

SPARQL2OWL: towards bridging the semantic gap between RDF and OWL

Mona Alshahrani¹, Hussein Almashouq¹, Robert Hoehndorf¹

¹Computational Bioscience Research Center, King Abdullah University of Science and Technology, Thuwal 23955-6900, KSA

ABSTRACT

Several large databases in biology are now making their information available through the Resource Description Framework (RDF). RDF can be used for large datasets and provides a graph-based semantics. The Web Ontology Language (OWL), another Semantic Web standard, provides a more formal, model-theoretic semantics. While some approaches combine RDF and OWL, for example for querying, knowledge in RDF and OWL is often expressed differently. Here, we propose a method to generate OWL ontologies from SPARQL queries using n -ary relational patterns. Combined with background knowledge from ontologies, the generated OWL ontologies can be used for expressive queries and quality control of RDF data. We implement our method in a prototype tool available at <https://github.com/bio-ontology-research-group/SPARQL2OWL>.

1 INTRODUCTION

Ontologies are widely used to describe and annotate biological data. They have been specifically useful as a way to communicate and integrate databases, and to enable functional analysis, measure semantic similarity and understand biological datasets through network-based statistical analysis.

An increasing number of biological databases is making data available through the Resource Description Framework (RDF) (Lassila and Swick, 1999). RDF is W3C Semantic Web standard that is intended to facilitate data integration and sharing between various resources. RDF is a graph-based language in which triples express edges between two nodes, nodes are intended to represent resources or specific values, and edges express the relation between both. RDF Schema (RDFS) (Brickley and Guha, 2014) is a semantic layer on top of RDF that offers some degree of semantics to RDF graphs. The Web Ontology Language (OWL) (Grau *et al.*, 2008), on the other hand, provides a model-theoretic semantics that can be used in conjunction with RDF data. In most RDF stores, however, RDF data and OWL ontologies are separated and cannot easily be used together. More importantly, knowledge is commonly expressed differently in RDF than it would be in OWL: while OWL requires an explicit distinction between classes and instances and knowledge is expressed through axioms, RDF does not require that such a distinction be made. Consequently, statements such as “Protein X has the function to perform process Y ” could be expressed in RDF through a simple triple, $(X, \text{has-function}, Y)$; in OWL, more nuanced distinctions are possible, including `X SubClassOf: has-function some (realized-by only Y)` (Hoehndorf *et al.*, 2010).

The rapidly growing popularity of RDF stores with public SPARQL endpoints has led to a disparity between Semantic Web data (in RDF) and its semantics (in OWL ontologies). While

some methods have been proposed to mitigate this “semantic gap” through combinations of SPARQL endpoints and OWL reasoning (Hoehndorf *et al.*, 2015; Glimm, 2011; Kollia *et al.*, 2011), the different ways in which knowledge is expressed in RDF and OWL makes it challenging to combine both sources of knowledge.

Here, we propose a method to convert some parts of RDF knowledge into OWL so that it can be used together with background knowledge from ontologies, or alone, for querying and quality control. At the core of our method is the assumption that some of the variables returned by a SPARQL query may stand for OWL entities and could be used within OWL axioms. To formalize this assumption, we extend prior research on conversion of the OBO Flatfile Format to OWL. The OBO format (version 1.2) specifies a graph in which nodes stand for classes and edges for relations between classes (Smith *et al.*, 2007). Originally, the OWL conversion was performed using a fixed translation pattern in which graph edges are converted into an OWL subclass axiom (Horrocks, 2007). Later, the OBO Relation Ontology (Smith *et al.*, 2005) was introduced and implemented through relational patterns in OWL (Hoehndorf *et al.*, 2010). A relational pattern is an OWL axiom with two variables standing for classes and is intended to provide an OWL-based semantics for a type of edge specified in the OBO format. Here, we extend these relational patterns to an arbitrary number of variables and further include variables for relations (i.e., object properties). These extended graph patterns can provide a richer semantics to information contained in RDF graphs, enable more expressive queries and may be used for quality control and consistency checks within RDF datasets. We implement our method in a prototype tool, available at <https://github.com/bio-ontology-research-group/SPARQL2OWL>.

2 METHODS

RDF graphs can be accessed through SPARQL (Prud'Hommeaux *et al.*, 2008), a query language for RDF data. SPARQL queries specify a graph pattern and a set of variables which may stand for nodes or edges in the query pattern. If n variables are specified in a SPARQL query, a (possibly empty) set of n -tuples is returned by a SPARQL query. Depending on the query pattern, some of the variables being returned may stand for instances, classes, object or datatype properties, or annotations.

We define a SPARQL-to-OWL pattern of arity i as a tuple (s_i, p_i) of a SPARQL query s_i with exactly i variables (v_1, \dots, v_i) , and an OWL axiom p_i in which at most i variables are used in the place of an OWL entity. As variables cannot be used for OWL entities in axioms, we define the semantics of the tuple (s_i, p_i) using the set of tuples returned by executing the SPARQL query. Specifically, for each tuple returned by the SPARQL query, we create a new OWL axiom in which each variable is substituted with the value of the

corresponding variable in the tuple returned by the SPARQL query. For a given SPARQL endpoint, multiple SPARQL-to-OWL patterns can be provided which ultimately may be used to generate OWL-based representations of some of the content accessible through SPARQL.

