that, this tool replaces the value of an attribute in the database with another one and saves that relation in a synonym. These synonyms are applied in all the tables of the database, such that the database remains similar but anonymized. The synonyms are stored in a file that can be saved for later use.

Aircloak is a privacy-preserving solution that uses a unique and patented data anonymization method [138]. *Aircloak* does not modify the database and supports all data types including unstructured text. *Aircloak*'s anonymization is based on existing techniques such as k-anonymity, low-count, suppression, top and bottom coding, differential privacy noise, and other patented open concepts. From these techniques, *Aircloak* provides a dynamic anonymization approach that consists in adding noise. Finally, *Aircloak* has a free-to-use version for universities and a full version for enterprises.

Table 1 summarizes and compares the presented privacy tools. As shown in the table, the majority of the discussed tools are available as open-source and can be extensible, which is an advantage. Regarding the data types accepted by the tools, due to the ease of data handling, most of the proposed tools are developed for structured data, namely, for tabular data (i.e. data stored in tables) and microdata (i.e. relational data about individuals). With respect to the implemented PPMs, in general, tools are focused on anonymization mechanisms, which can be a consequence of the supported data types, as most PPMs for structured data are based on anonymization (c.f. Fig. 5). Concerning the privacy and utility trade-off, most tools allow for the privacy evaluation, but some lack the utility assessment. This is a crucial drawback as the selection of a PPM should weigh this trade-off.

The last two aspects to consider in this comparison are related to Graphical User Interface (GUI) and Web App features, presented in the last two columns of Table 1. All of the tools have a GUI, with *ARX* providing an API, but only *Amnesia* and *Aircloak* provide a Web App, which allows the online use and access to the tool. From the discussion, these three tools are the most complete ones. However, *Aircloak* has the downside of being closed source, thus limiting improvements by the community, and might have additional utilization costs. Finally, although some of the tools allow evaluating the risk of re-identification through established risk-assessment models [139, Chapter 16], none of the tools implement attacks [140] over data, which are a relevant complement to assess the practical validity of the privacy level achieved by the available PPMs.

Table 1
Comparison between existing privacy tools.

Tool	Open source	Data type	PPMs	Privacy evaluation	Utility evaluation	GUI	Web app
ARX [131]	✓	Structured data: Tabular data	Syntactic, statistical, and semantic privacy models	✓	✓	✓[a]	✗
Amnesia [8]	✓	Structured data: Tabular data Unstructured data: Set-valued data	k-anonymity and k^m-anonymity	✓	✓	✓	✓
μ-argus [134]	✓	Structured data: Microdata	Anonymization	✓	✗	✓	✗
τ-argus [135]	✓	Structured data: Microdata	Anonymization	✓	✗	✓	✗
sdcMicro [136]	✓	Structured data: Microdata	Anonymization	✓	✓	✓	✗
Anonimatron [137]	✓	Structured data: Tabular data	Anonymization	✗	✗	✓	✗
Aircloack [138]	✗	All data types	Aircloak's anonymization approach	✓	✓	✓	✓

[a]API provided.

6. Open issues and future directions

Although privacy is being widely studied, due to the lack of a standardized and universal definition of privacy, it is still challenging to have standard methods to compare the existing PPMs. The existing tools aim to systematize and create logical structures for privacy, but are often focused on specific types of data. Thus, there is not yet a publicly available tool that implements and evaluates PPMs for heterogeneous data types. Furthermore, since selecting and configuring the proper PPM is not a trivial process, a future direction consists in creating a unified tool that is able to automatically suggest PPMs according to the data type.

Regarding the development of PPMs, current mechanisms are usually focused on a specific data type. In some cases (e.g. location data), the effect of multiple disclosures of data has been analyzed [86,141] and led to novel PPMs that take the correlation of multiple instances of the same type of data into consideration [85,87]. However, the increasing amount of data being gathered and shared nowadays, opens a venue to privacy attacks that take into account the correlation between different (heterogeneous) data types (e.g. sensed data from illumination and temperature) [23,142,143] that can be used for powerful/innovative side-channel attacks [144]. Therefore, novel PPMs should consider not only the correlation of multiple instances of the same data type, but also correlation with other (heterogeneous) data types.

The current landscape of PPMs has the common factor that these mechanisms require configuring privacy parameters that can either be hard to define (e.g. the meaning of epsilon in differential privacy) [32,145], or recalculated for each environment (e.g. the k parameter in k-anonymity [24] or follow-up alternatives of l-diversity and t-closeness). It is well known that the lack of usability has limited the successful application of security and privacy systems throughout time [146–148]. This calls for automated mechanisms that are able to successfully configure and adapt privacy mechanisms to current context as well as user profiles for different and heterogeneous data types.

Although some research has started considering mechanisms applied at collection time, this topic is far from being mature and is still considered an open issue. PPMs applied at collection time empower users to regain control over their data with no need to trust a third-party entity. To achieve enhanced mechanisms, the development should take into account not only the trade-off between the privacy level and the utility level, but also the efficiency of the mechanisms in order to be used in run time.

