D3 - Final Use-Cases Specification

Cédric Eichler, Benjamin Nguyen, Sara Taki, Adrien Boiret

PROJET SENDUP
ANR-18-CE23-0010

Final Use-Cases Specification

DELIVERABLE D3

Related task	Task 1
Partner	LIFO
Redactor	Cédric Eichler
Contributors	Adrien Boiret, Cédric Eichler, Benjamin Nguyen, Sara Taki
Versioning	30/10/2023, V1: initial report

Presentation of this deliverable

The goal of the SENDUP project is to propose anonymisation mechanisms for data organized as graphs with an underlying semantic. Such mechanisms triggers updates on the database. Target databases are presented in D2 [6]. This deliverable presents, with regard to each database, the scenarios tackled within SENDUP.

Contents

Chapter 1

Introduction

This deliverable builds on D2 [6] to summarize target scenarios for SENDUP. We consider firstly a scenario were queries on an unpublished dataset are to be sanitized to prevent the inference of sensitive data. We present the considered queries, target privacy model and the utility metric we tailored for this scenario. Secondly, we discuss three sanitization procedures we aim to support the specification and execution of in order to publish datasets. As an illustration, we provide an instantiation of each procedure on a target dataset.

Chapter 2

Sanitized queries

In this chapter, we discuss scenarios related to querying unpublished sensitive data.

2.1 Dataset

The considered dataset is the Sentiment140 dataset[1]. Its schema is shown in Fig. 1.1.

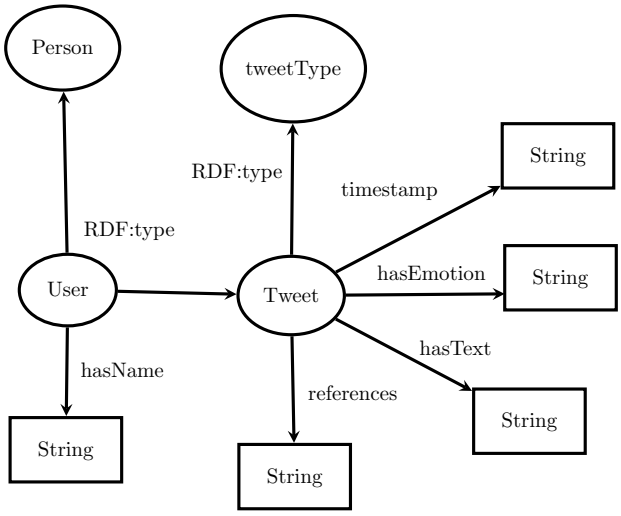

Figure 2.1: RDF schema for Sentiment140

[1] `https://www.kaggle.com/kazanova/sentiment140`

2.2 Privacy and assumption

In the considered scenario, we aim at providing formal guarantee by releasing differentially private answers to the considered queries.

Differential privacy [2] is the current go-to in terms of data anonymization, since it provides formal probabilistic guarantees against re-identification and inference without requiring assumption on the attacker's knowledge. An attacker is assumed to be an external observer analysing the output of some function(s). Intuitively, the goal of DP is to ensure that such an attacker is not able to infer (beyond a certain probabilistic threshold) which dataset among a neighbourhood was used to produce the output. The exact protection and the notion of individual contributions are defined based on the concept of neighboring (or adjacent) databases (i.e. databases at a distance of 1).

A formal definition of DP is given in the following.

Definition 1 (ϵ, δ-differential Privacy [4]) *Given $\epsilon > 0$, $0 \leqq \delta < 1$, and a distance d over $\mathcal{D}$, a randomized mechanism K: $\mathcal{D} \rightarrow \mathbb{R}$ preserves (ϵ, δ)-differential privacy if for any pair of databases D_1 and $D_2 \in \mathcal{D}$ such that $d(D_1, D_2) = 1$, and for all sets S of possible outputs:*

$$Pr[K(D_1) \in S] \ \leq e^{\epsilon} Pr[K(D_2) \in S] \ + \delta \tag{2.1}$$

where the probability is taken over the randomness of K.

Popular definition of neighborhood in tabular dataset assume that an individual contribute to a single row, and define neighboring dataset to differ by a row. We stress here that the definition of neighboring databases in the context of RDF is not trivial, and poses some of the research questions of SENDUP.

