KU LEUVEN

FACULTY OF ENGINEERING TECHNOLOGY

DE NAYER (SINT-KATELIJNE-WAVER) CAMPUS

Inverting knowledge graphs back to raw data

How can we leverage the rules we use to construct knowledge graphs to do the inverse?

Tijs VAN KAMPEN

Promotor: Prof dr. ir. Anastasia Dimou

Master thesis submitted to obtain the degree of Master of Science in the engineering technology: Electronics-ICT

academic year 2023 - 2024

Contents

Chapter 1

Introduction

The earliest academic definition of a knowledge graph can be found in a 1974 article as

> A mathematical structure with vertices as knowledge units connected by edges that represent the prerequisite relation (Marchi and Miguel, 1974; Bergman, 2019)

The idea of expressing knowledge in a graph structure predates even this definition, with the concept of semantic networks (Richens, 1956). However, the term knowledge graph only became well-known after Google announced they were using a knowledge graph to enhance their search engine in 2012 (Singhal, 2012). Knowledge graphs are used to make search engines, chatbots, question answering systems, etc more intelligent by injecting knowledge into them (Ji et al., 2022).

These knowledge graphs are constructed by extracting information from various sources, both unstructured sources such as text (using natural language processing) and (semi-)structured sources such as databases, CSV, XML, JSON, RDF (using mapping languages). Many mapping languages exist, some with a specific purpose, such as R2RML (Das et al., 2012) for relational databases, XSPARQL (Bischof et al., 2012) for XML. Others are more general, such as RML (Dimou et al., 2014) and D2RML (Chortaras and Stamou, 2018), having the ability to map from multiple sources in different formats.

To achieve this these mapping languages use a declarative approach, where the user specifies the mapping rules, and the implementation of the mapping language takes care of the actual mapping. Creating these mapping rules is often done by hand. There are tools that make creating these mappings easier, like RMLEditor (Heyvaert et al., 2018b) and YARRRML (Heyvaert et al., 2018a). Alternatively tools exist for automatic generation of mapping rules *cite some tools*.

Retrieving data from a knowledge graph, for consumption by other programs, is done by querying the knowledge graph using SPARQL (Seaborne and Prud'hommeaux, 2008) for tabular data and XSPARQL (Bischof et al., 2012) or XSLT for XML. XSPARQL is the only language that can both map[/lift] and query[/lower], but the syntax for mapping and querying differs, so it could be argued that XSPARQL is actually two languages.

Because of this disconnection between creating and consuming knowledge graphs, much potential is left untapped. We can not use knowledge graphs flexibly as an intermediate representation for

data, as we can not convert the knowledge graph back to the original data format using the same rules we created it with, if the data format has a method for querying[/lowering] at all. As such any changes we make to the data are hard to propagate back to the original data. We can not update, expand or improve the original data using e.g. knowledge graph refining. Nor can we apply changes to a virtual knowledge graph to change the original data.

We aim to improve this situation by extending an existing system implementation with the ability to invert the mapping rules, i.e. mapping the RDF knowledge graph back to raw data (*RQ2*). We choose to extend the Morph-KGC implementation (Arenas-Guerrero et al., 2022) of RML (Dimou et al., 2014) as its end-to-end characteristics make it a good candidate for this task. We also explore how we can leverage RML to construct raw data from heterogeneous data (*RQ1, but this is pretty vague, I could do with a more detailed explanation of what exactly the end goal of this RQ is*).

1.1 Thesis outline

This thesis aims to explore the possibility of inverting knowledge graphs back to their original data format using RML mapping rules. To achieve this we will first look at the current state of the art in chapter 2. We will take a closer look at the technologies used like RDF, SPARQL, and RML. We will also look at the current state of the art for inverting knowledge graphs. In chapter 3 we will look at our implementation of the inversion algorithm. We will look at the algorithm itself, and the implementation details. In chapter 4 we will evaluate our implementation using various benchmarks. For basic testing we use a subset of the rml test cases, which are designed to test the conformance of tools to the RML specification. For more advanced testing we will use various benchmarks simulating real-life use cases like LUBM4OBDA, GTFS-Madrid-Bench and SDM-Genomic-dataset. Finally in chapter 5 we will conclude this thesis, and look at possible future work.