Since data is collected with a given purpose, PPMs cannot disregard the utility of data, such that data collectors may still be able to extract useful information and provide relevant services. Several machine learning mechanisms have been designed by the community to learn from data. However, this data and the learned outcomes can contain sensitive information, raising privacy concerns [149]. Ideally, mechanisms would learn from data with privacy guarantees. Recent works [149–151] proposed mechanisms to learn from anonymized/encrypted data and showed that it is possible to reach satisfactory results, although many of the privacy-preserving machine learning techniques are related to a specific machine learning algorithm and/or computationally expensive [149].

Finally, most users are still unaware about the privacy risks of sharing data. This calls for mechanisms to raise users' awareness. For instance, people should be educated about the risks and how they can protect their privacy through changes in their behavior. Currently, there are some frameworks to educate users on privacy matters [152] and others to raise users' awareness [153]. It would be interesting to have combined mechanisms to raise awareness but also educate users by helping them in their privacy-related choices.

7. Conclusion

Due to the ubiquitousness of smart devices, there are large amounts of data continuously being collected by possibly untrustworthy entities, which raises several privacy concerns. Privacy-Preserving Mechanisms (PPMs) have been proposed to address this challenge and to protect users' privacy. However, due to the heterogeneity of the data and the lack of generic PPMs, selecting the proper mechanism remains a challenge. This survey identifies and classifies existing heterogeneous data types and presents the state-of-the-art PPMs according to their purpose. With this knowledge, we propose a novel privacy taxonomy that establishes a relation between PPMs and data types. Specifically, the proposed taxonomy differentiates which PPMs are applicable for the characteristics of each data type. Additionally, it distinguishes whether the PPMs are applicable in real-time or offline. Finally, this survey presents and compares tools for privacy protection. The performed analysis allows us to conclude about the need of novel PPMs for heterogeneous data types and a unified tool that implements PPMs for different types of data, as well as techniques for privacy evaluation, including methodologies for re-identification risk assessment, complemented with practical re-identification attacks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the European Regional Development Fund (FEDER), through the Regional Operational Programme of Centre (CENTRO 2020) of the Portugal 2020 framework and FCT, Portugal under the MIT Portugal Program [Project SNOB-5G with

Nr. 045929 (CENTRO-01-0247-FEDER-045929)]. Mariana Cunha and Ricardo Mendes wish to acknowledge the Portuguese funding institution FCT — Foundation for Science and Technology for supporting their research under the Ph.D. grant 2020.04714.BD and SFRH/BD/128599/2017, respectively.

