Are lobsters spiders? - Finding taxonomy relations using linked data

Sébastien Klasa

8 May 2019

1 Introduction

"Are lobsters spiders?" is an easy (if not obvious) question for a human, but difficult for a computer to answer unless using a specialized data model. Here we present a solution that uses linked data to answer the generalization of this question: given two taxa [1] A and B, we want to know if A is a B. We also present an implementation of this solution that retrieves results from a taxonomy dataset with SPARQL queries. We take our solution even further by allowing non-specialists to use our implementation by using general names like "spider" instead of taxa like "araneae".

2 Taxonomy relations

To answer the question "Are lobsters spiders?", we first have to define what does it mean for an animal A to be another animal B. In this case, we can say that if animals can be classified in groups, which can also be classified in higher rank groups, we end up with a tree where leafs are species and nodes are groups of species of some rank. This type of classification is named a taxonomy. We can use this type of classification to define: "A is a B" is true when A is a direct or indirect child of B in the tree. To answer our first question we need to get a taxonomy tree, find the nodes "lobster" and "spider" and check if the node "lobster" is a child of the node "spider".

The taxonomy we use is the NCBI organismal classification ontology [2], which contains nearly two millions taxa. We search for the lobster and spider URIS and then check if the lobster URI is a subclass of the spider URI using a SPARQL query (see figure 1). This query can be generalized to any pair of taxa by replacing the NCBI URIs by another ones.

[1] In biology, a taxon is a group of one or more populations of an organism or organisms seen by taxonomists to form a unit. For example, African elephants form the genus *Loxodonta* (Wikipedia).

[2] `http://www.obofoundry.org/ontology/ncbitaxon.html`

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX ncbi: <http://purl.obolibrary.org/obo/>
ASK
WHERE {
        ncbi:NCBITaxon_6693 rdfs:subClassOf* ncbi:NCBITaxon_6893 .
}
```

Figure 1: SPARQL query that asks if a lobster is a spider.

While this solution tells us if a A is a B, we wanted to make a step further by searching for the relation between the two animals when A is not a B. One interesting information that we can get from the taxonomy tree is the group of animals in which both A and B are. In the tree, this group is the least common parent of the two nodes, or in the case of the ontology, the least common superclass (LCS) of the two nodes. The figure 2 shows a SPARQL query that looks for the LSC of the lobster and the spider. Again, ge can extend this query to other animals by replacing the URIs.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX ncbi: <http://purl.obolibrary.org/obo/>
SELECT ?lcs ?label
WHERE {
        ?lcs ^rdfs:subClassOf*
                ncbi:NCBITaxon_6693 ,
                ncbi:NCBITaxon_6893 .
        FILTER NOT EXISTS {
                ?sublcs ^rdfs:subClassOf*
                        ncbi:NCBITaxon_6693 ,
                        ncbi:NCBITaxon_6893 .
                ?sublcs rdfs:subClassOf ?lcs;
        }
}
```

Figure 2: SPARQL query that searches for the least common superclass of the lobster and the spider.

Although this solution is sufficient for a specialist that knows the needed taxa, it is hard for non-specialist users to find the URIs to build this query: 1) they have to find the exact taxon of the animal they are looking for (e.g. 'araneae' for 'spider'), 2) the names of nodes in the taxonomy ontology are numbers, so the users have to search the dataset to find it. To make our implementation easier to use, we decided to take advantage of another dataset that is more self-descriptive.

3 Finding taxa using the DBpedia dataset

The DBpedia dataset is a linked data representation of Wikipedia articles, which makes it easy to search for subjects, and in our case, for animals. We wanted to take advange of both the DBpedia and the NCBI taxonomy datasets, in order to enable a user to search for two animals in DBpedia and then checks if they are related using the NCBI dataset. To achive that, we needed links between DBpedia and NCBI URIs, which does not exist in these datasets. However, a Wikipedia page for an animal ofter contains its taxon, which is represented in DBpedia by literals linked to the page by some properties. These property are not the same for every page so we needed to find the most relevant ones first.

We implemented a script that iterates through all nodes in the NCBI dataset and, for each node, gets its taxon. The script then searches literals of the taxon in the DBpedia dataset and counts the number of occurences of properties that links the literals to a Wikipedia page. We manually selected property that are commonly used and that are relevant for our purpose. The figure 3 lists the properties that we finnaly use to retrieve taxa from Wikipedia pages.

```
http://www.w3.org/2000/01/rdf-schema#label
http://www.w3.org/2000/01/rdf-schema#label
http://dbpedia.org/property/binomial
http://dbpedia.org/property/genus
http://xmlns.com/foaf/0.1/name
http://dbpedia.org/property/subdivision
http://xmlns.com/foaf/0.1/givenName
http://dbpedia.org/property/title
http://dbpedia.org/property/taxon
http://dbpedia.org/property/familia
http://dbpedia.org/property/species
http://dbpedia.org/property/name
```

Figure 3: Properties that we use to search for the taxon of a Wikipedia page.

4 Conclusion

We implemented our final solution in C#.NET Core using the dotNetRdf package. Our tool enables a user to enter two DBpedia URIs. The tool searches for the taxon of each URI and, based on the taxon, find the associated NCBI URIs. With those two URIs, which we will call A and B, we can finaly check if A is a B (A is a direct or indirect child of B in the toxonomy tree), or in the case of a negative answer, search for the common group of animals containing A and B (the least common superclass of A and B).