Towards a Question Answering System Over a Vocabulary Catalogue

Ghislain Auguste Atemezing
MONDECA
35 boulevard Strasbourg, Paris, France
ghislain.atemezing@mondeca.com

ABSTRACT

There is no doubt that there is an increasing presence of structured data on the web. At the same time, web users have different skills and want to be able to interact with Linked datasets in various manner, such as asking questions in natural language. Over the last years, the QALD challenges series are becoming the references for benchmarking question answering systems. However, QALD questions are targeted on datasets, not on vocabulary catalogues. This paper proposed a first implementation of 14 questions applied to the Linked Open Vocabularies (LOV) catalogue, mainly focused on metadata information retrieval. The goal is to provide to end users yet another access to metadata information available in LOV using natural language questions. Currently, the system handles almost 92,99% of the vocabularies in LOV.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; H.3.5 [**Online Information Services**]: Data sharing—*Web-based services*

Keywords

Question Answering, Vocabulary Catalogue, metadata

1. INTRODUCTION

The recent years have seen the adoption of semantic web in many domains, generating a mass of structured data available in RDF. Linked data has contributed to interlink different datasets across domains, where publishers use best practices for producing interoperable datasets by reusing ontologies and making alignments.

However, users need to have a minimal knowledge of SPARQL language and RDF skills to query RDF datasets. It is one barrier that is overcome by Question Answering (QA) systems, which directly take as input questions in natural language. The need for more advanced tools and QA systems that operate over large repositories of Linked Data has also been the motivation for the QALD Question Answering over Linked Data (QALD) series of workshops [1]. Started in 2011, QALD challenges offer a test sets questions whose answers can be found in the DBpedia and MusicBrainz, with more complex questions added each year. However, there is not yet a specific set of questions specific to retrieve metadata at schema level.

Vocabulary catalogues and semantic search engines are special datasets as classes and properties are used to model and generate datasets available in the Linked Data space. Ontologies are important part in the semantic web layer to build full interoperable datasets. Most vocabulary catalogues provide terms search and APIs to access their datasets. Linked Open Vocabulary provides five types of search criteria: metadata search, ontology search, APIs access, dump file in RDF and SPARQL endpoint access [4]. Consider the following question in natural language *"What is dcterms?"*. The answer in English corresponds to the following SPARQL query in the LOV endpoint:[1]

```
PREFIX vann:<http://purl.org/vocab/vann/>
PREFIX voaf:<http://purl.org/vocommons/voaf#>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT DISTINCT ?x1  {
  GRAPH <http://lov.okfn.org/dataset/lov>{
    ?x0 a voaf:Vocabulary.
  ?x0 vann:preferredNamespacePrefix "dcterms".
  ?x0 dcterms:description ?x1.

}}
```

This paper presents a prototype for a vocabulary backed question answering system that can transform natural language questions into SPARQL queries, thus giving the end users access to the information stored in vocabulary repositories. The paper is structured as follows: Section 2 describes the set of questions, followed by a system description in Section 3. An evaluation is presented in Section 4 and a short conclusion and future work in Section 5.

2. TYPES OF QUESTIONS

LOV is a highly curated observatory of the semantic vocabularies ecosystem, with the aim of promoting the reuse

[1] http://lov.okfn.org/dataset/lov/sparql

of well-documented vocabularies in the Linked Data space. Each version of a vocabulary in LOV contains metadata information which are the following:

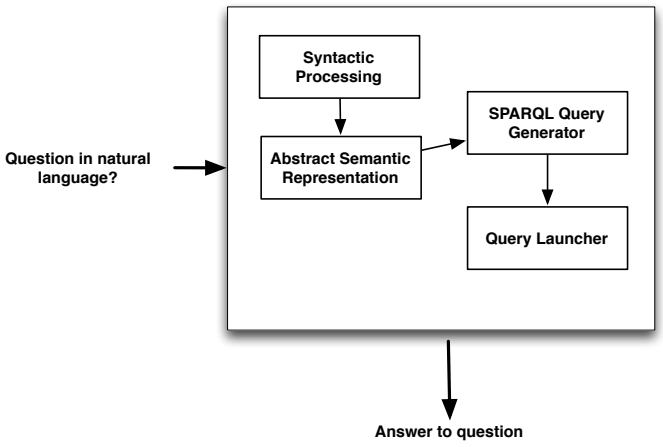
- A preferred namespace, useful when the vocabulary is not dereference-able;
- A URI, unique identifier of the vocabulary;
- A title, a short text;
- A homepage, mainly a page where to find the documentation of the vocabulary;
- A description, which gives more information about the scope of the vocabulary;
- The creators of the vocabulary, represented by their URIs.
- The publishers of the vocabulary, and
- The different languages in which the vocabulary is available.

Additionally, users could be interested in other interesting facts, such as the number of versions, the number of datasets using the vocabulary, the number of external vocabularies reusing the vocabulary and the category to which belong the vocabulary. A first set of 14 questions in natural language can be handle by the prototype, covering different type of metadata information available in a vocabulary. Table 1 shows the list of questions, where in the column "Template", [be] can be either the present or the past form of the verb, and [vocab] is the preferred namespace of the vocabulary.

3. SYSTEM DESCRIPTION

The system receives as input a natural language question formulated in English and outputs the query that will retrieve the answer to the question from the LOV catalogue. The architecture of the system is illustrated in Fig. 2.

Figure 2: Architecture of the system.



The implementation uses the Quepy tool from Machinalis [3]. The POS tagset used by Quepy is the Penn Tagset [2]. First, regular expressions are defined to match the natural language questions and transform them into an abstract semantic representation. Then, specific templates is defined to handle the questions that the system can handle. To handle regular expressions, Quepy uses the `refo` library[2] which work with regular expressions as objects.