References

[1] Z. Yan, P. Zhang, A.V. Vasilakos, A survey on trust management for Internet of Things, J. Netw. Comput. Appl. 42 (2014) 120–134.
[2] J. Clement, Online privacy in the United States - statistics & facts, 2020, https://www.statista.com/topics/2476/online-privacy/ (consulted in September 2020).
[3] Y.A.A.S. Aldeen, M. Salleh, M.A. Razzaque, A comprehensive review on privacy preserving data mining, SpringerPlus 4 (1) (2015) 694.
[4] A. Shah, R. Gulati, Privacy preserving data mining: Techniques classification and implications—A survey, Int. J. Comput. Appl. 137 (12) (2016) 40–46.
[5] R. Mendes, J.P. Vilela, Privacy-preserving data mining: Methods, metrics, and applications, IEEE Access 5 (2017) 10562–10582.
[6] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, J.-Y. Le Boudec, Protecting location privacy: Optimal strategy against localization attacks, in: Proceedings of the 2012 ACM Conference on Computer and Communications Security, CCS '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 617–627, http://dx.doi.org/10.1145/2382196.2382261.
[7] F. Prasser, F. Kohlmayer, Putting statistical disclosure control into practice: The arx data anonymization tool, in: Medical Data Privacy Handbook, Springer International Publishing, Cham, 2015, pp. 111–148, http://dx.doi.org/10.1007/978-3-319-23633-9_6.
[8] M. Terrovitis, D. Tsitsigkos, Amnesia, Institute for the Management of Information Systems, https://amnesia.openaire.eu/ (consulted in September 2020).
[9] R. Shokri, V. Bindschaedler, G. Theodorakopoulos, G. Danezis, J.-P. Hubaux, J.-Y. Le Boudec, Location-Privacy and Mobility Meter, http://icapeople.epfl.ch/rshokri/lpm/doc/index.html (consulted in September 2020).
[10] V. Primault, M. Maouche, A. Boutet, S.B. Mokhtar, S. Bouchenak, L. Brunie, ACCIO: How to make location privacy experimentation open and easy, in: 2018 IEEE 38th International Conference on Distributed Computing Systems, ICDCS, IEEE, 2018, pp. 896–906.
[11] M. Sharma, A. Chaudhary, M. Mathuria, S. Chaudhary, A review study on the privacy preserving data mining techniques and approaches, Int. J. Comput. Sci. Telecommun. 4 (9) (2013) 42–46.
[12] V. Puri, S. Sachdeva, P. Kaur, Privacy preserving publication of relational and transaction data: Survey on the anonymization of patient data, Comp. Sci. Rev. 32 (2019) 45–61, http://dx.doi.org/10.1016/j.cosrev.2019.02.001.
[13] T. Kanwal, A. Anjum, A. Khan, Privacy preservation in e-health cloud: Taxonomy, privacy requirements, feasibility analysis, and opportunities, Cluster Comput. (2020) 1–25, http://dx.doi.org/10.1007/s10586-020-03106-1.
[14] C.C. Aggarwal, S.Y. Philip, A general survey of privacy-preserving data mining models and algorithms, in: Privacy-Preserving Data Mining: Models and Algorithms, Springer US, Boston, MA, 2008, pp. 11–52.
[15] Top 10 IoT sensor types, 2019, https://behrtech.com/blog/top-10-iot-sensor-types/ (consulted in September 2020).
[16] B.D.W. Group, Big data taxonomy, 2014, https://downloads.cloudsecurityalliance.org/initiatives/bdwg/Big_Data_Taxonomy.pdf (consulted in September 2020).
[17] T. Efraim, R. Sharda, D. Delen, Business Intelligence and Analytics: Systems for Decision Support, Prentice Hall, New Jersey, 2010.
[18] T. Gupta, Types of Data Sets in Data Science, Data Mining & Machine Learning, 2019, https://towardsdatascience.com/types-of-data-sets-in-data-science-data-mining-machine-learning-eb47c80af7a (consulted in September 2020).
[19] L. Wang, Heterogeneous data and big data analytics, Autom. Control Inf. Sci. 3 (1) (2017) 8–15.
[20] R. Patgiri, A taxonomy on big data: Survey, 2019, arXiv:1808.08474.
[21] X. Xiao, Y. Tao, Anatomy: Simple and effective privacy preservation, in: Proceedings of the 32nd International Conference on Very Large Data Bases, VLDB '06, VLDB Endowment, 2006, pp. 139–150.
[22] T. Li, N. Li, J. Zhang, I. Molloy, Slicing: A new approach for privacy preserving data publishing, IEEE Trans. Knowl. Data Eng. 24 (3) (2012) 561–574, http://dx.doi.org/10.1109/TKDE.2010.236.
[23] B.C. Fung, K. Wang, R. Chen, P.S. Yu, Privacy-preserving data publishing: A survey of recent developments, ACM Comput. Surv. 42 (4) (2010) 1–53.
[24] P. Samarati, L. Sweeney, Protecting Privacy when Disclosing Information: k-Anonymity and Its Enforcement Through Generalization and Suppression, Tech. Rep., SRI International, 1998.
[25] P. Samarati, L. Sweeney, Generalizing data to provide anonymity when disclosing information, in: PODS, vol. 98, Citeseer, 1998, p. 188.
