A Semantic-Preserving Differentially Private Method for Releasing Query logs

David Sánchez[a1], Montserrat Batet[b], Alexandre Viejo[a], Mercedes Rodríguez-García[c], Jordi Castellà-Roca[a]

[a]*CYBERCAT-Center for Cybersecurity Research of Catalonia, UNESCO Chair in Data Privacy, Department of Computer Science and Mathematics, Universitat Rovira i Virgili, Av. Països Catalans, 26, 43007 Tarragona, Spain*

[b]*Internet Interdisciplinary Institute (IN3), Universitat Oberta de Catalunya, Av. Carl Friedrich Gauss, 5, Parc Mediterrani de la Tecnologia, 08860 Castelldefels (Barcelona), Spain*

[c]*School of Engineering, University of Cadiz, Av. de la Universidad 10, 11519 Puerto Real, Cadiz, Spain*

Abstract

Query logs are of great interest for data analysis. They allow characterizing user profiles, user behaviors and search habits. However, since query logs usually contain personal information, data controllers should implement appropriate data protection mechanisms before releasing them for secondary use. In the past, the anonymization of query logs was tackled from the perspective of statistical disclosure control and by relying on privacy models such as *k*-anonymity, which do not scale well with the high dimensionality and dynamicity of query logs. To offer better privacy protection, some authors have recently embraced the robust privacy guarantees of ε-differential privacy. However, this comes at the cost of limiting the number and types of analyses that can be made on the protected queries. To tackle this issue, in this paper we propose a privacy protection method for query logs that joins the flexibility and convenience of privacy-preserving data releases with the strong privacy guarantees of ε-differential privacy. Moreover, to retain the analytical utility of the protected query, we have put special care in capturing, managing and preserving the semantics of the queries during the protection process. The empirical experiments we report show that our method produces differentially private query logs that are more useful for analysis than related works.

Keywords: differential privacy, query logs, data utility, user profiling.

[1] Corresponding author. Address: Departament d’Enginyeria Informàtica i Matemàtiques. Universitat Rovira i Virgili.
Avda. Països Catalans, 26. 43007. Tarragona. Spain
Tel.: +34 977559657; Fax: +34 977 559710;
E-mail: david.sanchez@urv.cat.

1. Introduction

Among the activities performed by Internet users, the most common action is probably the use of web search engines (WSEs) as the entry point for finding information and surfing around the Web. When a WSE receives a search query, it looks for the required information among billions of indexed web pages, and returns the corresponding search results in the form of ranked documents. During this process, the WSE automatically stores the submitted query (i.e., the keywords) and some related metadata (e.g., date of the query, an identifier of the sender or the search result selected by the sender). The recorded search queries together with their metadata are stored in files named *query logs* [9].

The analysis of these query logs allows the characterization of user profiles, user behaviors and search habits. This information represents the cornerstone of marketing techniques such as *Behavioral Targeting*, which are applied by website publishers and advertisers to increase the effectiveness of advertisements [10]; or the so-called *Search Engine Marketing*, which employs user profiles and related search data to improve keyword advertising campaigns and extract market tendencies, among others [21]. Marketing companies and other third parties are well aware of the usefulness of query logs and they buy these data from WSE operators in order to exploit them for their own economic purposes [35]; specifically, constructing user profiles by means of the data stored in query logs has been acknowledged as a relevant topic by the scientific community and has received significant attention [20, 46].

Despite its benefits, the exploitation of query logs does not come without cost. In particular, query logs may contain:

i) Pieces of data that may allow re-identifying the individuals who have generated them; for instance, queries related to the user's address, sex, occupation or age may be aggregated to enable univocal re-identifications [43].

ii) Information related to very sensitive topics, such as religion, sexual preferences or medical conditions, among others [3]; these may be linked to individuals in case of re-identification and may be used for discriminatory purposes (e.g., in health insurance or credit applications) [35].

The uncontrolled disclosure of query logs, thus, poses a serious privacy threat to the users of WSEs. In order to mitigate this threat, query logs should be anonymized before releasing them to untrusted parties for secondary use. The anonymization method should minimize the probability of re-identifying individuals, while ensuring that the protected data are still useful for analysis; specifically, this would allow data controllers to generate anonymous but accurate enough user profiles that may still be exploited for a variety of purposes.

1.1. Related work

The anonymization of query logs has already been tackled in the literature from different perspectives.

The most straightforward schemes use two kind of approaches:

i) Removal of queries that are considered to be specially privacy-threatening [8], such as *infrequent queries* [1] or queries that contain *direct identifiers* of the individuals (e.g., social security numbers, names or addresses) [3].

ii) Mix of queries among senders that share the same interests in a way that the users never appear linked to their own original queries in the anonymized query logs [29].

These techniques do not rely on privacy models, which constitute the means to provide formal and beforehand privacy guarantees on the protected query logs. As a result, they fail to balance the trade-off between privacy protection and data utility preservation, which is the main goal behind privacy-preserving data releases.

Among the systems enforcing privacy models, those based on *k-anonymity* [36] are the most common ones [4, 7, 28]. They build groups of *k* users and aggregate or generalize their query logs in order to make them indistinguishable and, thus, lower the probability of re-identifying the individuals to, at most, 1/*k*. Nevertheless, *k*-anonymity was designed for structured databases with a limited number of attributes and, therefore, its performance is severely hampered with high dimensional data such as query logs [2]. Moreover, because *k*-anonymity relies on making data sets more uniform, it is designed to work with static data. As a result, it does not easily support privacy-preserving updates, either adding more queries to an existing user or adding new users' logs.

ε-Differential privacy, on the other hand, is a privacy model that was originally proposed in [14] to be used in *interactive settings*. In these settings, an anonymization mechanism sits between an analyst submitting (statistical) queries on the sensitive data set and a trusted database holder that stores the clear data and provides sanitized (differentially private) answers. *ε-Differential privacy* guarantees that the protected response to a certain statistical query is very similar to the output obtained for the same query when the data of an individual are removed (or modified) from the data set. In this way, the presence or absence of the data of a certain individual in the data set does not have a significant influence on the protected results, thus, preventing attackers from performing re-identification inferences. This privacy guarantee is stronger than the one provided by *k*-anonymity; however, it comes at the cost of severely restricting the type, number and/or accuracy of the statistical queries submitted to the database.

Some researchers have used ε-differential privacy to protect query logs. In [22] the authors define an interactive scenario in which the entities that send requests to a remote database holding the query logs are classified according to a certain "requester profile". Once a requester is assigned to a certain profile, the proposed scheme applies a privacy policy to the requester, which is based on adding a certain level of noise to the answers in order to fulfill differential privacy. The goal of this proposal is to enable stakeholders to perform general privacy-preserving data analyses; that is, it is capable of answering statistical queries such as "number of people searching for Starbucks over a period of one month", but it cannot answer to individual-level queries.

Other approaches are based on releasing a differentially private data structure (e.g., a query click graph), in which the queries performed by the users and their corresponding clicks are aggregated without their individual attribution. Proposals that follow this approach generate a protected output that keeps intact the statistics needed to implement the typical uses of query logs, such as query suggestions, inferring spelling corrections or classifying queries according to a set of topics [23, 27]; or to compute common information retrieval operations such as "ordering of top results" or "re-ranking of query results" [49, 50]. However, by discarding the users that have performed the queries, these methods cannot be used to build user profiles.

In [18, 19] the authors propose a mechanism that preserves the schema of the input query logs (i.e., the specific queries performed by the users) but distorts the query counting in a differentially private way. However, a strict enforcement of differential privacy should consider *all* the queries that could be sent by the user and not only those that have been actually performed. Moreover, modifying query counts without considering the meaning of the queries may significantly distort the semantics of the query log and, therefore, the user profile that may be derived from it.

1.2. Contributions and plan of this paper

In this paper, we propose a privacy protection method for query logs that joins the flexibility and convenience of privacy-preserving data releases with the strong privacy guarantees of ε-differential privacy. Specifically, we enforce differential privacy in the context of non-interactive releases of query logs; that is, on the contrary to the related works discussed above, we produce a protected version of the original query logs with the same cardinality and granularity. Thanks to the fact that we are producing differentially private query logs, we do not limit their posterior uses (including per-individual analyses, such as user profiling). For such purpose, we tackle several challenges, which include the configuration of a differentially private mechanism that can cope with the discrete and nominal nature of queries, and the implementation of tools to deal with natural language textual inputs. Special care has been put on retaining the analytical utility of the protected query logs by: i) capturing, managing and preserving the semantics of the queries during the protection process; and ii) allowing to balance the trade-off between the data protection guarantees and the utility of the protected query logs according to the kind of analysis to be carried out.

The rest of the paper is organized as follows. Section 2 provides background on differential privacy and reviews the main approaches enabling differentially private data releases. Section 3 characterizes the main features of query logs from a privacy-preserving perspective and proposes a general method to protect them that preserves the semantics of the queries while enforcing the robust privacy guarantees of *ε*-differential privacy. Section 4 reports the empirical evaluation of our method and compares its ability to preserve the utility of query logs in user profiling against related works. The final section presents the conclusions and proposes several lines of future research.

2. Background on differential privacy

As introduced above, *ε*-differential privacy was initially proposed as a privacy model for interactive settings [14], that is, to protect the results of statistical queries performed to a server storing a sensitive database. In this setting, a differentially private mechanism (*sanitizer*) sits between an entity that submits queries and the database server that answers them.

Definition 1 (ε-Differential privacy). *A randomized function k gives ε-differential privacy if, for any pair of databases X_1, X_2 that differ in a single record, and all $S \subset$ Range(k), it holds.*

$$\Pr(k(X_1) \in S) \le \exp(\varepsilon) \times \Pr(k(X_2) \in S) \qquad (1)$$

Definition 1 implies that differentially private responses should be similar for any pair of databases differing in one record (i.e., the data of one individual). As a result, the protected outcomes do not give insights on the presence or absence of any specific individual in the database with a probability depending on *ε*. This provides a very strong privacy guarantee for low values of *ε*.

