How-Provenance Polynomials for Efficient and Greener Rule Mining

Luis Galárraga

9 septembre 2022

Lab : Inria – Rennes Bretagne Atlantique
Contact person : Luis GALÁRRAGA (luis.galarraga@inria.fr)
Team : LACODAM, **Keywords :** Rule Mining, How-Provenance

1 Introduction

Multiple AI-assisted tasks rely on the ability to learn patterns from knowledge graphs (KGs). These are large machine-readable collections of knowledge, as depicted in Figure 1. KGs are used in many applications that impact our daily lives, such as search engines and smart assistants. Patterns in KGs can take the form of logical rules such as $birthCountry(x, y) \Rightarrow nationality(x, y)$ (people are usually citizens of the country they were born in). Rules are used by machines to make deductions. For example, an AI agent confronted with a query asking for the nationality of a celebrity may deduce the answer from the place of birth, if the actual nationality is unknown to the KG. Rules are automatically discovered or *mined* from the data using resource-intensive algorithms [1].

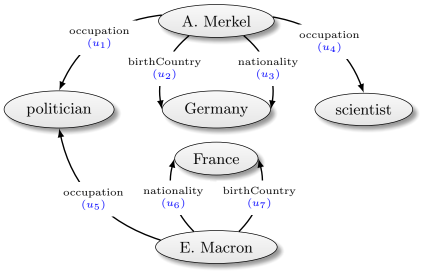

FIGURE 1 – An example KG.

On the other hand, today's KGs are continously updated to be in sync with an ever-changing world where countries change governments, wars break out, and new animal species are discovered. This dynamicity can turn previously learned patterns obsolete. If one or several countries abolish birthright citizenship, then the confidence of our example may decrease. So far the traditional way to handle data updates when mining KGs is to rerun the mining algorithm on the latest version of the KG. This can prohibitively resource-intensive for large and dynamic KGs. We therefore envision a mechanism to mine the rules *incrementally* so that the computational cost of data updates is proportional to the size of the changes – and not to the size of the data. We call this philosophy *mine once, update sometimes.*

1.1 Envisioned Solution and Research Questions

Incremental rule mining is an instance of *update propagation* in databases. This is so because pattern mining relies on executing many queries on the data. For instance, our rule is a query with solutions $S = \{A.\ Merkel/Germany, E.\ Macron/France\}$, support $|S| = 2$ and a confidence of 100%. Now imagine an editor removes edge u_3 from the graph. In that case the pair *Merkel/Ger* is not anymore a solution, which decreases both the rule's support and confidence (by 50%). To avoid rerunning the query from scratch, the literature in databases has proposed to propagate updates via *how-provenance explanations* [2]. For the solution *A. Merkel/Ger*, this is the polynomial $u_2 \otimes u_3$, which attributes this solution to the presence of edges u_2 and u_3. The abstract operator $\otimes$ can be interpreted in different ways. As a logical operator on the *removal* (0) or *not removal* (1) of edges, i.e., $\otimes \mapsto \wedge$, $u_3 \to 0$, $u_2 \to 1$, how-provenance explanations can tell us if a solution remains a solution after an update : In our example this leads to $1 \wedge 0 = 0$, which tells us that A. Merkel is no longer a solution. We can therefore update the query answer without re-executing the query. The polynomials associated to the solutions that support a rule have the same template, e.g., the meta-polynomial $u_{birthCountry} \otimes u_{nationality}$ for our example. We devise two research questions : (a) How much time and energy consumption can be saved with a mine-once-update-sometimes approach to rule mining on KGs ?, (b) what is a good representation for the KG, the rules, and the meta-polynomials ?

1.2 Requirements

We are looking for a M2 student to work on the development of novel algorithms and data representations to mine rules on KGs fully incrementally. This implies (a) to understand the relevant algorithmic challenges and trade-offs ; (b) to become familiar with rule mining on KGs, how-provenance, as well as database and energy benchmarking, and (c) to write efficient code, preferably in a compiled language such as C++ or Rust. Knowledge of Java comes handy but it is not compulsory. Depending on the internship's outcome we envision to search for funding to start a PhD thesis.

Références

[1] Luis Galárraga, Christina Teflioudi, Katja Hose, and Fabian Suchanek. Fast Rule Mining in Ontological Knowledge Bases with AMIE+. *VLDB Journal*, 24(6), 2015.

[2] Daniel Hernández, Luis Galárraga, and Katja Hose. Computing How-Provenance for SPARQL Queries via Query Rewriting. *Proc. VLDB Endow.*, 14(13) :3389–3401, 2021.