We developed a prototype tool which takes SPARQL endpoint, SPARQL query and a relational pattern in OWL Manchester syntax as input and produces an OWL ontology. The tool consists of a backend and a frontend. In the frontend, a relational pattern is specified by the user as well as a SPARQL query; the pattern defines how the SPARQL query results are transformed into OWL axioms. The conversion can be applied multiple times to generate an ontology from multiple SPARQL queries and different relational patterns. The resulting ontology can also be combined with several background ontologies and then downloaded to process it further using automated reasoning or perform Description Logic (DL) queries.

The backend is implemented in Java using the Manchester OWLAPI (Horridge and Bechhofer, 2011) and the Jena library (Jena, 2007) for performing SPARQL queries. The Manchester syntax parser is used to parse the axioms based on the entered relational pattern definitions and add them to the ontology.

3 RESULTS AND DISCUSSION

We developed a method for converting results of SPARQL queries into OWL using an approach based on relational patterns. Our method takes as input a SPARQL query, a SPARQL endpoint in which the query is performed, and a relational pattern definition. Through the application of multiple queries, with multiple different relational pattern definitions, different pieces of information can be combined together. The result is an OWL ontology which contains the transformed knowledge and which can be combined with other ontologies. Figure 1 provides an overview over our method and illustrates the workflow of SPARQL2OWL conversion tool.

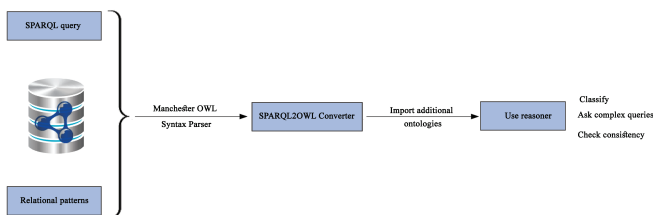


Fig. 1. workflow of SPARQL2OWL conversion tool

An example of a binary query pattern, used again the DisGeNet SPARQL endpoint (Piero *et al.*, 2015), is the following query to retrieve the phenotypes associated with disease in OrphaNet:

```

SELECT ?orphanet ?phenotype WHERE {
  ?orphanet sio:SIO_001279 ?phenotype .
  FILTER regex(?orphanet, "identifiers.org/orphanet")
  ?orphanet dct:terms:title ?orphanetName .
}
  
```

A suitable relational pattern definition to represent the query results could be

```
?orphanet SubClassOf: has-phenotype some ?phenotype
```

or another form of representing this information in OWL. The resulting ontology can then be combined with the Human Phenotype Ontology or Mammalian Ontology so that the axioms in these ontologies can be used for querying as well as quality control.

REFERENCES

- Brickley, D. and Guha, R. V. (2014). Rdf schema 1.1. *W3C Recommendation*, **25**, 2004–2014.
- Glimm, B. (2011). Using sparql with rdfs and owl entailment. In *Reasoning Web. Semantic Technologies for the Web of Data*, pages 137–201. Springer.
- Grau, B. C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., and Sattler, U. (2008). Owl 2: The next step for owl. *Web Semantics: science, services and agents on the World Wide Web*, **6**(4), 309–322.
- Hoehndorf, R. *et al.* (2010). Relations as patterns: bridging the gap between obo and owl. *BMC bioinformatics*, **11**(1), 1.
- Hoehndorf, R., Slater, L., Schofield, P. N., and Gkoutos, G. V. (2015). Aber-owl: a framework for ontology-based data access in biology. *BMC bioinformatics*, **16**(1), 1.
- Horridge, M. and Bechhofer, S. (2011). The owl api: A java api for owl ontologies. *Semantic Web*, **2**(1), 11–21.
- Horrocks, I. (2007). Obo flat file format syntax and semantics and mapping to owl web ontology language. *Technical Report, University of Manchester*.
- Jena, A. (2007). semantic web framework for java.
- Kollia, I., Glimm, B., and Horrocks, I. (2011). Sparql query answering over owl ontologies. In *The Semantic Web: Research and Applications*, pages 382–396. Springer.
- Lassila, O. and Swick, R. R. (1999). Resource description framework (rdf) model and syntax specification.
- Piero, J., Queralt-Rosinach, N., Bravo, ., Deu-Pons, J., Bauer-Mehren, A., Baron, M., Sanz, F., and Furlong, L. I. (2015). Disgenet: a discovery platform for the dynamical exploration of human diseases and their genes. *Database*, **2015**.
- Prud'Hommeaux, E., Seaborne, A., *et al.* (2008). Sparql query language for rdf. *W3C recommendation*, **15**.
- Smith, B. *et al.* (2005). Relations in biomedical ontologies. *Genome biology*, **6**(5), R46.
- Smith, B. *et al.* (2007). The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, **25**(11), 1251–1255.