ε-Differential privacy is usually attained by adding noise to the protected outcomes. For example, the response $f(X)$ of a user query f computed on the data set X is protected by adding random noise, $Y(X)$, to $f(X)$; then, the result, $k(X)=f(X)+Y(X)$, is returned. Noise $Y(X)$ is usually generated with a Laplace distribution with zero mean and $\Delta(f)/\varepsilon$ scale parameter, where $\Delta(f)$ is the *L_1-sensitivity* of f, that is, the maximum variation of the query result for any pair of data sets differing in one record. The lower the ε and/or the larger the sensitivity, the more noise is needed to fulfill ε-differential privacy.

Laplacian noise can be added to continuous numerical data. For discrete numerical data it is possible to use the geometric mechanism, which offers a discrete probability distribution alternative to the continuous Laplace distribution [16]. For categorical data, the exponential mechanism proposed in [26] probabilistically chooses the output of a discrete function according to the ε parameter and a quality criterion on the input values.

On the contrary to k-anonymity, which lacks composability of successively anonymized data, ε-differential privacy offers the following composition properties:

Theorem 1 (Sequential composition). *A sequence of n sanitizers, each one providing ε-differential privacy, applied over non-disjoint (i.e., correlated) data provides (n* ε)-differential privacy.*

Theorem 2 (Parallel composition). *A sequence of n sanitizers, each one providing ε-differential privacy, applied over disjoint (i.e., independent) data provides ε-differential privacy.*

That is, when providing answers to successive queries on correlated data (e.g., queries on the same value or on non-independent attribute values of the same record/individual), due to sequential composition, more distortion should be applied to attain a certain level of ε-differential privacy; specifically, noise with parameter ε/n, where n is the number of queries, should be added to attain *ε-differential privacy*. On the other hand, for successive queries on disjoint data (e.g., records of independent individuals), the amount of required distortion is constant, thanks to parallel composition.

Recently, differential privacy has also been applied to non-interactive settings, which consist on releasing a protected version of the original data set [39], much like k-anonymity does. Non-interactive settings offer more flexibility than the interactive one because protected data releases can be used to answer *any* number of queries of *any* type. Focusing on this last scenario, two approaches to differentially private data releases can be identified:

i) The most common approach consists in releasing differentially private histograms (i.e., exhaustive responses to counting queries) [11, 19, 47, 48]. In this case, the data distribution is approximated (discretized) by partitioning the data domain and counting the number of records in each partition. In order to make it independent of the data set, the partitioning is fixed beforehand for the data domain. Then, discrete noise is added to the record counting of each histogram bin in order to fulfill ε-differential privacy. Histogram-based approaches do not scale well for high dimensional data because of the large number of bin combinations to consider (i.e., one for each combination of bins of each attribute), which results in exponential runtime and large data distortion [41].

ii) Very recently, some authors have proposed methods for differentially private data releases that perturb the actual attribute values [39, 41], rather than the record counting. Because no assumptions on the data uses are made, these methods tend to preserve the utility of the protected data better than the approaches based on protecting histograms [41]. Moreover, they also scale better (i.e., (quasi-)linearly) with high dimensional data because only the values appearing in the data set (not in the whole domain) are considered during the noise addition process. Finally, unlike histogram-based approaches, value perturbation methods preserve the cardinality of the data set and the granularity of the values. Because these benefits fit well with the needs of utility-preserving releases of query logs, we employ this approach in the method we propose in the next section.

3. Releasing differentially private query logs

As discussed in Section 1.1, most of the works applying differential privacy to query logs are framed in the interactive setting. This setting assumes specific data uses and only supports aggregated statistics and counting queries on the whole data set. However, crucial uses of query logs, such as *user profiling*, require analyzing data at a record level, e.g., to characterize the distribution of interests of an anonymous individual from her query log; and this can only be done with non-interactive data releases. In fact, non-interactive data releases are more flexible and independent of the posterior uses than the interactive setting because the whole (protected) data in their original format and, ideally, with similar cardinality and granularity, are made available for unconstrained data analyses. In the following, we detail the method we propose for releasing differentially private query logs.

3.1. Particularities of query logs

The privacy models proposed in the literature have been mostly applied to structured statistical databases, in which each record details a fixed set of single-valued attributes of an individual [13]. However, protecting query logs conveys additional challenges:

- First, query logs, which are also referred as *set-valued data* [43], do not constitute well-defined sets of attributes. Their cardinality significantly varies from one user to another, and their dimensionality is much larger (hundreds or thousands of queries per user) than the number of attributes usually found in relational databases. This challenges privacy models and statistical disclosure control methods intended for structured databases, which assume that records have the same number of single-valued attributes. Moreover, privacy models such as *k*-anonymity are known to perform poorly with high dimensional data due to the *curse of dimensionality*: when the data contain a large number of attributes that may be considered *quasi-identifiers*, it becomes difficult to anonymize the data without an unacceptably high amount of information loss [2].
- From the privacy protection perspective, query logs challenge the standard classification of attributes into *quasi-identifiers* or *confidential* attributes. *Quasi-identifiers* are attributes that external entities may know of an individual (e.g., sex, age, occupation, etc.) and whose combination may be unequivocally associated to a record in a data set containing such attributes. Quasi-identifiers should be protected in order to avoid re-identifications and, also, to avoid disclosing the confidential attributes (e.g., medical conditions) stored in the same record. *Confidential* attributes (e.g., medical conditions, income, etc.) are sensitive values that, as long as they cannot be associated to an identity, can be left in clear to preserve the analytical utility of the data. Because queries lack structure (i.e., we cannot classify them as identifying or confidential data), it is commonly assumed that queries are both *quasi-identifying* and *confidential* [4, 43]. Due to this assumption, which is much more stringent than the *fixed* characterization of attributes in relational databases, *all* the queries in a query log should be subjected to protection.
- Whereas attributes in a statistical database are usually numerical (e.g., age) or categorical with a finite set of categories (e.g., sex or occupation), queries are free text nominal data (e.g. “new Apple phones”). This introduces additional challenges because, on the one hand, free text data management requires linguistic tools. On the other hand, whereas the analytical utility of numeric and categorical data depends on their statistical and distributional features (which should be preserved during the protection process), the utility of nominal data depends on the preservation of their meaning or *semantics* [12, 25].

Fortunately, query logs also present some features that make re-identifications more difficult:

- Because query logs are dynamic data, unique combinations of (quasi-identifying) queries that may enable re-identifications may no longer be unique in future updates (i.e., when adding new queries of existing users or when adding query logs of new users). Therefore, in the event of an attacker knowing a set of features of an individual, the fact that they may unequivocally match with the queries of a certain user in the released query logs does not mean that this still holds in future updates.
- Whereas the attributes in a database usually refer to non-modifiable and unambiguous personal features (e.g., demographic data), queries commonly refer to circumstantial needs or ambiguous facts that do not *define* the individuals. Therefore, it is less probable for an attacker to know a large set of features that can be used to re-identify subjects.
- In many occasions, statistical databases contain exhaustive samples of a population, such as all the patients of a hospital or all the inhabitants of an area. Therefore, if the quasi-identifiers of an individual are unique in the database and those are known by an attacker, re-identification is unequivocal. Query logs, on the other hand, usually correspond to much wider and less bounded populations, from which the released query logs contain only a fraction of those populations. In the most extreme case of a general-purpose search engine like Google, the population would correspond to all the user of the Internet. With non-exhaustive samples, the uniqueness of a set of quasi-identifying queries does not imply uniqueness in the population and this precludes unequivocal re-identifications [40].

In the following, we tackle the challenges and exploit the particularities detailed above to design a method to enable differentially private releases of query logs.

3.2. Differentially private query logs

Let $U=\{u_1,...,u_n\}$ be a data set of query logs, where $u_i=\{q_{i1},...,q_{im}\}$ is the query log of the i^{th} user; notice that the cardinality of u_x may be different for any x in $1..n$. In the context of differential privacy, we can think of a release of query logs as the collected answers to successive identity queries on U, where $I_x(U)$ returns all the queries in u_x. Then, the differentially private version of the data set, U^ε, can be generated by providing ε-differentially private answers to the set of $I_x(U)$ for all x in $1...n$; that is, by distorting the actual queries in the query logs in a differentially private way. In this manner, we maintain both the cardinality of the data set (on the contrary to differentially private approaches based on distorting counts [11, 19]) and the granularity of the queries (on the contrary to approaches enforcing k-anonymity via generalizations [28]).

For the sake of simplicity, let us first assume that the cardinality of u_x for all x in $1...n$ is 1; that is, each user performed a single query. To attain ε-differential privacy to the answer of $I_x(U)$ for a certain $x=j$ we can apply a differentially private mechanism with the *L_1-sensitivity* of $I_x()$ and ε as parameters. We discuss the specific mechanism in the next section. Once we have the ε-differentially private query for a single user j, $I_j^{\varepsilon}(U)$, we can generate the whole differentially private data set, U^{ε}, by executing successive $I_x^{\varepsilon}(U)$ for the remaining users.

Proposition 1. *U^{ε}, with $|u_x|=1$ for all x in 1..n, satisfies ε-differential privacy.*

Proof. Successive differentially private identity queries $I_x^{\varepsilon}(U)$ on U refer to different query logs. Because WSEs use browsing identifiers (e.g., cookies, IPs or login sessions) to compile queries and associate them to individual users, different query logs correspond to different users accessing to the WSE from different computers, IPs and/or networks. Therefore, users are independent, and their query logs are *disjoint* among each other. According to Theorem 2 (*parallel composition*), the composition of ε-differentially private disjoint data (i.e., ε-differentially private query logs of independent users resulting from successive identity queries $I_x^{\varepsilon}(U)$) provides ε-differential privacy. Therefore, U^{ε} satisfies ε-differential privacy. □

Let us now generalize the procedure to query logs with *several* queries for each user. For the sake of simplicity, let us first assume that the number of queries per query log, m, is the same for all the users; we consider later the case of query logs with different cardinalities. In this case, we define $I_x(U)$ as the collected answers to successive requests for m queries in u_x. Let $I_{xy}(U)$ be the function that returns the y^{th} query from u_x. Then, we can formulate $I_x(U)=(I_{x1}(U), \ldots, I_{xm}(U))$, and the differentially private data set U^{ε} can be generated by collecting differentially private answers to $I_{xy}(U)$, for all x in $1...n$ and all y in $1...m$.