A commonly employed mechanism for achieving DP for numerical queries (i.e., functions f : $\mathcal{D} \rightarrow \mathbb{R}$) is the Laplace mechanism [3]:

Definition 2 (Laplace Mechanism [3]) *The Laplace distribution centered at μ with scale b being the distribution with probability density function*

$$h(x) = \frac{1}{2b} exp(\frac{-|x - \mu|}{b}) \tag{2.2}$$

For any $f : \mathcal{D} \rightarrow \mathbb{R}$ and $D \in \mathcal{D}$, let $K(D)$ be defined as

$$K(D) = f(D) + Lap(\frac{\Delta f}{\epsilon}) \tag{2.3}$$

where Lap $(\frac{\Delta f}{\epsilon})$ represents a random draw from the Laplace distribution centered at 0 with scale $\frac{\Delta f}{\epsilon}$. $K(D)$ satisfies ϵ-DP.

In this definition, Δf represents the global sensitivity of f. It measures the maximal variation of the query result when evaluated upon any two neighboring databases. Δf depends only on the type of query f, the considered space of databases, and the distance it is associated with (i.e., that identify neighbouring databases). It is independent of the database itself:

Definition 3 (Global sensitivity [4]) *For a function $f : \mathcal{D} \to \mathbb{R}$ and all $D_1, D_2 \in \mathcal{D}$, the l_1-sensitivity of f is*

$$\Delta f = GS_f = \max_{D_1, D_2 : d(D_1, D_2)=1} \| f(D_1) - f(D_2) \|_1 \tag{2.4}$$

where $\|\|_1$ denotes the L1 norm.

2.3 Queries

We consider the following queries:

1. Compute maximum out-degree.
2. Compute maximum label specific out-degree.
3. Count how many users tweeted more than 25 tweets.
4. Count the number of users a specific user has referenced.

Their implementation in SPARQL is available at `https://github.com/sarataki/dp-projection-queries`.

These queries provide example of classical structural analysis (e.g. 1), as well as analysis taking into account the type and semantic of the dataset. Interestingly, the global sensitivity of these queries (for any known distances) is infinite. A naive application of the Laplacian mechanism would therefore produce purely random noise.

Chapter 3

Publishing anonymous databases

In this chapter, we discuss scenarios related to the sanitization of datasets prior to their publication. Two datasets are considered: the travel dataset as a motivating example and the Sentiment140 datasets for experimental evaluation.

3.1 Travel dataset

We briefly recall herein the travel dataset structure and semantic. It has nodes for relevant entities, people and travels, whose attributes are an identifier. It also has nodes for every literal describing informations on those entities, e.g. last name, first name and address for people, date and destination for travels. We do not differentiate nodes representing entities or literals.

Its edges describe both relations between entities, e.g. "this person participated in this travel", represented by an edge of attribute ‘‘`attends`’’, but also relations between entities and their information, e.g. "this person's name is in this literal", represented by an edge of attribute ‘‘`name`’’. Typing falls within this second case e.g. "this node is a person" or "this literal is a city", represented by an edge of attribute ‘‘`type`’’.

An example of such a database is provided in Fig. 2.1. In this instance, id105 (named Miller) attended travel id207 to Paris for professional reasons.

In this dataset, we consider the names to be direct indentifier, travels to be quasi identifiers, and the destination of personal travels to be sensitive.

3.2 Privacy objectives

We aim at providing an sanitization engine expressive enough to support intricate procedures. We target in particular procedures guaranteeing Local Differ-

