Ensuring License Compliance in Federated Query Processing

Benjamin Moreau, Patricia Serrano-Alvarado

Ensuring License Compliance in Federated Query Processing

Benjamin Moreau
Nantes University, LS2N, CNRS
OpenDataSoft
Benjamin.Moreau@ls2n.fr
Benjamin.Moreau@opendatasoft.com

Patricia Serrano-Alvarado
Nantes University, LS2N, CNRS
Patricia.Serrano-Alvarado@ls2n.fr

KEYWORDS

Licenses, Federated Query, Privacy, Linked Data, RDF, Query Relaxation

1 INTRODUCTION AND MOTIVATION

A *federated SPARQL query* can retrieve information from several RDF data sources distributed across the Linked Data.

To facilitate reuse, data owners should systematically associate licenses with resources before sharing or publishing them[3, 14]. Licenses specify precisely the conditions of reuse of data, i.e., what actions are permitted, obliged, and prohibited.

When two or more licensed data sources participate in the evaluation of a federated query, the query result must be protected by a license such that each license of involved datasets is compatible with it. A license l_i is compatible with a license l_j if a resource licensed under l_j can be licensed under l_i without violating conditions of l_j.

Unfortunately, it is not always possible to find such a license[9]. In this case, the query result should not be reused nor published. We consider that a query whose result set cannot be licensed should not be executed.

Consider datasets of LargeRDFBench[12], a benchmark with 32 queries for federated query processing. Figure 1 shows the compatibility graph of Creative Commons licenses that protect LargeRDFBench datasets. The whole set of datasets of Figure 1 cannot be queried together because there is no license compliant with the fourth licenses. In this benchmark, 16 queries produce results that cannot be licensed.

One solution to the incompatibility of licenses is to negotiate with data providers to change a conflicting license, e.g., to ask DBpedia to change their license to CC BY or CC BY-NC. But negotiation takes time and is not always possible.

A second solution is to discard datasets that are protected by conflicting licenses. However, this solution can lead to a query with an empty result set. To face this problem, we use query relaxation techniques. That is to relax triple patterns to match triples of other datasets. But the number of possible relaxed queries can be huge and the least relaxed query may produce an empty result set. In a distributed environment, verifying each relaxed query is not feasible. So the challenge is to find the least relaxed query that returns a non-empty result while limiting communication costs.

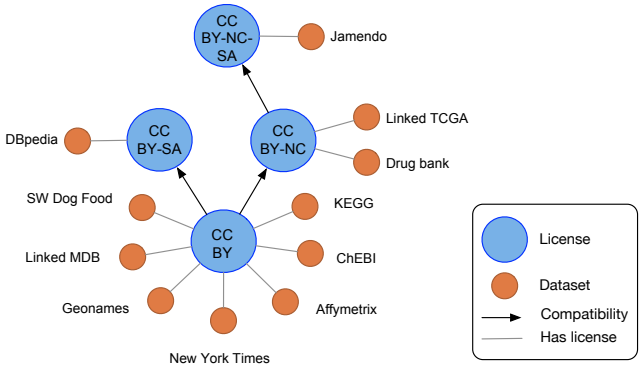

Figure 1: The compatibility graph of licenses for datasets of LargeRDFBench.

Our research question is, *given a SPARQL query and a federation of licensed datasets, how to guarantee a relevant and non-empty query result whose license is compliant with each license of involved datasets?*.

To our knowledge, there is no federated query engine that ensures license compliance. Many works focus on access control over linked data[1, 2, 6, 7, 10, 11]. These approaches do not resolve our problem because having the right to query datasets individually does not mean that it is possible to execute a federated query on them.

We propose FLiQue[1], a Federated License-aware Query processing strategy. FLiQue is designed to detect and prevent license conflicts and gives informed feedback with licenses able to protect a result set of a federated query. If necessary, it applies distributed query relaxation to propose a set of most similar relaxed queries whose result set can be licensed.

Our contributions are:

- a license-aware query processing strategy,
- an implementation of a license-aware federated query engine, and
- an experimental evaluation of our approach.

2 FLIQUE: A FEDERATED LICENSE-AWARE QUERY PROCESSING STRATEGY

We propose FLiQue, a federated license-aware query processing strategy to detect and prevent license conflicts in federated query

[1] In French, FLiQue is a homophone of *flic*, which means *cop*.

engines. FLiQue is located between the query parsing and the query optimization functions. It ensures that a query returning a non-empty licensed result set, i.e., a *candidate query*, is executed.

When the result set of a federated query cannot be licensed, we define sub-federations that avoid license conflicts. If there is no sub-federation able to produce a licensable and non-empty result set, we propose the least relaxed federated query for each sub-federation.

Compatibility graph of licenses. To know if a result set can be licensed, we need to know the license(s) with whom all licenses of datasets involved in a federated query are compatible. A compatibility graph of licenses contains a set of licenses partially ordered by compatibility. It can be defined by hand but licenses used in the Linked Data are not limited to well known licenses.

In this work, we use CaLi [8, 9], a lattice-based model for license orderings. It automatically orders a set of licenses in terms of compatibility. CaLi can provides all the licenses than should protect a result set and can also identify which licenses are in conflict.

If the result set of the original query cannot be licensed and any sub-federation can evaluate the original query, FLiQue finds the least relaxed query that can be evaluated on each sub-federation.