Proposition 2. *U^{ε}, with $|u_x|=m$ for all x in 1..n, satisfies $m\varepsilon$-differential privacy.*

Proof. We have explained above how to generate ε-differentially private answers to single queries of individual users, $I_{xy}^{\varepsilon}(U)$. Because the queries of a query log u_x correspond to the same user x, when $|u_x|=m$ with $m>1$, we should consider the worst-case scenario in which the m queries of the user are correlated (i.e., not disjoint). According to Theorem 1 (*sequential composition*), the collection of m differentially private queries on a single user, $I_x^{\varepsilon}(U)=(I_{x1}^{\varepsilon}(U), \ldots, I_{xm}^{\varepsilon}(U))$, only fulfills $m\varepsilon$-differential privacy, rather than ε-differential privacy. Therefore, U^{ε}, which is generated by collecting differentially private answers to $I_{xy}(U)$ for all x in $1...n$ and for all y in $1...m$, only fulfills $m\varepsilon$-differential privacy. □

To maintain a privacy level of ε per query log, we should proportionally lower the privacy parameter applied to each query (and, thus, increase the level of query distortion) according to the total privacy budget defined by ε; that is, ε/m or $I_{xy}^{\varepsilon/m}(U)$. According to Proposition 2, $U^{\varepsilon/m}$ fulfills ε-differential privacy for individual query logs (up to m released queries per user), and also for all query logs (due to parallel composition, as stated in Proposition 1).

If the cardinalities of the query logs are different (i.e., $|u_x|=m_x$ for each x in $1..n$) and we know them at the protection phase (e.g., if we are protecting static data), the privacy parameter can be set on user-basis to ε/m_x. This results in a more accurate protection and better data utility than a fixed criterion of maximal cardinality.

Proposition 3. *$U^{\varepsilon/mx}$, with $|u_x|=m_x$ for all x in $1..n$, satisfies ε-differential privacy.*

Proof. According to Proposition 2, the proof is immediate replacing ε by ε/m_x for each u_x. □

Moreover, on the contrary to *k*-anonymous releases of query logs, which can only deal with static data, our method supports dynamic data (e.g., data streams) and updating already released query logs:

- If a new query log u_{n+1} of a new user is added to the data set, as stated in Proposition 1, parallel composition applies. Thus, u_{n+1} should be protected in the same manner (i.e., with the same privacy parameter) as the already released ones.
- If a new query q_{im+1} of an existing user i is to be published, as stated in Proposition 2, sequential composition applies. In this case, we have two options. The first one is to keep some privacy budget for these additional queries. To do so, we can set a *max* number of queries per user and protect each query according to a ε/max privacy parameter, that is, $I_{xy}^{\varepsilon/max}(U)$, so that we can still fulfill ε-differential privacy. The second option consists on lowering the privacy guarantee proportionally to a number a of additional queries, which will only offer $a\varepsilon$-differential privacy.

3.3. A semantic differentially private mechanism for query logs

To enforce the protection schema depicted in the previous section we need a mechanism capable of providing the differentially private outcomes of $I_{xy}^{\varepsilon}(U)$; that is, a method that distorts the actual values of the queries instead of their counts or distributions. As discussed in Section 2, differential privacy can be straightforwardly enforced for numerical data by adding Laplacian noise to the outcomes. However, because queries are discrete and nominal, we must employ an alternative mechanism.

To enforce differential privacy over discrete functions, we can use the exponential mechanism [26]. Given a function with discrete outputs *t* ($I_{xy}(U)$, in our case), the mechanism chooses a differentially private output that is close to the optimum (i.e., the original query, in our case) according to: i) the input data *X*; ii) a quality criterion $\sigma(X,t)$ (the higher the better); iii) the ε parameter; and iv) the *L_1-sensitivity* of σ. Each output is then associated with a selection probability Pr(*t*), which grows exponentially w.r.t. the quality criterion:

$$\Pr(t) \propto \exp\left(\frac{\varepsilon \times \sigma(X,t)}{2 \times L_1 - sensitivity(\sigma)}\right). \qquad (2)$$

Because probabilities depend exponentially on the quality criterion, we should carefully define this criterion so that it is consistent with the features of the data that we want to preserve. In this respect, because queries are textual nominal entities and most of their uses exploit their meaning to perform inferences [20, 44, 46], their utility should be understood from a semantic perspective; that is, the better the query semantics are preserved, the more useful the protected query logs will be [4]. In the following, we detail how query semantics can be captured, managed and considered during the enforcement of the exponential mechanism.

The semantics of the queries (e.g., "new Apple phone") are given by the *concepts* they refer to (e.g., Apple phone). Therefore, to preserve these semantics after the protection process, the differentially private mechanism should replace the original queries by others that refer to concepts with *similar* meanings (e.g., Huawei tablet). Hence, we propose using the *semantic similarity* between the concept referred by the input query and each candidate for replacement as the quality criterion σ in the exponential mechanism. With this, given an input query q_i, the optimal output will be a query q_o that is a synonym of q_i, which is the most semantically similar one and, thus, the one that maximizes the quality criterion σ; whereas the outputs that are close to the optimum will be the queries q_o with a meaning very similar to that of q_i.

To semantically manage queries, we first need to *map* them to the *concepts* they refer. We rely on taxonomically-structured knowledge bases [17] to capture and exploit the semantics of these concepts. Formally, a taxonomy τ is an upper semilattice $\leq$ on a set of concepts with a top element (*root*). According to the relative positions of two concepts c_1 and c_2 in τ (that would correspond to queries q_1 and q_2, respectively), we can numerically quantify the resemblance between their meanings according to their *semantic similarity* [5], that is, *sim*(c_1,c_2). More details on this are given in Section 3.5. Therefore, we propose using *sim*(c_o,c_i), as the *quality criterion* σ in eq. (2). This is employed to compute the probability of choosing q_o (corresponding to concept c_o) as a differentially private output to q_i (corresponding to concept

c_i). The *L_1-sensitivity* of $sim(\cdot,\cdot)$ in the denominator of eq. (2), is the maximum variation of $sim(\cdot,\cdot)$ when changing q_i for any other one in the domain of queries. For any data set, this corresponds to the variation of $sim(\cdot,\cdot)$ when going from the two most similar queries to the two least similar. Formally, being *min_sim* and *max_sim* the minimum and maximum similarities between any pair of concepts to which queries may refer to, the *L_1-sensitivity* of $sim(\cdot,\cdot)$ is |*max_sim-min_sim*|. Notice that, by definition, *max_sim* corresponds to the similarity between two identical concepts (i.e., synonym queries). By applying these notions to eq. (2), we can rewrite the exponential mechanism so that it provides differentially private and semantically preserving outputs:

$$\Pr(q_o) \propto \exp\left(\frac{\varepsilon \times sim(c_i, c_o)}{2\times |\,max\_sim - min\_sim\,|}\right), \qquad (3)$$

where c_i is the concept to which q_i refers to, and c_o is the concept to which q_o refers to; q_o is created by using the lexical labels (i.e., synonym terms) associated to the concept c_o in the underlying taxonomy.

In the following sections we detail: i) how to *map* plain textual queries to concepts in a taxonomy; and ii) which function is the most appropriate to measure the *semantic similarity* between concepts from the perspective of differential privacy.

3.4. Mapping queries to concepts

Queries, being free text nominal data, are challenging to manage. A query may correspond to an individual concept (e.g. *"*phone*"*), to a specialization (e.g. *"*Apple phone*"*), to a concatenation of several concepts within a syntactically coherent sentence (e.g. *"*stores offering the new Apple phones*"*), or even to a raw list of terms (e.g. *"*phones Apple store new*"*). These last types of *complex queries* are usually performed by the users of web search engines [38].

To map *queries* to the concepts they refer to in a semantically consistent way, we apply a pipeline of linguistic analyses. First, each query q_i is morpho-syntactically analyzed to extract *semantic units*, which are pieces of text that refer to unique concepts. Syntactically, semantic units correspond to nouns or noun phrases (NPs). These are sets of words in which at least one of them is a noun. To extract NPs, we apply the following natural language processing techniques: *sentence detection*, *tokenization*, *part-of-speech (POS) tagging* and *syntactic parsing*. As a result, a query q_i (e.g. *"*stores offering the new Apple phones*"*) is split into several ones q_i={q_{i1},…, q_{il}}, each one corresponding to a NP sentence (e.g. *"*stores" and "the new Apple phones*"*). In this manner, *complex* queries with several NPs are treated as several *individual* queries for anonymization purposes because we consider that each of them contributes to the semantic characterization of the user.

Afterwards, we map the resulting *individual* queries to the concepts they refer to in the taxonomy τ by terminologically matching query strings and concept labels. Due to the morpho-syntactical variability of the NPs referring to the same concept (e.g. *"*Apple phone*"*, *"*Apple phones*"*, *"*new Apple phone*"*, etc.), we apply additional analyses to detect equivalent references of the same concept. First, *stop words* (i.e., domain independent words with very general meanings such as determinants, prepositions and adverbs) are removed from the NPs (e.g., "the new Apple phones" → "new Apple phones"). After that, both NPs in the queries and concept labels in the taxonomy are *stemmed* [33] to remove derivational affixes, such as plurals, of the same root word (e.g., "new Apple phones" → "new Apple phone"). Finally, in the case that a NP composed of several words cannot be matched with any concept label in the taxonomy, we try simpler forms of the NP by progressively removing adjectives/nouns starting from the one most on the left (e.g., *"*new Apple phone*"* → *"*Apple phone*"*). With this strategy, we improve the recall of the conceptual mapping while maintaining the core semantics of the queries.