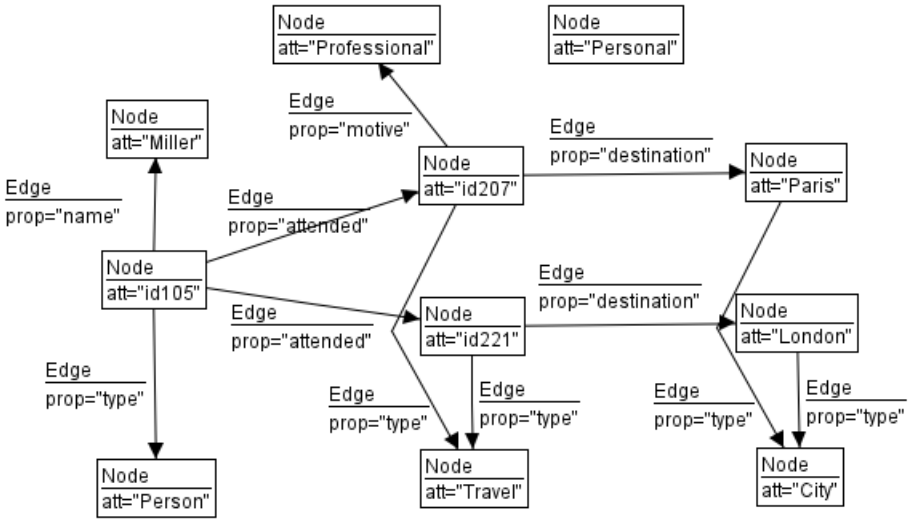

Figure 3.1: Running example: instance of a database

ential privacy, anatomization, and pseudonymisation. Example of applications of such procedures are described thereafter.

3.2.1 Local Differential Privacy

Local differential privacy [5] emerges as an alternative approach to differentialy privacy that does not rely on the presence of any trusted third-party data curator.

Definition 4 (Local differential privacy (Duchi *et al.*) [1]) *Let χ be a set of possible values and Y the set of noisy values. A mechanism $\mathcal{M}$ is ϵ-locally differentially private (ϵ-LDP) if for all $x, x' \in \chi^2$ and for all $y \in Y$ we have*

$$Pr[\mathcal{M}(x) = y] \leq e^{\epsilon} \times Pr[\mathcal{M}(x') = y]$$

We want to provide plausible deniability with regard to the relation destination between personal travels and cities. To do so, we want to randomize this relation for every personal trip specifically, to preserve privacy, with a bias towards correct answers to preserve utility. This corresponds to guaranteeing local differential privacy on trips with a motive edge leading to personal. More precisely, we want to modify the database such that querying it to output the destination of personal trips would be locally differentially private.

The procedure must be expressive enough to provide precision on the relations that need privacy; in the example it is not necessary to randomize the destination of professional trips.

3.2.2 Anatomization

Thouvenot et al. [7] adapted the anatomization approach originally used in the relational data model to the context of the RDF data model. In the context of knowledge graphs (or RDF graphs), anatomization involves breaking the relationships between the QIDs and their SAs. Instead of generalizing or suppressing entity QIDs, anatomization alters the graphs structure by introducing additional nodes and edges. The retaining of the original QID values without any transformation maintains the correlation and consequently facilitates a high-quality data analysis of the published anonymized data.

We assume that names are direct identifiers, professional trips ('s id) are quasi-identifiers, and their destination is sensitive. Here we assume that an attacker has no knowledge on personal trip.

Therefore, we want to 1) suppress persons' name and 2) for professional trips, obfuscate the relation destination between travels and cities. This can be done, by grouping trips in certain cities together (e.g. Paris, Bordeaux, Toulouse all grouped in the more nebulous group France) and rerouting the destination edges towards those groups rather than a precise value. This would means that we want to apply anatomization [7] where the destination attributes of travels is considered sensitive and travels with attribute motive set to professional are quasi-identifiers.

3.2.3 Pseudonymisation

Pseudonymisation is one of the most basic form of sanitization. It consists in removing direct identifiers. We assume the attribute of a kind of node to be a direct identifier and wish to replace all such nodes by a blank.

In the example, we consider the id of nodes of type person to be a direct identifier.

Bibliography

[1] Duchi, J.C., Jordan, M.I., Wainwright, M.J.: Local privacy and statistical minimax rates. In: 2013 IEEE 54th Annual Symposium on Foundations of Computer Science. pp. 429–438. IEEE (2013)

[2] Dwork, C.: Differential privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II. Lecture Notes in Computer Science, vol. 4052, pp. 1–12. Springer (2006)

[3] Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Theory of cryptography conference. pp. 265–284. Springer (2006)

[4] Dwork, C., Roth, A., et al.: The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science **9**(3–4), 211–407 (2014)

[5] Kasiviswanathan, S.P., Lee, H.K., Nissim, K., Raskhodnikova, S., Smith, A.: What can we learn privately? SIAM Journal on Computing **40**(3), 793–826 (2011)

[6] project team, S.: D2 specification of data and databases. Tech. rep., INSA Centre Val de Loire (2023)

[7] Thouvenot, M., Curé, O., Calvez, P.: Knowledge graph anonymization using semantic anatomization. In: 2020 IEEE International Conference on Big Data (Big Data). pp. 4065–4074. IEEE (2020)