[26] T.M. Truta, B. Vinay, Privacy protection: p-sensitive k-anonymity property, in: 22nd International Conference on Data Engineering Workshops, ICDEW'06, IEEE, 2006, p. 94.
[27] A. Machanavajjhala, J. Gehrke, D. Kifer, M. Venkitasubramaniam, ℓ-diversity: Privacy beyond k-anonymity, in: 22nd International Conference on Data Engineering, ICDE'06, IEEE, 2006, p. 24.
[28] K. Rajendran, M. Jayabalan, M.E. Rana, A study on k-anonymity, l-diversity, and t-closeness techniques, Int. J. Comput. Sci. Netw. Secur. 17 (12) (2017) 172.
[29] N. Li, T. Li, S. Venkatasubramanian, t-closeness: Privacy beyond k-anonymity and l-diversity, in: 2007 IEEE 23rd International Conference on Data Engineering, IEEE, 2007, pp. 106–115.
[30] V.S. Susan, T. Christopher, Anatomisation with slicing: A new privacy preservation approach for multiple sensitive attributes, SpringerPlus 5 (1) (2016) 1–21.
[31] M. Wang, Z. Jiang, Y. Zhang, H. Yang, T-closeness slicing: A new privacy-preserving approach for transactional data publishing, INFORMS J. Comput. 30 (3) (2018) 438–453.
[32] C. Dwork, Differential privacy, in: Automata, Languages and Programming, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 1–12.
[33] C. Dwork, Differential privacy: A survey of results, in: Theory and Applications of Models of Computation, Springer, Berlin, Heidelberg, 2008, pp. 1–19.
[34] C. Dwork, F. McSherry, K. Nissim, A. Smith, Calibrating noise to sensitivity in private data analysis, in: Theory of Cryptography, Springer, Berlin, Heidelberg, 2006, pp. 265–284.
[35] C. Dwork, A. Roth, et al., The algorithmic foundations of differential privacy, Found. Trends Theor. Comput. Sci. 9 (3–4) (2014) 211–407.
[36] R. Chen, N. Mohammed, B.C. Fung, B.C. Desai, L. Xiong, Publishing set-valued data via differential privacy, Proc. VLDB Endow. 4 (11) (2011) 1087–1098.
[37] C. Uhlerop, A. Slavković, S.E. Fienberg, Privacy-preserving data sharing for genome-wide association studies, J. Priv. Confid. 5 (1) (2013) 137.
[38] L. Fan, Image pixelization with differential privacy, in: Data and Applications Security and Privacy XXXII, Springer International Publishing, Cham, 2018, pp. 148–162.
[39] B. Bebensee, Local differential privacy: A tutorial, 2019, arXiv:1907.11908.
[40] S.P. Kasiviswanathan, H.K. Lee, K. Nissim, S. Raskhodnikova, A. Smith, What can we learn privately? SIAM J. Comput. 40 (3) (2011) 793–826.
[41] Ú. Erlingsson, V. Pihur, A. Korolova, RAPPOR: Randomized aggregatable privacy-preserving ordinal response, in: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, CCS '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 1054–1067, http://dx.doi.org/10.1145/2660267.2660348.
[42] A. Differential Privacy Team, Learning with privacy at scale, 2017, https://machinelearning.apple.com/docs/learning-with-privacy-at-scale/appledifferentialprivacysystem.pdf.
[43] B. Ding, J. Kulkarni, S. Yekhanin, Collecting telemetry data privately, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 3571–3580.
[44] T. Wang, J. Blocki, N. Li, S. Jha, Locally differentially private protocols for frequency estimation, in: 26th USENIX Security Symposium, USENIX Security 17, USENIX Association, Vancouver, BC, 2017, pp. 729–745, https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/wang-tianhao.
[45] Z. Qin, Y. Yang, T. Yu, I. Khalil, X. Xiao, K. Ren, Heavy hitter estimation over set-valued data with local differential privacy, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 192–203, http://dx.doi.org/10.1145/2976749.2978409.
[46] M. Yang, L. Lyu, J. Zhao, T. Zhu, K.-Y. Lam, Local differential privacy and its applications: A comprehensive survey, 2020, arXiv:2008.03686.
[47] Q. Ye, H. Hu, X. Meng, H. Zheng, PrivKV: Key-value data collection with local differential privacy, in: 2019 IEEE Symposium on Security and Privacy, SP, IEEE, 2019, pp. 317–331.
[48] N. Wang, X. Xiao, Y. Yang, J. Zhao, S.C. Hui, H. Shin, J. Shin, G. Yu, Collecting and analyzing multidimensional data with local differential privacy, in: 2019 IEEE 35th International Conference on Data Engineering, ICDE, IEEE, 2019, pp. 638–649.
[49] X. Xiao, Y. Tao, Personalized privacy preservation, in: Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data, 2006, pp. 229–240.
[50] M. Yuan, L. Chen, P.S. Yu, Personalized privacy protection in social networks, Proc. VLDB Endow. 4 (2) (2010) 141–150.
[51] B. Agir, T.G. Papaioannou, R. Narendula, K. Aberer, J.-P. Hubaux, User-side adaptive protection of location privacy in participatory sensing, GeoInformatica 18 (1) (2014) 165–191.