In principle, the taxonomy should be large and detailed enough to contain most of, if not all, the NPs obtained after the linguistic analysis. However, if a query refers to newly coined terms (e.g., technological products) or named entities (e.g., proper nouns) that are not found in the taxonomy, this query is discarded because we cannot protect it in a meaningful way and releasing it "as is" would cause a significant disclosure risk.

By applying the exponential mechanism detailed in the former section, the concepts to which the queries have been mapped (e.g., "Apple phone") are replaced by new concepts (e.g., "Huawei tablet"). Then, the distorted queries associated to such concepts are constructed by undoing the mapping operations in inverse order; specifically, the nouns/adjectives that were removed during the mapping are re-added (e.g., "Huawei tablet" → "new Huawei tablet"), plurals and derivative forms are recovered (e.g., "Huawei tablet" → "Huawei tablets"), stop words are added, and the individual queries corresponding to a complex query are concatenated (e.g., "web sites offering the new Huawei tablets"). With this process we aim at generating distorted queries that look realistic because they follow the same (complex) syntactical construction of the original ones.

3.5. Semantic domain and semantic similarity

By means of the semantically-grounded exponential mechanism depicted in Section 3.3, we replace each q_i by a semantically similar query that fulfills differential privacy. The queries q_o that we use as candidates for replacement are the linguistic labels of the concepts c_o that belong to the *semantic domain* of the input query q_i ($D(q_i)$); that is, $D(q_i)$ encompasses the concepts to which q_i may refer to. For example, in a medical search engine such as PubMed, $D(q_i)$ would encompass medical concepts, whereas in a general-purpose search engine such as Google, $D(q_i)$ would encompass any possible concept of any domain. Since we rely on taxonomies to gather candidates for replacements and to compute the semantic similarity between concepts, we define $\tau(D(q_i))$ as the *minimum taxonomy* that includes all the concepts in $D(q_i)$. In this way, we restrict candidates for replacement to those concepts that strictly belong to $D(q_i)$. Formally, $\tau(D(q_i))$ is the taxonomy encompassing all the concepts that are taxonomic specializations of the *Least Common Subsumer* of $D(q_i)$, $LCS(D(q_i))$, including itself.

$$\tau(D(q_i)) = \bigcup \{c_i \mid LCS(D(q_i)) \geq c_i\}, \tag{4}$$

where $LCS(D(q_i))$ is the most specific concept that subsumes all the concepts of $D(q_i)$. A concept c_i subsumes a concept c_j, i.e., $c_i \geq c_j$, if c_i is a generalization (i.e., taxonomic ancestor) of c_j, or c_i and c_j are the same concept.

In the exponential mechanism detailed in Section 3.3, the probability of selecting a candidate q_o as replacement for q_i is a function of the *semantic similarity* between the concepts they refer to (c_o and c_i, respectively). The semantic similarity $sim(c_1,c_2) \rightarrow \mathbb{R}$, is a function mapping a pair of concepts to a real number that quantifies the resemblance between their meanings according to the semantic evidences gathered from a knowledge source [5]. A similarity measure well-suited to achieve semantically-preserving differentially private outputs should have the following features [41]:

1) It should accurately differentiate concepts in the taxonomic neighborhood of the input concept. In this way, those concepts that share more semantics with the input concept are considered more similar and, thus, are more likely to be selected by the exponential mechanism.
2) It should have a low numerical sensitivity to outlying concepts. These are the concepts that define the *minimum similarity* lower bound that, in turn, defines the *L_1-sensitivity* in eq. (3). With this, we can achieve low *L_1-sensitivity*, which contributes to increase the probability of selecting the optimal outcomes in the exponential mechanism.

Regarding the first point, the accuracy of a semantic similarity measure depends on the techniques and the knowledge bases exploited to perform the semantic assessments [5]. Among the measures relying on taxonomies, feature-based measures tend to provide the most accurate results [37]. These quantify the semantic similarity according to the number of taxonomic ancestors shared and not shared among the concepts to compare. Regarding the second point, the sensitivity to outlying concepts depends on the way in which semantic evidences are quantified. Many classical measures [34] are linearly proportional to the amount of semantic evidences observed in the taxonomy. As a result, the similarity values between outlying concepts (that define the sensitivity of the quality criterion in the exponential mechanism) are significantly smaller than those between more central concepts. This leads to a relatively high sensitivity. More recent methods evaluate similarities in a non-linear way, which implicitly weight the contribution of the semantic evidences as they become more specific [24, 32, 37]. Among these, the measures that aggregate similarities in a logarithmic way are the best suited, because they reduce the relative numerical distances associated to outlying concepts. The feature-based similarity measure in [37] fulfills this criteria and, therefore, we propose using it as quality criterion in eq. (3). Formally, this measure computes the similarity between two concepts c_1, c_2 in $\tau(D(q_i))$ as the inverse logarithmic ratio between the number of non-common taxonomic ancestors of the two concepts divided (for normalization) by their total number of ancestors:

$$sim(c_1,c_2)=1-\log_2\left(1+\frac{|S(c_1)\cup S(c_2)|-|S(c_1)\cap S(c_2)|}{|S(c_1)\cup S(c_2)|}\right), \qquad (5)$$

where $S(c_x)$ is the set of taxonomic ancestors (or *subsumers*) of c_x in $\tau(D(q_i))$, including itself.

The use of eq. (5) as quality criterion in the exponential mechanism can be tuned to increase the chance of selecting the best (i.e., the most similar) replacements. This will contribute to improve the preservation of data utility. To do so, we consider the posterior uses of differentially private query logs, specifically, in user profiling. Profiling algorithms define a set of general topics (such as *Science*, *Sports*, *Health*, *Society*, etc.) and construct the profile of a user according to the topic distribution resulting from categorizing her queries into such topics [38]. Preserving the *topic distribution* is, therefore, crucial in this type of data use. With this goal in mind, in the following, we define a quality criterion based on eq. (5) that increases the probability of replacing an input query q_i for another one q_o that belongs to the *same* topic.

Formally, let $P=\{p_1, \ldots, p_r\}$ be the set of topics defined by the profiling algorithm, and $\rho(c_x)\to P$ the function that categorizes a concept c_x (corresponding to a certain query q_x) into one of the categories in P. We define the quality criterion as follows:

$$sim'(c_1,c_2)=\begin{cases} sim(c_1,c_2) & if\ \rho(c_1)=\rho(c_2) \\ 0 & otherwise \end{cases}. \tag{6}$$

By using eq. (6) instead of eq. (5) as quality criterion in the exponential mechanism, we prioritize the replacements that maintain the topic distribution and that, therefore, contribute towards preserving the utility of the query logs; specifically, with this quality criterion we have that:

- The probability of a query replacement belonging to the topic of q_i depends on the semantic similarity vs. q_i (eq. (5)).
- The probability of replacements belonging to a topic different to that of q_i is minimum (proportional to exp^0).

Formally, the exponential mechanism with eq. (6) as quality criterion is:

$$\Pr(q_o)\propto \exp\left(\frac{\varepsilon\times sim'(c_i,c_o)}{2\times max\_sim}\right). \tag{7}$$

Notice that, according to eq. (6), min_sim=0 when $\rho(c_1)\neq\rho(c_2)$; therefore, the L_1-*sensitivity*=|*max_sim-min_sim*| in the denominator is now |*max_sim-0*|=*max_sim*.

This new L_1-*sensitivity* is, in most cases, larger than that of eq. (3). In principle, a larger L_1-*sensitivity* may produce larger distortions. However, as we will evaluate in the empirical experiments, we expect to compensate this larger sensitivity with the better suited probabilities resulting from eq. (6).

3.6. Balancing data protection and utility preservation

With the differentially private mechanism detailed above, the trade-off between the level of protection and the level of (semantic) utility preservation depends on: i) the ε parameter (i.e., the lower it is, the more distorted the protected outcomes become); and ii) the domain $D(q_i)$ of the queries to be protected. In fact, $D(q_i)$ defines two aspects of the exponential mechanism that directly influence the accuracy of the outcomes: i) the set of concepts (from $\tau(D(q_i))$) that can be used as replacements of q_i; and, therefore, ii) the *max_sim* and *min_sim* constants that define the L_1-*sensitivity*. The broader $D(q_i)$ is, the larger the number of marginal concepts in $\tau(D(q_i))$; and these are the ones that negatively influence the accuracy of the differentially private outcomes because:

1) As the number of marginal concepts increases, the chance of selecting them as replacements in the exponential mechanism also increases. Because marginal concepts

are semantically far from the central concepts in the domain, using the former as replacements of the latter will significantly hamper data semantics.

2) The more marginal the concepts in $\tau(D(q_i))$ are, the smaller *min_sim* is and, thus, the larger the L_1*-sensitivity* becomes. Notice that *max_sim* is constant for any domain because it corresponds to the similarity between two identical concepts.

Therefore, the definition of $D(q_i)$ has a crucial influence on the (semantic) utility preservation of the protected outcomes. For example, if q_i=*flu*, we may define $D(q_i)$ to cover *any respiratory disease*, *any disease* or *any entity* (being a disease or not). With the latter domains, the number of marginal concepts (i.e., those semantically distant from q_i=*flu*) will increase; and this will also increase the probability of choosing a poor replacement. Moreover, the sensitivity of the semantic similarity measure used as quality criterion will be larger because the concepts will be less similar in a domain encompassing any possible *disease* than a domain encompassing just *respiratory diseases*.

If we consider that the users may perform queries of any topic to the WSE, then $D(q_i)$ should cover all domains of knowledge. In such case, $\tau(D(q_i))$ should encompass any concept of any domain. This is the case of general-purpose knowledge bases such as WordNet [15] or ODP[2], which have been extensively used in the past to semantically manage WSE queries [4, 38, 44]. This scenario offers the best protection: by enforcing differential privacy with such broad $D(q_i)$, we guarantee that external entities cannot make unequivocal inferences on the topics referred by the protected queries, because the original queries may have been replaced by other queries from any other possible domain. However, due to the same reason, this will severely distort the distribution of query topics.

