A Declarative Approach to Linked Data Anonymization

Position paper

R. Delanaux\textsuperscript{1} \quad A. Bonifati\textsuperscript{1} \quad M-C. Rousset\textsuperscript{2,3} \quad R. Thion\textsuperscript{1}

\textsuperscript{1}LIRIS-CNRS, Université Claude Bernard Lyon 1

\textsuperscript{2}LIG, Université Grenoble-Alpes

\textsuperscript{3}Institut Universitaire de France

January 23, 2018
Linked Open Data and why anonymize it

Connected structured datasets are freely accessible...

- E.g.: public transportation
- How do we prevent re-identification?

2 complementary aspects

→ Anonymize sensitive data...
→ ...presented as LOD graphs

Figure: Linked Open Data cloud in 2017 (http://lod-cloud.net/)
Current background

Existing anonymization techniques are not fit for the LOD context

- Adapted to relational databases: $k$-anonymity and derived methods
- Focused on statistical integrity: Differential privacy ...

What can we do with LOD data?

- Existing data modeling and querying standards: RDF graphs, SPARQL queries
- Usage is different for LOD than for relational DB (w.r.t. statistics, for example)
- ⇒ We need techniques to keep data integrity, and adapted to LOD graphs
Summary

**Figure:** A classic problem: balancing data privacy and data utility

![Diagram showing the balance between privacy and utility with options for total deletion, update or partial deletion, slight modifications, and no modification.](image-url)
Figure: A classic problem: balancing data privacy and data utility
Our positioning: a **complete, static, query-based framework** to handle constraints, operators and graph anonymization

1. *Privacy and utility policies*: objects to handle constraints from the data provider as meaningful queries

2. *Anonymization operators*: operations on RDF graphs as queries

3. *Anonymization problems*: towards complete anonymization framework and algorithms
Query-based policy specification

**Policies** model data provider constraints

- A policy is a set of queries
- 2 types of policies with opposite semantics: privacy / utility policies

**Policy semantics**

- A privacy query must **not return any result** on the anonymized graph
- An utility query must **return the same results** on the anonymized graph than on the original graph
Query-based anonymization operators

Anonymization process = Sequence of operations
- Goal: Transform the input graph into a graph satisfying both policies
  - Triple deletion, value replacement, IRI replacement...

Example: Deletion queries
- Defined using SPARQL queries
- `DELETE GPD(\bar{x}) WHERE GPW(\bar{x}, \bar{y})` FILTER C
Full-fledged anonymization system

Problems of interest to be solved using the framework:

1. **Compatibility problem**: define when are policies incompatible and impossible to satisfy at the same time

2. **Candidate anonymizations**: exhibit candidate sequences of operations

3. **Best anonymization**: find the optimal operation sequence
   - Choosing a *relevant partial order binary relation*: compare number of anonymization queries, number of edited items, Jaccard distance between original and updated graph...
Current progress and future prospects

ESWC2018 submission

- Complete framework for a deletion operator
- Graph-independent solution for the candidate anon. problem
- Implementation and experimental study (Python)

Future work

- *Extend coverage* - support more complex queries
- *Extend operators* - more subtle update operations
- *Optimization* - find & apply relevant anonymization measurements
- *Complexity* - evaluate algorithms and their efficiency
Acknowledgements

This Ph.D. thesis is made possible by the A.R.C. 6 program and the doctoral funding by Région Auvergne-Rhône-Alpes.