[52] E.G. Komishani, M. Abadi, F. Deldar, PPTD: Preserving personalized privacy in trajectory data publishing by sensitive attribute generalization and trajectory local suppression, Knowl.-Based Syst. 94 (2016) 43–59.
[53] B. Zhou, J. Pei, W. Luk, A brief survey on anonymization techniques for privacy preserving publishing of social network data, ACM SIGKDD Explor. Newsl. 10 (2) (2008) 12–22.
[54] X. Wu, X. Ying, K. Liu, L. Chen, A survey of privacy-preservation of graphs and social networks, in: Managing and Mining Graph Data, Springer, Boston, MA, 2010, pp. 421–453, http://dx.doi.org/10.1007/978-1-4419-6045-0_14.
[55] J. Casas-Roma, J. Herrera-Joancomartí, V. Torra, A survey of graph-modification techniques for privacy-preserving on networks, Artif. Intell. Rev. 47 (3) (2017) 341–366.
[56] K. Liu, E. Terzi, Towards identity anonymization on graphs, in: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 93–106, http://dx.doi.org/10.1145/1376616.1376629.
[57] E. Zheleva, L. Getoor, Preserving the privacy of sensitive relationships in graph data, in: Privacy, Security, and Trust in KDD, Springer, Berlin, Heidelberg, 2008, pp. 153–171.
[58] X. Yang, C. Li, Secure XML publishing without information leakage in the presence of data inference, in: Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30, VLDB '04, VLDB Endowment, 2004, pp. 96–107.
[59] A.H. Landberg, K. Nguyen, E. Pardede, J.W. Rahayu, δ-dependency for privacy-preserving XML data publishing, J. Biomed. Inform. 50 (2014) 77–94.
[60] Y. Saygin, D. Hakkini-Tur, G. Tur, Sanitization and anonymization of document repositories, in: Web and Information Security, IGI Global, 2006, pp. 133–148.
[61] V.T. Chakaravarthy, H. Gupta, P. Roy, M.K. Mohania, Efficient techniques for document sanitization, in: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 843–852, http://dx.doi.org/10.1145/1458082.1458194.
[62] W. Jiang, M. Murugesan, C. Clifton, L. Si, t-plausibility: Semantic preserving text sanitization, in: 2009 International Conference on Computational Science and Engineering, Vol. 3, IEEE, 2009, pp. 68–75.
[63] B. Anandan, C. Clifton, W. Jiang, M. Murugesan, P. Pastrana-Camacho, L. Si, t-plausibility: Generalizing words to desensitize text, Trans. Data Priv. 5 (3) (2012) 505–534.
[64] Y. Xu, K. Wang, A.W.-C. Fu, P.S. Yu, Anonymizing transaction databases for publication, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 767–775, http://dx.doi.org/10.1145/1401890.1401982.
[65] M. Terrovitis, N. Mamoulis, P. Kalnis, Privacy-preserving anonymization of set-valued data, Proc. VLDB Endow. 1 (1) (2008) 115–125.
[66] Y. He, J.F. Naughton, Anonymization of set-valued data via top-down, local generalization, Proc. VLDB Endow. 2 (1) (2009) 934–945.
[67] J. Liu, K. Wang, Anonymizing transaction data by integrating suppression and generalization, in: Advances in Knowledge Discovery and Data Mining, Springer, Berlin, Heidelberg, 2010, pp. 171–180.
[68] G. Ghinita, P. Kalnis, Y. Tao, Anonymous publication of sensitive transactional data, IEEE Trans. Knowl. Data Eng. 23 (2) (2010) 161–174.
[69] A.B. Sakpere, A.V. Kayem, A state-of-the-art review of data stream anonymization schemes, in: Information Security in Diverse Computing Environments, IGI Global, 2014, pp. 24–50.
[70] J. Li, B.C. Ooi, W. Wang, Anonymizing streaming data for privacy protection, in: 2008 IEEE 24th International Conference on Data Engineering, IEEE, 2008, pp. 1367–1369.
[71] J. Cao, B. Carminati, E. Ferrari, K.-L. Tan, Castle: Continuously anonymizing data streams, IEEE Trans. Dependable Secure Comput. 8 (3) (2010) 337–352.
[72] B. Zhou, Y. Han, J. Pei, B. Jiang, Y. Tao, Y. Jia, Continuous privacy preserving publishing of data streams, in: Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, EDBT '09, Association for Computing Machinery, New York, NY, USA, 2009, pp. 648–659, http://dx.doi.org/10.1145/1516360.1516435.
[73] S. Kim, M.K. Sung, Y.D. Chung, A framework to preserve the privacy of electronic health data streams, J. Biomed. Inform. 50 (2014) 95–106.
[74] J. Wang, R. Zhu, S. Liu, A differentially private unscented Kalman filter for streaming data in IoT, IEEE Access 6 (2018) 6487–6495.
[75] K. Al-Hussaeni, B.C. Fung, W.K. Cheung, Privacy-preserving trajectory stream publishing, Data Knowl. Eng. 94 (2014) 89–109.
[76] J. Wang, C. Deng, X. Li, Two privacy-preserving approaches for publishing transactional data streams, IEEE Access 6 (2018) 23648–23658.
[77] J. Zhang, H. Li, X. Liu, Y. Luo, F. Chen, H. Wang, L. Chang, On efficient and robust anonymization for privacy protection on massive streaming categorical information, IEEE Trans. Dependable Secure Comput. 14 (5) (2015) 507–520.
[78] M.A.P. Chamikara, P. Bertók, D. Liu, S. Camtepe, I. Khalil, Efficient data perturbation for privacy preserving and accurate data stream mining, Pervasive Mob. Comput. 48 (2018) 1–19.