To improve the utility of the results, we propose defining separate and more specific domains for each q_i. This is more consistent with the way in which profiling algorithms characterize users from their queries [38], that is, according to the distribution of topics resulting from categorizing user queries into those topics. By applying the same notion to the protection of query logs, we propose defining beforehand a set of general topics $P=\{p_1, \ldots, p_r\}$ (e.g., those used in the profiling process), and narrowing $D(q_i)$ for each q_i to the topic p_j to which q_i belongs. For example, if q_1=*tennis* and ρ(*tennis*)=*Sports*, then $D(q_1)$=*Sports*, if q_2=*flu* and ρ(*flu*)=*Health* then $D(q_2)$=*Health*, and so on. Formally, this implies that the concept c_i to which q_i refers to is a specialization of the *root* concept of the taxonomy $\tau(D(q_i))$, that is, $c_i \leq root(\tau(D(q_i)))$. In this way, the candidates q_o to replace q_i will better preserve the semantics of q_i. For example, if q_i is related to *Sports*, q_o will also be related to *Sports*, because the latter is gathered from the

[2] Open Directory Project: http://dmoz-odp.org/

concepts below the root node of the taxonomy modelling the topic *Sports*. Moreover, the *L_1-sensitivity* of the similarity measure in the exponential mechanism will also be reduced, because *min_sim* will be larger in taxonomies modelling narrower domains.

It is important to note that, by defining independent domains for each query, the differentially private outcomes will provide more information to external entities (i.e., data analysts, but also attackers): they will be able to discern the general topic p_j to which the original queries refer to (e.g., *Sports*, *Health*, *Science*, etc.). In this case, the protected outcomes only provide a guarantee of non-unequivocal inferences within each of the domains defined by the topics in P, but not across topics. For example, a protected query about *Sports* reveals that the original query also referred to *Sports*, even though the specific sport cannot be unequivocally inferred due to the uncertainty added by the differentially private mechanism. Although one may argue that revealing this information increases the risk, the particularities of query logs (i.e., dynamicity, non-exhaustive samples of populations, etc.) makes re-identifications unfeasible in practice. Moreover, because all the queries are subjected to protection, in the advent of a re-identification, the remaining (potentially confidential) queries of the user will only disclose the general topics in P to which they refer to. For example, an attacker may learn the presence of a health condition for a certain user, but not the actual condition. In this way, we prevent confidential *attribute disclosure*, which is the most harmful threat from the perspective of data privacy.

On the other hand, from the perspective of data analysis, the approach we propose is very convenient: if the queries are protected according to the same set of general topics P that are used to profile the users, then, the protected outcomes will perfectly preserve the utility of the queries. In other words, building user profiles from the protected outcomes will be as accurate as doing so from the original queries.

In any case, the balance between the level of protection and the level of utility preservation can be configured according to the granularity of the topics used during the protection process. This is formalized as follows. Let $P=\{p_1, \dots p_r\}$ be the set of general topics to which the queries may refer to, each one modelled in a separate taxonomy. Then, for a given q_i, we have that $D(q_i) \in P$ and that $c_i \leq root(\tau(D(q_i))$. Then, the scope of replacements considered by the exponential mechanism can be constrained as follows:

$$\Pr(q_o) \propto \exp\left(\frac{\varepsilon \times sim(c_i, c_o)}{2 \times |\, max\_sim - min\_sim \,|} \right) \quad \forall c_o \in \tau(D(q_i)), \tag{8}$$

4. Empirical experiments

In this section, we evaluate the capability of our method of retaining the utility of the protected query logs in user profiling, which is the most common per-individual use of query logs [20, 46]. Given a set of topics $P=\{p_1, \ldots, p_r\}$ (e.g., *sport*, *science*, *heath*, *society*), we construct the profile Π_i of each user i as the topic distribution resulting from categorizing each query q_{ix} (e.g., "tennis club") into P via the function $\rho(q_{ix})\rightarrow P$ (e.g., "tennis club" → *sport*). This process is the cornerstone of most profiling algorithms [44, 45]. Because user profiles are defined as distributions of topics (e.g., <*sport*,3>, <*science*,1>, <*health*,10>, <*society*,5>), we evaluate the utility of the differentially private query logs according to the divergence between the distributions of profiles constructed from the original queries (Π_x with x in $1..n$) and those obtained from the protected queries (Π_x^*). To compare topic distributions, we employ the well-known *Jensen–Shannon divergence* (or *IRad*). This measure, which is based on the *Kullback–Leibler divergence*, quantifies the divergence between two probability distributions. The main advantages of the *JSD* measure over the *Kullback–Leibler divergence* (which has been used in the past to evaluate profile protection methods [30, 31]) are that the former is symmetric and it is not unbounded. This facilitates the interpretation of the results. The *Jensen–Shannon divergence* (*JSD*) is computed as follows:

$$JSD(\Pi_x \,||\, \Pi_x^*) = \frac{1}{2} D_{KL}(\Pi_x \,||\, M) + \frac{1}{2} D_{KL}(\Pi_x^* \,||\, M), \tag{9}$$

where $M=½(\Pi_x + \Pi_x^*)$ and the Kullback–Leibler divergence (D_{KL}) is computed as:

$$D_{KL}(\Pi_x \,||\, M) = \sum_i T(i)\log_2(\Pi_x(i)/M(i)), \tag{10}$$

where $\Pi_x(i)$ is the probability of the i^{th} topic in Π_x. When the base 2 logarithm is used, the *JSD* ranges $0 \leq JSD(T||T^*) \leq 1$.

Once we have calculated the *JSD* for each user (original vs. protected version), we measure the utility loss of the protected query logs as the arithmetic average of the *JSD* for all the profiles Π^* generated from such protected logs.

$$Utility\_loss(\Pi^*) = \frac{1}{n}\sum_{x=1}^{n} JSD(\Pi_x \,||\, \Pi_x^*), \tag{11}$$

As evaluation data, we used 1,000 real query logs randomly taken from the AOL query logs released in 2006 [3]. Notice that the number of logs does not have a significant influence the results (other than smoothing the effect of the random distortion) because, as stated in Proposition 1, query logs are protected independently. For user profiling, we have selected those queries that belong to four general topics typically used in profiling methods: *Health*, *Science*, *Sports* and *Society* [44, 45]. The average number of queries in each query log is around 100, which is large enough to characterize the users [44] but small enough to keep the distortion applied to each query under control (see Proposition 3). To semantically interpret the queries and to map them to the appropriate topics, we used WordNet as knowledge base. WordNet [15] is a general-purpose semantic electronic repository. The lexical units (or words) encompassed by WordNet are grouped into sets of synonyms, named *synsets*, each one representing a distinct concept. Synsets are interlinked by means of semantic relations such as hyperonymy/hyponymy (e.g. *cat* is a *mammal*), meronymy (*wheel* is part of a *car*), etc. The result is a network of meaningfully related words that can be exploited to interpret the semantics of concepts. Queries have been mapped to concepts in Wordnet by following the procedure detailed in Section 3.4.

We have evaluated two versions of our method. The first one uses eq. (5) as (semantic) quality criterion in the exponential mechanism. We named this version *semantic quality criterion 1* (*SQC1*). The second one uses eq. (6) as quality criterion, which prioritizes query replacements within the same topic. We named this version *semantic quality criterion 2* (*SQC2*).

As stated in Section 3.6, the privacy guarantees and the utility preservation offered by our method depend on the ε parameter and the domain $D(q_i)$ of the queries to be protected. The former has been set within the range commonly used in differentially private data releases [39]: [0.1 … 10]. For the latter, we can distinguish different scenarios according to the topics considered during the protection and profiling processes:

- As stated in section 3.6, if the domains $D(q_i)$ used during the protection process, e.g., *health*, *science*, *society*, *sport*, are the same as those used afterwards to profile users (P), the protected outcomes will perfectly preserve the distribution of topics. Indeed, because queries can only be replaced by other queries within the same topic, the topic distributions and, hence, the user profiles are not altered. In this case, we can expect the *JSD* to be exactly zero for all query logs, regardless the quality criteria. We denote this scenario $D(q_i)=P$.
- If the query domains are *more general* (which result in more robust privacy guarantees) than the topics used to profile the users, we should expect non-zero *JSD*. The *JSD* will increase proportionally to the difference between the specificity of the query domains and the topics used for profiling. To test this scenario, which we denote $D(q_i)>P$, we have defined the following settings (see Figure 1):
    - *Setting A*: the semantic domains of the input queries q_i ($D(q_i)$) may correspond to one of the four main topics mentioned above: *Health*, *Science*, *Sports* and *Society.* These topics have been mapped to the following WordNet concepts: *disease*, *scientific discipline*, *sport*, *social event*. For example, if q_i=*asthma* then

D(*asthma*)=*disease*, and the queries q_o we use as replacements of q_i will be taxonomic specializations of *disease*. On the other hand, user profiles are generated according to 19 topics that are specializations of the former four ones; these involve more specific health-related topics such as *respiratory diseases* or *genetic diseases*, scientific disciplines such as *natural science* or *social science*, sport types such as *contact sports* or *water sports*, and recreational events such as *shows* or *competitions*.

- *Setting B*: the domain of the input queries is unique and encompasses all the queries. In WordNet, the most specific concept that generalizes *disease*, *scientific discipline*, *sport*, *social event* is *Abstraction*, which we used as $D(q_i)$ for all q_i. This setting offers the most robust protection, because the protected queries do not unequivocally reveal any concrete topic. Like in the previous setting, the profiles are constructed according to the 19 topics mentioned above.
- *Setting C*: this setting uses *Abstraction* as domain for all the queries (like in *Setting B*), but profiles users according to the four general topics: *disease*, *scientific discipline*, *sport* and *social event*.

One may also conceive a third scenario in which query domains are *more specific* than those used to profile users (i.e., $D(q_i)<P$). By definition, this scenario will also result in zero *JSD*, as in $D(q_i)=P$, but with weaker privacy guarantees. Therefore, it will be less preferable.