Query relaxation techniques. In this work, we use query relaxation using RDFS entailment rules as proposed in [5]. The idea consists of relaxation rules that use information from the ontology; these include relaxing a class to its super-class, a property to its super-property and, a term to a variable. The number of relaxed queries grows combinatorially with the number of relaxation rules, the richness of the ontology, and the relaxation possibilities of each triple pattern in the original query.

To avoid testing all relaxed queries, FLiQue executes them in a similarity-based ranking order, as in [4], until finding the candidate query. The *similarity measure* between a relaxed query and the original query is computed using statistical information about the concerned dataset, like the number of entities per class and the number of triples per property. However, the number of failing relaxed queries executed before finding the candidate query can be considerable.

FLiQue uses a strategy defined in [4]. Based on the source selection of the query engine, it identifies unnecessary relaxations generating failing relaxed queries. But, Computing similarity and identifying failing relaxed queries requires to communicate with data sources.

Data summaries. Some federated query engines, use statistics to reduce the communications to data sources during query processing, in particular in the source selection and query optimization steps.

FLiQue uses CostFed[13] source selection. It proposes data summaries, called *dataset capabilities*, containing datasets statistics such as the distinct properties with all the URI authorities of their subjects and objects prefixes. CostFed succeed in selecting relevant sources with precision while minimizing communication cost. Moreover, dataset capabilities can provide statistics to compute similarity measure.

3 CONCLUSION

In this work, we propose FLiQue, a federated license-aware query processing strategy. It ensures that a license protects the result set of any SPARQL query. To our knowledge, this is the first work that uses query relaxation in a distributed environment. Our implementation extends an existing federated query engine with our license-aware query processing strategy. The source code is available on GitHub under MIT license[2].. Our prototype demonstrates the usability of our approach. Experimental evaluation shows that FLiQue ensures license compliance with a limited overhead in terms of execution time. FLiQue is a step towards facilitating and encouraging the publication and reuse of licensed resources in the Web of Data. FLiQue is not a data access control strategy. It empowers well-intentioned data users in respecting the licenses of datasets involved in a federated query.

REFERENCES

[1] Luca Costabello, Serena Villata, and Fabien Gandon. 2012. Context-Aware Access Control for RDF Graph Stores. In *European Conference on Artificial Intelligence (ECAI)*.

[2] Alban Gabillon and Léo Letouzey. 2010. A View Based Access Control Model for SPARQL. In *International Conference on Network and System Security (NSS)*.

[3] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan F. Sequeda, Steffen Staab, and Antoine Zimmermann. 2020. Knowledge Graphs. *CoRR* abs/2003.02320 (2020).

[4] Hai Huang, Chengfei Liu, and Xiaofang Zhou. 2012. Approximating Query Answering on RDF Databases. *Journal of World Wide Web* 15 (2012).

[5] Carlos A Hurtado, Alexandra Poulovassilis, and Peter T Wood. 2008. Query Relaxation in RDF. *Journal on Data Semantics X* (2008).

[6] Yasar Khan, Muhammad Saleem, Aftab Iqbal, Muntazir Mehdi, Aidan Hogan, Axel-Cyrille Ngonga Ngomo, Stefan Decker, and Ratnesh Sahay. 2014. SAFE: Policy Aware SPARQL Query Federation Over RDF Data Cubes. In *Semantic Web Applications and Tools for Life Sciences (SWAT4LS)*.

[7] Sabrina Kirrane, Ahmed Abdelrahman, Alessandra Mileo, and Stefan Decker. 2013. Secure Manipulation of Linked Data. In *International Semantic Web Conference (ISWC)*.

[8] Benjamin Moreau, Patricia Serrano-Alvarado, Matthieu Perrin, and Emmanuel Desmontils. 2019. A License-Based Search Engine. In *Extended Semantic Web Conference (ESWC), Demo.*

[9] Benjamin Moreau, Patricia Serrano-Alvarado, Matthieu Perrin, and Emmanuel Desmontils. 2019. Modelling the Compatibility of Licenses. In *Extended Semantic Web Conference (ESWC)*.

[10] Said Oulmakhzoune, Nora Cuppens-Boulahia, Frédéric Cuppens, Stephane Morucci, Mahmoud Barhamgi, and Djamal Benslimane. 2014. Privacy Query Rewriting Algorithm Instrumented by a Privacy-Aware Access Control Model. *Annals of Telecommunications* 69 (2014).

[11] Pavan Reddivari, Tim Finin, Anupam Joshi, et al. 2007. Policy-Based Access Control for an RDF Store. In *Workshop Semantic Web for Collaborative Knowledge Acquisition (SWeCKa) collocated with IJCAI.*

[12] Muhammad Saleem, Ali Hasnain, and Axel-Cyrille Ngonga Ngomo. 2018. LargeRDFBench: a Billion Triples Benchmark for Sparql Endpoint Federation. *Journal of Semantic Web* 48 (2018).

[13] Muhammad Saleem, Alexander Potocki, Tommaso Soru, Olaf Hartig, and Axel-Cyrille Ngonga Ngomo. 2018. CostFed: Cost-Based Query Optimization for SPARQL Endpoint Federation. In *International Conference on Semantic Systems (SEMANTICS)*.

[14] Oshani Seneviratne, Lalana Kagal, and Tim Berners-Lee. 2009. Policy-Aware Content Reuse on the Web. In *International Semantic Web Conference (ISWC)*.

[2] github.com/benjimor/FLiQue