Chapter 2

Related work

Introduction text providing an overview of the related work

2.1 Semantic Web

Tim Berneers-Lee didn't stop at creating the world wide web. He envisioned a version of the web that would also be understandable by machines, and thus the semantic web was born.

2.2 RDF

2.3 SPARQL

2.4 Mapping languages

2.4.1 R2RML

2.4.2 RML

2.5 if meaningful: provenance

Chapter 3

Implementation

Basic text about PoC implementation (for now). Including a high level algorithmic overview of the implementation.

Chapter 4

Evaluation

Placeholder chapter for refering to the evaluation chapter. This should be removed in the intermediate report.

Chapter 5

Conclusion

Placeholder chapter for refering to the conclusion chapter. This should be removed in the intermediate report.

Bibliography

Arenas-Guerrero, J., Chaves-Fraga, D., Toledo, J., Pérez, M. S., and Corcho, O. (2022). Morph-KGC: Scalable knowledge graph materialization with mapping partitions. *Semantic Web*.

Bergman, M. K. (2019). A common sense view of knowledge graphs. *AI3:::Adaptive Information*, 1(1):1–1.

Bischof, S., Decker, S., Krennwallner, T., Lopes, N., and Polleres, A. (2012). Mapping between rdf and xml with xsparql. *Journal on Data Semantics*, 1(3):147–185.

Chortaras, A. and Stamou, G. (2018). D2rml: Integrating heterogeneous data and web services into custom rdf graphs. In *LDOW@WWW*.

Das, S., Cyganiak, R., and Sundara, S. (2012). R2RML: RDB to RDF mapping language. W3C recommendation, W3C. https://www.w3.org/TR/2012/REC-r2rml-20120927/.

Dimou, A., Vander Sande, M., Colpaert, P., Verborgh, R., Mannens, E., and Van de Walle, R. (2014). RML: a generic language for integrated RDF mappings of heterogeneous data. In Bizer, C., Heath, T., Auer, S., and Berners-Lee, T., editors, *Proceedings of the 7th Workshop on Linked Data on the Web*, volume 1184 of *CEUR Workshop Proceedings*.

Heyvaert, P., De Meester, B., Dimou, A., and Verborgh, R. (2018a). Declarative rules for linked data generation at your fingertips! In Gangemi, A., Gentile, A. L., Nuzzolese, A. G., Rudolph, S., Maleshkova, M., Paulheim, H., Pan, J. Z., and Alam, M., editors, *The Semantic Web: ESWC 2018 Satellite Events*, pages 213–217, Cham. Springer International Publishing.

Heyvaert, P., Dimou, A., De Meester, B., Seymoens, T., Herregodts, A.-L., Verborgh, R., Schuurman, D., and Mannens, E. (2018b). Specification and implementation of mapping rule visualization and editing: MapVOWL and the RMLEditor. *Journal of Web Semantics*, 49:31–50.

Ji, S., Pan, S., Cambria, E., Marttinen, P., and Yu, P. S. (2022). A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2):494–514.

Marchi, E. and Miguel, O. (1974). On the structure of the teaching-learning interactive process. *International Journal of Game Theory*, 3(2):83–99.

Richens, R. H. (1956). Preprogramming for mechanical translation. *Mech. Transl. Comput. Linguistics*, 3:20–25.

Seaborne, A. and Prud'hommeaux, E. (2008). SPARQL query language for RDF. W3C recommendation, W3C. https://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

Singhal, A. (2012). Introducing the knowledge graph: things, not strings. 2020-11-13.

Van Assche, D., Delva, T., Haesendonck, G., Heyvaert, P., De Meester, B., and Dimou, A. (2023). Declarative rdf graph generation from heterogeneous (semi-)structured data: A systematic literature review. *Journal of Web Semantics*, 75:100753.