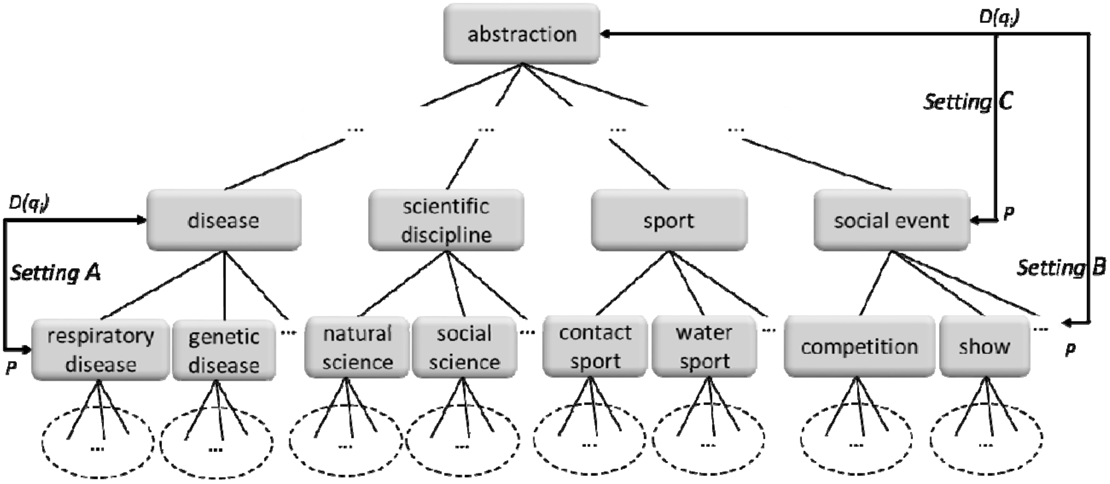

Figure 1. Fragment of the taxonomies (extracted from WordNet) used to define the query domains ($D(q_i)$) and the profile topics *P*. Arrows depict the evaluation settings.

To contextualize our results and compare the performance of our approach against related works, we have implemented two baseline methods:

- The first one aims at assessing the contribution of our semantic quality criteria to the preservation of data utility. For such purpose, this method does not consider query semantics at all during the application of the exponential mechanism. Instead of semantics, it uses a quality criterion that evaluates queries as equal or non-equal strings (eq. (12)). We named this baseline method *non-semantic quality criterion* (*NSQC*).

  $$sim(q_1, q_2) = \begin{cases} 1 & if\ q_1 = q_2 \\ 0 & otherwise \end{cases}. \qquad (12)$$

- The second one implements the most common approach to differentially private data releases (see [11, 19, 47, 48] discussed in Section 2), that is, distorting query counts rather than the actual queries. In this case, query logs are transformed into histograms, where each bin represents the number of times each query in the domain appears in the query log. Then, discrete Laplace noise is added to the counting of each histogram bin (i.e., the query counting) to fulfill differential privacy. Since we are protecting the complete query logs at once, we do not need to sequentially compose the results for each query log and we can use the same ε to distort the counting of each bin. On the other hand, as we do in our method, we can also use parallel composition to release the logs of different users. The *L_1-sensitivity* is, in this case, the maximum number of queries a user can perform (or that the data controller wants to release), which we set to the maximum number of queries of any log in the data set. We named this baseline method *histogram-based differential privacy* (*HBDP*).

All the figures reported below for all the methods are the result of averaging three consecutive runs, so that we minimize the influence of the randomization in the exponential or Laplace mechanisms.

Figure 2 depicts the *utility loss* (eq. (11)) for $D(q_i)>P$ with *Setting A* for the three quality criteria detailed above (*SQC1*, *SQC2* and *NSQC*) and the *histogram-based differential privacy* approach (*HBDP*) for ε=[0.1…10]. We also include the utility loss for $D(q_i)=P$ for all the methods.

The figures sustain our assumptions. Analyzing the results provided by the methods using the exponential mechanism we see that, on the one hand, when queries are protected according to the same set of topics that are used to profile the users ($D(q_i)=P$), the profiling accuracy is perfectly preserved (*JSD*=0 for any value of ε) for all the quality criteria (*SQC1*, *SQC2* and *NSQC*). Therefore, the user profiles generated from the protected queries are the same as those constructed from the original queries. On the other hand, when the topics used to profile the users are more specific than the query domains ($D(q_i)>P$), the profile divergence is not null and it decreases inversely proportionally to the value of ε. The degree of improvement varies according to the quality criterion and, in the best case, the semantic criteria achieves roughly 4 times lower divergence than the non-semantic one when ε increases from 1 to 10; specifically, we can see that, for $\varepsilon>1$, the semantic quality criterion *SQC2* (eq. (6)), preserves the query log utility significantly better than the *SQC1* (eq. (5)) and the *NSQC* (eq. (12)). This is due to the *SQC2* prioritizes query replacements within the same query topic (e.g., *sports*), and this contributes to maintain the topic distribution in the user profile. In fact, we can see that the more accurate query replacements resulting from the *SQC2* compensate the, a priori, higher distortion resulting from the larger L_1*-sensitivity* of the *SQC2* w.r.t. the *SQC1*. Finally, even though the non-semantic quality criterion *NSQC* provides the worst results, these are not very different to those produced by the *SQC1*. In this case, the number of profile topics is quite large (19) and, as a result, the probability of picking up a query replacement that is outside the original domain with the *SQC1* and the *NSQC* (which do not prioritize replacements within the same domain) is also large.

In comparison with the methods above, the *histogram-based method* (HBDP) provides significantly worse results. On the one hand, when $D(q_i)=P$, the utility loss is not zero (in fact, it is similar to that of the NSQC with $D(q_i)>P$); and, on the other hand, when $D(q_i)>P$, it produces the largest utility loss. Instead of distorting the queries within their query domains, this method builds histograms encompassing all the query domains considered (i.e., 4 topics when $D(q_i)>P$ and 19 topics when $D(q_i)=P$). Because the joint cardinality of these query domains is much larger than the number of queries in the logs (i.e., domains account thousands of concepts whereas query logs contain 100 queries in average), the histograms constructed by this method have a large number of bins (one per concept) and they tend to be very sparse (i.e., many bins have a query counting of 0 or 1). Due to this sparsity, the noise added to the query counts to fulfill differential privacy significantly distorts the distribution of the queries. Also, because query domains cannot be fixed to individual queries, there is a significant loss of utility when $D(q_i)=P$ because the distribution of query topics is also distorted.

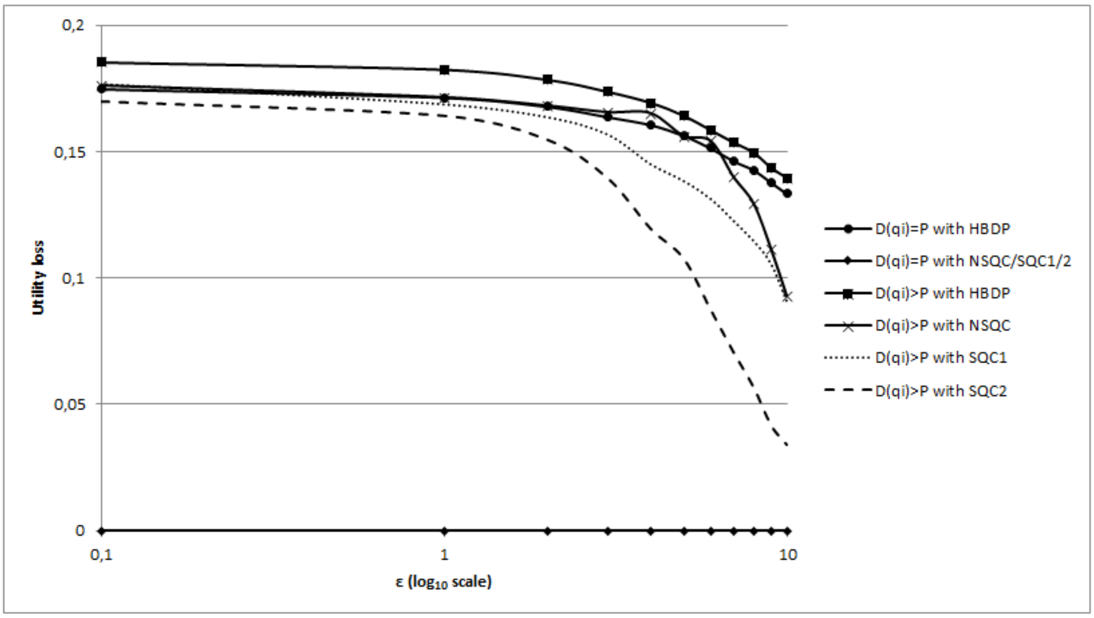

Figure 2. Utility loss for *D*(q_i)>*P* with *Setting A* with *D*(q_i)=*P*, for *ε*=[0.1…10].

Figure 3 depicts the same results for the two scenarios and *Setting B*. In this setting the set of concepts that can be used as replacements of q_i is broader than in *Setting A* because *D*(q_i) covers all the concepts that are taxonomic specializations of the general concept *Abstraction*. These include *Health*, *Sport*, *Society* and *Science*, but others, such as *Arts* or *Business*. This results in stronger privacy guarantees than in *Setting A* because, now, attackers cannot unequivocally infer the topic/domain of the protected queries. However, in turn, it results in a greater data distortion because the probability of choosing concepts that are semantically distant to q_i increases. As a result, we observe larger divergences between the user profiles built from the original and the protected queries for a given quality criterion and *ε*. Even though the relative differences between the *SQC1* and the *SQC2* are similar to those of *Setting A*, we observe a slightly worse performance with the *NSCQ* for values near *ε*=10. The inability of the latter to differentiate between semantically similar concepts belonging to the same topic from those that are semantically distant, results in more probable suboptimal replacements; and this affects negatively the profiling accuracy. Finally, the *HBDP* method produces again in the greater utility loss, which is now more than two times larger than in *Setting A*. This is expected because, in *Setting B*, the histograms built by this method are much larger because they encompass all the specializations of *Abstraction*. Therefore, the sparsity of the histograms and the relative distortion of the query distributions resulting from the noisy counts are significantly larger. This is also illustrated by the fact that the utility loss hardly decreases when increasing *ε*.