[79] B. Denham, R. Pears, M.A. Naeem, Enhancing random projection with independent and cumulative additive noise for privacy-preserving data stream mining, Expert Syst. Appl. 152 (2020) 113380.
[80] K. Sampigethaya, R. Poovendran, A survey on mix networks and their secure applications, Proc. IEEE 94 (12) (2006) 2142–2181.
[81] A.R. Beresford, F. Stajano, Location privacy in pervasive computing, IEEE Pervasive Comput. 2 (1) (2003) 46–55.
[82] M.E. Andrés, N.E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, Geo-indistinguishability: Differential privacy for location-based systems, in: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, CCS '13, Association for Computing Machinery, New York, NY, USA, 2013, pp. 901–914, http://dx.doi.org/10.1145/2508859.2516735.
[83] K. Chatzikokolakis, E. Elsalamouny, C. Palamidessi, Efficient utility improvement for location privacy, Proc. Privacy Enhanc. Technol. 2017 (4) (2017) 308–328.
[84] S. Oya, C. Troncoso, F. Pérez-González, Rethinking location privacy for unknown mobility behaviors, in: 2019 IEEE European Symposium on Security and Privacy, EuroS&P, IEEE, 2019, pp. 416–431.
[85] R. Al-Dhubhani, J.M. Cazalas, An adaptive geo-indistinguishability mechanism for continuous LBS queries, Wirel. Netw. 24 (8) (2018) 3221–3239.
[86] R. Mendes, M. Cunha, J.P. Vilela, Impact of frequency of location reports on the privacy level of geo-indistinguishability, Proc. Privacy Enhanc. Technol. 2020 (2) (2020) 379–396.
[87] M. Cunha, R. Mendes, J.P. Vilela, Clustering geo-indistinguishability for privacy of continuous location traces, in: 2019 4th International Conference on Computing, Communications and Security, ICCCS, IEEE, 2019, pp. 1–8, http://dx.doi.org/10.1109/CCCS.2019.8888111.
[88] R.A. Cole, Y. Yan, B. Mak, M. Fanty, T. Bailey, The contribution of consonants versus vowels to word recognition in fluent speech, in: 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, Vol. 2, IEEE, 1996, pp. 853–856.
[89] D. Kewley-Port, T.Z. Burkle, J.H. Lee, Contribution of consonant versus vowel information to sentence intelligibility for young normal-hearing and elderly hearing-impaired listeners, J. Acoust. Soc. Am. 122 (4) (2007) 2365–2375.
[90] F. Chen, J. Adcock, S. Krishnagiri, Audio privacy: Reducing speech intelligibility while preserving environmental sounds, in: Proceedings of the 16th ACM International Conference on Multimedia, MM '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 733–736, http://dx.doi.org/10.1145/1459359.1459472.
[91] D. Liaqat, E. Nemati, M. Rahman, J. Kuang, A method for preserving privacy during audio recordings by filtering speech, in: 2017 IEEE Life Sciences Conference, LSC, IEEE, 2017, pp. 79–82.
[92] E.C. Larson, T. Lee, S. Liu, M. Rosenfeld, S.N. Patel, Accurate and privacy preserving cough sensing using a low-cost microphone, in: Proceedings of the 13th International Conference on Ubiquitous Computing, UbiComp '11, Association for Computing Machinery, New York, NY, USA, 2011, pp. 375–384, http://dx.doi.org/10.1145/2030112.2030163.
[93] S. Kumar, L.T. Nguyen, M. Zeng, K. Liu, J. Zhang, Sound shredding: Privacy preserved audio sensing, in: Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications, HotMobile '15, Association for Computing Machinery, New York, NY, USA, 2015, pp. 135–140, http://dx.doi.org/10.1145/2699343.2699366.
[94] M. Boyle, C. Edwards, S. Greenberg, The effects of filtered video on awareness and privacy, in: Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, CSCW '00, Association for Computing Machinery, New York, NY, USA, 2000, pp. 1–10, http://dx.doi.org/10.1145/358916.358935.
[95] J. Wickramasuriya, M. Datt, S. Mehrotra, N. Venkatasubramanian, Privacy protecting data collection in media spaces, in: Proceedings of the 12th Annual ACM International Conference on Multimedia, MULTIMEDIA '04, Association for Computing Machinery, New York, NY, USA, 2004, pp. 48–55, http://dx.doi.org/10.1145/1027527.1027537.
[96] A. Senior, S. Pankanti, A. Hampapur, L. Brown, Y.-L. Tian, A. Ekin, J. Connell, C.F. Shu, M. Lu, Enabling video privacy through computer vision, IEEE Secur. Priv. 3 (3) (2005) 50–57.
[97] F. Dufaux, T. Ebrahimi, Scrambling for privacy protection in video surveillance systems, IEEE Trans. Circuits Syst. Video Technol. 18 (8) (2008) 1168–1174.
[98] M. Upmanyu, A.M. Namboodiri, K. Srinathan, C. Jawahar, Efficient privacy preserving video surveillance, in: 2009 IEEE 12th International Conference on Computer Vision, IEEE, 2009, pp. 1639–1646.
[99] H. Wang, Y. Hong, Y. Kong, J. Vaidya, Publishing video data with indistinguishable objects, in: Advances in Database Technology - EDBT 2020, OpenProceedings.org, 2020, pp. 323–334.