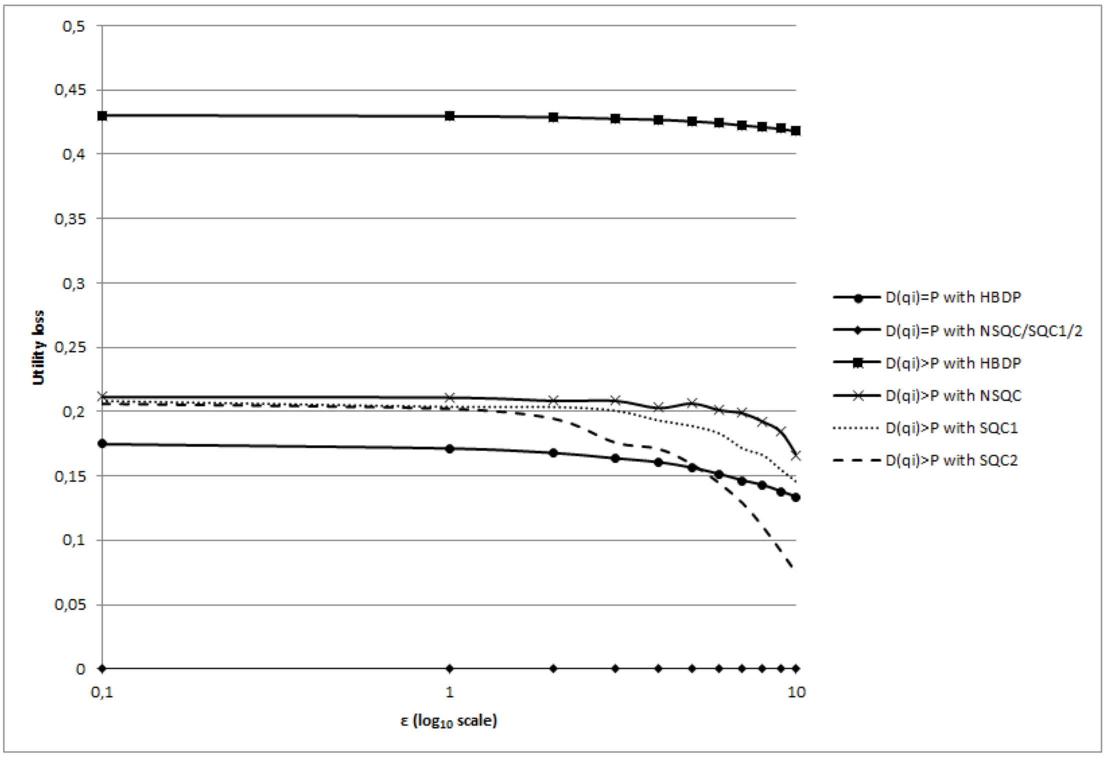

Figure 3. *Utility loss* for $D(q_i)>P$ with *Setting B* and with $D(q_i)=P$, for ε=[0.1…10].

Figure 4 depicts the same results for the two scenarios and *Setting C*. Notice that, in this case, the *Y* axis also uses a log_{10} scale due to the very large differences between the *HBDP* method and those using the exponential mechanism. In *Setting C*, the topics that are used to profile the users are more general than those used in *Settings A* and *B*. Due to the fact that the number of topics is now significantly smaller than in the former settings (19 vs. 4), the average profile divergences are also smaller, because the number of concepts covered by each topic is larger. Specifically, the probability of optimally replacing q_i by a concept of the same topic increases w.r.t. *Settings A* and *B*. Due to the same reason, the semantic criteria (*SQC1* and *SQC2*) are able to significantly improve the non-semantic one (*NSQC*), because the latter is not able to discern between the optimal replacements (within the same topic) and the large number of non-optimal ones (specializations of *Abstraction* outside the profiling topic). For these methods, we also observe that the sum of utility losses for *Settings A* and *C* roughly corresponds to those of *Setting B*. This is consistent with the taxonomic differential between query domains and profile topics of the three settings. On the other hand, the utility loss produced by the *HBDP* method for *Setting C* is roughly the same of that of *Setting B*. This is expected because, in both settings, the cardinality of the query domains (and, therefore, the size of the histograms) is the same; and this results in a proportionally similar distortion of the distribution of profile topics. As a result, we can see that the *HBDP* method does not benefit from making the profile topics more general and, therefore, causes a loss of utility that is an order of magnitude larger than that produced by the methods distorting the actual queries.

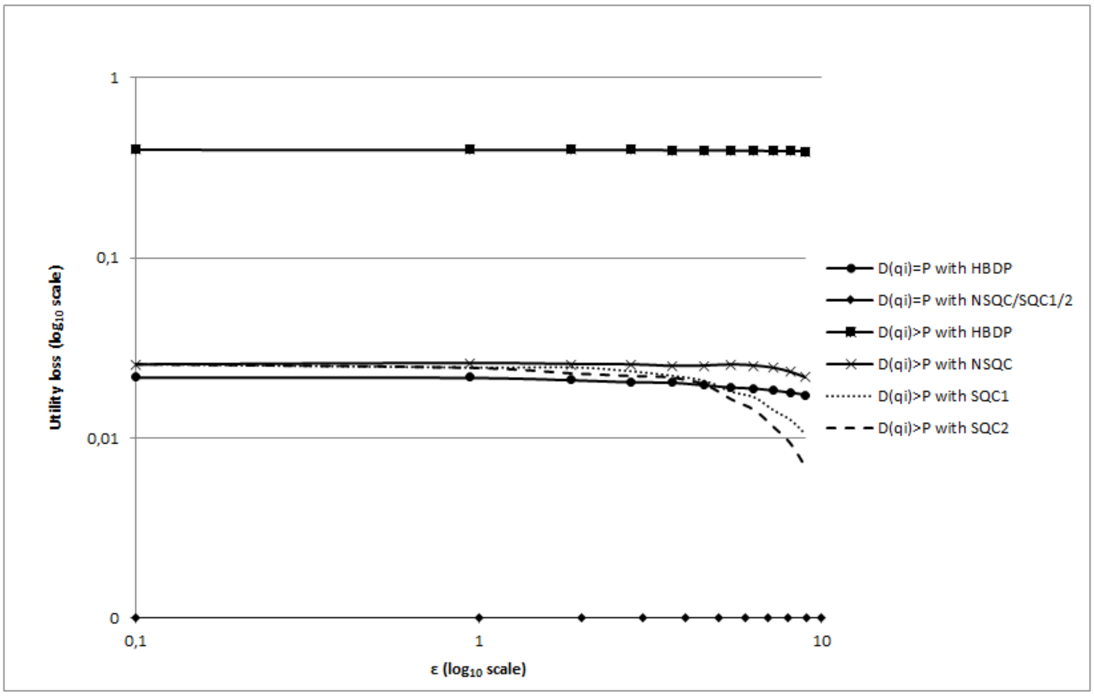

Figure 4. *Utility loss* for $D(q_i)>P$ with *Setting C* and with $D(q_i)=P$, for ε=[0.1…10].

5. Conclusions and future work

Differential privacy provides several benefits over the classic methods that have been employed to protect query logs. These benefits include more robust privacy guarantees and composability properties that are convenient to cope with incremental releases of queries and query logs. However, so far, differential privacy has been used to protect query logs mainly in interactive settings. This limits the number and type of analyses that can be executed and prevents executing per-individual analyses such as user profiling. Being these latter analyses of great interest to data controllers and third parties, in this paper we have proposed a protection method for query logs that reaps the advantages of privacy-protected data releases (i.e., flexibility and lack of constrains on the posterior data uses) and the strong privacy and composability properties of differential privacy. On the contrary to related works, our approach distorts the *actual* queries rather than their counting. In this way, we not only preserve the cardinality of the query logs, but also drive the distortion consistently with the semantics of the queries; that is, the queries used as replacements of the original ones are probabilistically chosen according to their semantic similarity and the desired level of protection. This contributes to better preserve the semantics and, therefore, the utility of the protected query logs. Moreover, in addition to being able to set the desired level of distortion (via the ε parameter), our approach enables to balance the trade-off between the protection guarantees and the level of utility preserved by the protected query logs. Concretely, by making the protection consistent with the posterior analyses (i.e., profiling topics), we are able to significantly (and, in some cases, perfectly) preserve the accuracy of query logs, as shown in the empirical experiments. The only requirement to benefit from this possibility is that the data provider must know beforehand what type of profiling will be applied on the data or, at least, the level of profile granularity that is acceptable.

As future work, we plan to evaluate the utility of the results provided by our method with other user-centered data uses of query logs, such as behavioral analysis [6] or goal extraction [42]. Structured knowledge sources other than WordNet, such as ODP, YAGO or DBPedia, which may offer more detailed and finer grained taxonomies, could also be used to increase the accuracy. Finally, data preprocessing can be used to reduce the sensitivity of the quality criterion: as discussed in [41], data microaggregation, which consists on grouping similar records/logs together and replacing them by a representative value, can be used to decrease the sensitivity of the differentially private mechanism proportionally to the size of the data aggregation. The main challenge would be to adapt the microaggregation algorithm, which is intended for structured databases, to the lack of structure of query logs. Afterwards, we should also evaluate whether the benefits resulting from the lower sensitivity compensate the loss of information produced by the prior microaggregation step.

Acknowledgements and disclaimer

This work was partly supported by the European Commission (H2020-700540 project "CANVAS"), by the Spanish Government (projects TIN2014-57364-C2-R "SmartGlacis", TIN2015-70054-REDC "Red de excelencia Consolider ARES" and TIN2016-80250-R "Sec-MCloud") and by the Spanish DGT under the project "ISSUM" SPIP2017-02250. The opinions expressed in this paper are those of the authors and do not necessarily reflect the views of UNESCO.