[100] N. Kaaniche, M. Laurent, Data security and privacy preservation in cloud storage environments based on cryptographic mechanisms, Comput. Commun. 111 (2017) 120–141.
[101] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, M.Y. Zhu, Tools for privacy preserving distributed data mining, ACM SIGKDD Explor. Newsl. 4 (2) (2002) 28–34.
[102] B. Pinkas, Cryptographic techniques for privacy-preserving data mining, ACM SIGKDD Explor. Newsl. 4 (2) (2002) 12–19.
[103] O. Goldreich, Secure multi-party computation, Manuscript. Preliminary version 78, 1998.
[104] Y. Lindell, Secure multiparty computation for privacy preserving data mining, in: Encyclopedia of Data Warehousing and Mining, IGI global, 2005, pp. 1005–1009.
[105] S. Even, O. Goldreich, A. Lempel, A randomized protocol for signing contracts, Commun. ACM 28 (6) (1985) 637–647.
[106] Y. Huang, D. Evans, J. Katz, L. Malka, Faster secure two-party computation using garbled circuits, in: Proceedings of the 20th USENIX Conference on Security, SEC'11, USENIX Association, USA, 2011, p. 35.
[107] M. Bellare, V.T. Hoang, P. Rogaway, Foundations of garbled circuits, in: Proceedings of the 2012 ACM Conference on Computer and Communications Security, CCS '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 784–796, http://dx.doi.org/10.1145/2382196.2382279.
[108] M.J. Freedman, K. Nissim, B. Pinkas, Efficient private matching and set intersection, in: International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2004, pp. 1–19.
[109] C. Gentry, Fully homomorphic encryption using ideal lattices, in: Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing, STOC '09, Association for Computing Machinery, New York, NY, USA, 2009, pp. 169–178, http://dx.doi.org/10.1145/1536414.1536440.
[110] T.S. Fun, A. Samsudin, A survey of homomorphic encryption for outsourced big data computation, KSII Trans. Internet Inf. Syst. 10 (8) (2016) 3826–3851.
[111] W. Jiang, C. Clifton, Privacy-preserving distributed k-anonymity, in: IFIP Annual Conference on Data and Applications Security and Privacy, Springer, 2005, pp. 166–177.
[112] W. Jiang, C. Clifton, A secure distributed framework for achieving k-anonymity, VLDB J. 15 (4) (2006) 316–333.
[113] N. Mohammed, B.C. Fung, M. Debbabi, Anonymity meets game theory: Secure data integration with malicious participants, VLDB J. 20 (4) (2011) 567–588.
[114] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, Private information retrieval, in: Proceedings of IEEE 36th Annual Foundations of Computer Science, IEEE, 1995, pp. 41–50.
[115] R. Ostrovsky, W.E. Skeith, Private searching on streaming data, in: Advances in Cryptology – CRYPTO 2005, Springer, Berlin, Heidelberg, 2005, pp. 223–240.
[116] R. Ostrovsky, W.E. Skeith, Private searching on streaming data, J. Cryptol. 20 (4) (2007) 397–430.
[117] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, K.-L. Tan, Private queries in location based services: Anonymizers are not necessary, in: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 121–132, http://dx.doi.org/10.1145/1376616.1376631.
[118] M. Akgün, A.O. Bayrak, B. Ozer, M.Ş. Sağıroğlu, Privacy preserving processing of genomic data: A survey, J. Biomed. Inform. 56 (2015) 103–111.
[119] M. Naveed, E. Ayday, E.W. Clayton, J. Fellay, C.A. Gunter, J.-P. Hubaux, B.A. Malin, X. Wang, Privacy in the genomic era, ACM Comput. Surv. 48 (1) (2015) 1–44.
[120] M.M.A. Aziz, M.N. Sadat, D. Alhadidi, S. Wang, X. Jiang, C.L. Brown, N. Mohammed, Privacy-preserving techniques of genomic data—a survey, Brief. Bioinform. 20 (3) (2019) 887–895.
[121] E. Ayday, J.L. Raisaro, U. Hengartner, A. Molyneaux, J.-P. Hubaux, Privacy-preserving processing of raw genomic data, in: J. Garcia-Alfaro, G. Lioudakis, N. Cuppens-Boulahia, S. Foley, W.M. Fitzgerald (Eds.), Data Privacy Management and Autonomous Spontaneous Security, Springer, Berlin, Heidelberg, 2013, pp. 133–147.
[122] E. Shi, T.H. Chan, E. Rieffel, R. Chow, D. Song, Privacy-preserving aggregation of time-series data, in: Proc. NDSS, Vol. 2, Citeseer, 2011, pp. 1–17.
[123] M. Kaur, V. Kumar, A comprehensive review on image encryption techniques, Arch. Comput. Methods Eng. 27 (1) (2020) 15–43.
[124] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, K. Ren, A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing, IEEE Trans. Inf. Forensics Secur. 11 (11) (2016) 2594–2608.
[125] M. Shen, G. Cheng, L. Zhu, X. Du, J. Hu, Content-based multi-source encrypted image retrieval in clouds with privacy preservation, Future Gener. Comput. Syst. 109 (2020) 621–632.
[126] J. Yu, B. Zhang, Z. Kuang, D. Lin, J. Fan, iPrivacy: Image privacy protection by identifying sensitive objects via deep multi-task learning, IEEE Trans. Inf. Forensics Secur. 12 (5) (2016) 1005–1016.
[127] K. Barker, M. Askari, M. Banerjee, K. Ghazinour, B. Mackas, M. Majedi, S. Pun, A. Williams, A data privacy taxonomy, in: A.P. Sexton (Ed.), Dataspace: The Final Frontier, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 42–54.
[128] B.B. Mehta, U.P. Rao, Privacy preserving unstructured big data analytics: Issues and challenges, Procedia Comput. Sci. 78 (2016) 120–124.
[129] N. Victor, D. Lopez, J.H. Abawajy, Privacy models for big data: A survey, Int. J. Big Data Intell. 3 (1) (2016) 61–75.
[130] F. Prasser, J. Eicher, H. Spengler, R. Bild, K.A. Kuhn, Flexible data anonymization using ARX—Current status and challenges ahead, Softw. - Pract. Exp. 50 (7) (2020) 1277–1304.
[131] ARX - Data Anonymization Tool, ARX https://arx.deidentifier.org/ (consulted in September 2020).
[132] M. Terrovitis, D. Tsitsigkos, Amnesia Dashboard, Institute for the Management of Information Systems, https://amnesia.openaire.eu/amnesia/ (consulted in September 2020).
[133] sdcTools, Joinup https://joinup.ec.europa.eu/solution/sdctools-tools-statistical-disclosure-control/about (consulted in September 2020).
[134] μ-ARGUS, https://github.com/sdcTools/muargus (consulted in September 2020).
[135] τ-ARGUS, https://github.com/sdcTools/tauargus (consulted in September 2020).
[136] sdcMicro: Statistical Disclosure Control Methods for Anonymization of Data and Risk Estimation, sdcMicro https://cran.r-project.org/package=sdcMicro (consulted in September 2020).
[137] Realrolfje, Anonimatron, https://realrolfje.github.io/anonimatron/ (consulted in September 2020).
[138] Aircloak, https://aircloak.com (consulted in September 2020).
[139] K. El Emam, Guide to the de-Identification of Personal Health Information, CRC Press, 2013.
[140] E.P.I. Center, Concerning the re-identification of consumer information, 2017, https://epic.org/privacy/reidentification (consulted in September 2020).
[141] R. Mendes, J. Vilela, On the effect of update frequency on geo-indistinguishability of mobility traces, in: Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, ACM, 2018, pp. 271–276.
[142] J. Krumm, A survey of computational location privacy, Pers. Ubiquitous Comput. 13 (6) (2009) 391–399.
[143] K. Basu, V. Debusschere, S. Bacha, Residential appliance identification and future usage prediction from smart meter, in: IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society, IEEE, 2013, pp. 4994–4999.
[144] S. Sami, Y. Dai, S.R.X. Tan, N. Roy, J. Han, Spying with your robot vacuum cleaner: Eavesdropping via lidar sensors, in: Proceedings of the 18th Conference on Embedded Networked Sensor Systems, 2020, pp. 354–367.
[145] J. Lee, C. Clifton, How much is enough? Choosing ε for differential privacy, in: International Conference on Information Security, Springer, 2011, pp. 325–340.
[146] K.-P. Yee, Aligning security and usability, IEEE Secur. Priv. 2 (5) (2004) 48–55.
[147] R. Kainda, I. Flechais, A. Roscoe, Security and usability: Analysis and evaluation, in: 2010 International Conference on Availability, Reliability and Security, IEEE, 2010, pp. 275–282.
[148] M. Alshamari, A review of gaps between usability and security/privacy, Int. J. Commun. Netw. Syst. Sci. 9 (10) (2016) 413–429.
[149] M. Al-Rubaie, J.M. Chang, Privacy-preserving machine learning: Threats and solutions, IEEE Secur. Priv. 17 (2) (2019) 49–58.
[150] H. Bae, J. Jang, D. Jung, H. Jang, H. Ha, S. Yoon, Security and privacy issues in deep learning, 2019, arXiv:1807.11655.
[151] E. Hesamifard, H. Takabi, M. Ghasemi, R.N. Wright, Privacy-preserving machine learning as a service, Proc. Privacy Enhanc. Technol. 2018 (3) (2018) 123–142.
[152] I.C.S. Institute, U. of California-Berkeley, Teaching Privacy, http://teachingprivacy.org (consulted in September 2020), 2017.
[153] A. Boutet, S. Gambs, Inspect what your location history reveals about you: Raising user awareness on privacy threats associated with disclosing his location data, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 2861–2864, http://dx.doi.org/10.1145/3357384.3357837.