References

[1] E. Adar, User 4xxxxx9: Anonymizing query logs, in: In Query Logs Workshop at the 16th International World Wide Web Conference -- WWW '07, 2007, pp. 1-8.

[2] C.C. Aggarwal, On k-anonymity and the curse of dimensionality, in: 31st international conference on Very large data bases ACM, Trondheim, Norway, 2005, pp. 901 - 909.

[3] M. Barbaro, T. Zeller, A face is exposed for aol searcher no. 4417749, The New York Times, (2006).

[4] M. Batet, A. Erola, D. Sánchez, J. Castellà-Roca, Utility preserving query log anonymization via semantic microaggregation, Information Sciences, 242 (2013) 49-63.

[5] M. Batet, D. Sánchez, Review on semantic similarity, in: Encyclopedia of Information Science and Technology, 2014, pp. 7575-7583.

[6] D.J. Brenes, D. Gayo-Avello, Stratified analysis of AOL query log, Information Sciences, 179 (2009) 1844–1858.

[7] C. Carpineto, G. Romano, Semantic Search Log k-Anonymization with Generalized k-Cores of Query Concept Graph, in: Proceedings of the 35th European Conference on IR Research - ECIR'13, 2013, pp. 110-121.

[8] A. Cooper, A survey of query log privacy-enhancing techniques from a policy perspective, ACM Transactions on the Web, 2 (2008) 1-27.

[9] M. Chau, X. Fang, O.R.L. Sheng, Analysis of the Query Logs of a Web Site Search Engine, Journal of the American Society for Information Science and Technology, 56 (2005) 1363-1376.

[10] J. Chen, J. Stallaert, An Economic Analysis of Online Advertising Using Behavioral Targeting, MIS Quarterly, 38 (2014) 429–449.

[11] R. Chen, N. Mohammed, B.C.M. Fung, B.C. Desai, L. Xiong, Publishing set-valued data via differential privacy, in: Proceedings of the 37th International Conference on Very Large Data Bases, 2011, pp. 1087-1098.

[12] J. Domingo-Ferrer, D. Sánchez, G. Rufian-Torrell, Anonymization of nominal data based on semantic marginality, Information Sciences, 242 (2013) 35-48.

[13] J. Domingo-Ferrer, D. Sánchez, J. Soria-Comas, Database Anonymization: Privacy Models, Data Utility and Microaggregation-based Inter-model Connections, Morgan & Claypool, 2016.

[14] C. Dwork, Differential privacy, in: Automata, Languages and Programming, Springer, 2006, pp. 1-12.

[15] C. Fellbaum, WordNet: An Electronic Lexical Database, MIT Press, 1998.

[16] A. Ghosh, T. Roughgarden, M. Sundararajan, Universally utility-maximizing privacy mechanisms, in: Proceeding of the ACM Symposium on Theory of Computing (STOC'09), 2009, pp. 351-360.

[17] N. Guarino, Formal Ontology in Information Systems, in: N. Guarino (Ed.) 1st International Conference on Formal Ontology in Information Systems, FOIS 1998, IOS Press, Trento, Italy, 1998, pp. 3-15.

[18] Y. Hong, J. Vaidya, H. Lu, P. Karras, S. Goel, Collaborative search log sanitization: toward differential privacy and boosted utility, IEEE Transactions on Dependable and Secure Computing, 12 (2015) 504-518.

[19] Y. Hong, J. Vaidya, H. Lu, M. Wu, Differentially private search log sanitization with optimal output utility, in: Proceedings of the 15th International Conference on Extending Database Technology, ACM, 2011, pp. 50-61.

[20] B. Jansen, M. Zhang, D. Booth, D. Park, Y. Zhang, A. Kathuria, P. Bonner, To What Degree Can Log Data Profile a Web Searcher? , Proceedings of the Association for Information Science and Technology, 46 (2009) 1-19.

[21] B.J. Jansen, Search log analysis: What is it; what's been done; how to do it, Library and Information Science Research, 28 (2006) 407-432.

[22] P. Kodeswaran, E. Viegas, Applying differential privacy to search queries in a policy based interactive framework, in: Proceedings of the ACM 1st international workshop on Privacy and anonymity for very large databases - PAVLAD'09, 2009, pp. 25-32.

[23] A. Korolova, K. Kenthapadi, N. Mishra, A. Ntoulas, Releasing Search Queries and Clicks Privately, in: Proceedings of the 18th international conference on World wide web - WWW'09, 2009, pp. 171-180.

[24] Y. Li, Z. Bandar, D. McLean, An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources., IEEE Transactions on Knowledge and Data Engineering, 15 (2003) 871-882.

[25] S. Martínez, D. Sánchez, A. Valls, A semantic framework to protect the privacy of electronic health records with non-numerical attributes, Journal of Biomedical Informatics, 46 (2013) 294-303.

[26] F. McSherry, K. Talwar, Mechanism design via differential privacy, in: Proceeding of Annual IEEE Symposium on Foundations of Computer Science (FOCS'07), 2007, pp. 94-103.

[27] X. Meng, Z. Xu, B. Chen, Y. Zhang, Privacy-Preserving Query Log Sharing Based on Prior N-Word Aggregation, in: Proceedings of the 2016 IEEE Trustcom/BigDataSE/ISPA, 2016, pp. 722-729.

[28] G. Navarro-Arribas, V. Torra, Tree-Based Microaggregation for the Anonymization of Search Logs, in: Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, 2009, pp. 155-158.

[29] D. Pàmies-Estrems, J. Castellà-Roca, A. Viejo, Working at the Web Search Engine Side to Generate Privacy-Preserving User Profiles, Expert Systems with Applications, 64 (2016) 523–535.

[30] J. Parra-Arnau, J.P. Achara, C. Castelluccia, MyAdChoices: Bringing Transparency and Control to Online Advertising, ACM Transactions on the Web, 11 (2017) Article No. 7.

[31] J. Parra-Arnau, D. Rebollo-Monedero, J. Forné, Measuring the privacy of user profiles in personalized information systems, Future Generation Computer Systems, 33 (2014) 53-63.

[32] G. Pirró, A semantic similarity metric combining features and intrinsic information content, Data & Knowledge Engineering, 68 (2009) 1289-1308.

[33] M.F. Porter, An algorithm for suffix stripping, in: Readings in Information Retrieval, Morgan Kaufmann Publishers Inc, San Francisco, 1997, pp. 313-316.

[34] R. Rada, H. Mili, E. Bicknell, M. Blettner, Development and application of a metric on semantic nets, IEEE Transactions on Systems, Man and Cybernetics, 19 (1989) 17-30.

[35] E. Ramirez, J. Brill, M.K. Ohlhausen, J.D. Wright, T. McSweeny, Data Brokers: A Call for Transparency and Accountability, in: Report, U.S. Federal Trade Commission, May 2014.

[36] P. Samarati, Protecting respondents identities in microdata release, IEEE Transactions on Knowledge and Data Engineering, 13 (2001) 1010-1027.

[37] D. Sánchez, M. Batet, D. Isern, A. Valls, Ontology-based semantic similarity: A new feature-based approach, Expert Systems with Applications, 39 (2012) 7718-7728.

[38] D. Sánchez, J. Castellà-Roca, A. Viejo, Knowledge-Based Scheme to Create Privacy-Preserving but Semantically-Related Queries for Web Search Engines, Information Sciences, 218 (2013) 17-30.

[39] D. Sánchez, J. Domingo-Ferrer, S. Martínez, J. Soria-Comas, Utility-preserving differentially private data releases via individual ranking microaggregation, Information Fusion, 30 (2016) 1-14.

[40] D. Sánchez, S. Martínez, J. Domingo-Ferrer, Comment on "Unique in the shopping mall: On the reidentifiability of credit card metadata", Science, 351 (2016) 1274.

[41] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, S. Martínez, Enhancing Data Utility in Differential Privacy via Microaggregation-based K-anonymity, The VLDB Journal, 23 (2014) 771-794.

[42] M. Strohmaier, M. Kröll, Acquiring knowledge about human goals from Search Query Logs, Information Processing and Management, 48 (2012) 63–82.

[43] M. Terrovitis, N. Mamoulis, P. Kalnis, Privacy-preserving anonymization of set-valued data., Proceedings of the VLDB Endowment, PVLDB, 1 (2008) 115-125.

[44] A. Viejo, D. Sánchez, Profiling social networks to provide useful and privacy-preserving web search, Journal of the Association for Information Science and Technology, 65 (2014) 2444-2458.

[45] A. Viejo, D. Sánchez, J. Castellà-Roca, Preventing automatic user profiling in Web 2.0 applications, Knowledge-Based Systems, 36 (2012) 191-205.

[46] K. Wai-Ting Leung, D. Lun Lee, Deriving Concept-based User Profiles from Search Engine Logs, IEEE Transactions on Knowledge and Data Engineering, 6 (2007) 969-982.

[47] Y. Xiao, L. Xiong, C. Yuan, Differentially private data release through multidimensional partitioning, in: Proceedings of the 7th VLDB conference on Secure data management - SDM'10, 2010, pp. 150-168.

[48] J. Xu, Z. Zhang, X. Xiao, Y. Yang, G. Yu, Differentially Private Histogram Publication, in: Proceedings of the 2012 IEEE International Conference on Data Engineering - ICDE'12, 2012, pp. 32-43.

[49] S. Zhang, H. Yang, L. Singh, Anonymizing Query Logs by Differential Privacy, in: Proceedings of the 39th Annual ACM SIGIR Conference, 2016, pp. 753-756.

[50] S. Zhang, H. Yang, L. Singh, Applying Epsilon-Differential Private Query Log Releasing Scheme to Document Retrieval, in: In the 2nd International Workshop on Privacy-Preserving Information Retrieval Workshop (PIR'15), 2015, pp. 1-4.