Table 2: Relationship between Questions and Properties used to generate the SPARQL query from natural language

Question ID	Property
Q1	`dcterms:description`
Q2	`dcterms:publisher`
Q3	`dcterms:issued`
Q4	`dcterms:issued`
Q5	`dcterms:contributor`
Q6	`dcterms:modified`
Q7	`dcat:distribution`
Q8	`dcterms:language`
Q9	`foaf:homepage`
Q10	`voaf:reusedByVocabularies`
Q11	`voaf:reusedByDatasets`
Q12	`vann:preferredNamespaceUri`
Q13	`dcterms:title`
Q14	`dcat:keyword`

A vocabulary is defined by a fixed relation `voaf:Vocabulary`[3] and a POS associated to a `vann:preferredNamespacePrefix`. LOV uses a unique prefix to identify namespaces, which is a string from length 2 to length 17, although the recommendation for publishers is to use a prefix with less than 10 characters [5].

The syntactic processor is based on regular expressions using POS terms. As a vocabulary is identified by its prefix, we use the following syntactic patterns: NN, NNS, FW, DT, JJ and VBN. Each question Q1 to Q14 is associated to a unique template. After, when a prefix is recognized, the semantic interpreter uses fixed relations with the English tag, which are properties in RDF triple pattern. Table 2 presents the different fixed relations currently used in the system to cover the set of the 14 questions.

4. SYSTEM EVALUATION

This first set of questions are not found in the QALD challenges, where answers had to be found either in DBpedia or in a federated dataset such as Yago2 and MusicBrainz. In terms of complexity, the test questions are either simple questions related to retrieve metadata information.

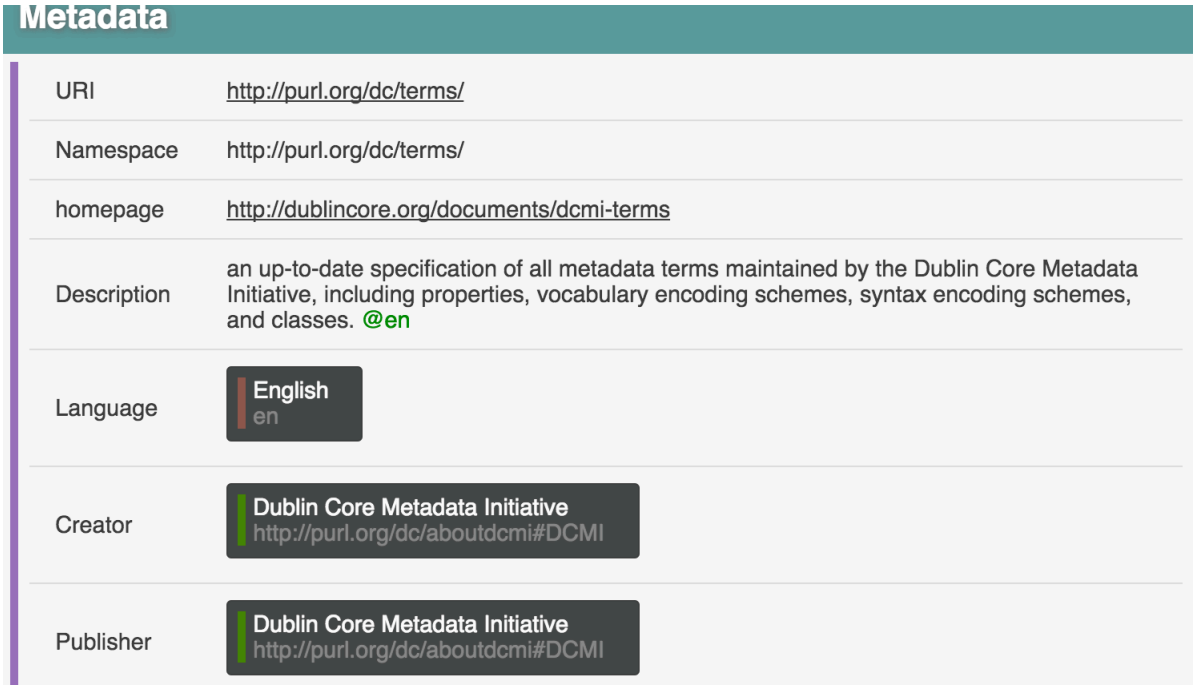
All the questions which generated SPARQL query gives satisfactory results. The most challenging issue is to determine the most suitable POS that cover all the vocabulary prefixes. For example, out of 528[4] vocabularies in LOV, 13 of them contain an hyphen and the system can not generate query: `elseweb-modelling`, `sdmx-dimension`, `omn-federation`,

[2] `https://github.com/machinalis/refo`
[3] All the prefixes are the ones used at `http://lov.okfn.org/dataset/lov/vocabs`
[4] This number corresponds to the total number of vocabularies inserted in LOV as of January, 8th 2016.

Table 1: Questions in natural language for retrieving metadata information in a vocabulary catalog.

ID	Template	Sample Question
Q1	*What [be] [vocab]?*	*What is prov?*
Q2	*Where [be] [vocab] from?*	*Where is foaf from?*
Q3	*How old [be] [vocab]?*	*How old is prov?*
Q4	*When [be] [vocab] release?*	*When was voaf release?*
Q5	*Who [be] the contributors of [vocab]?*	*Who are the contributors of adms?*
Q6	*When [be] [vocab] last update?*	*When was schema last update?*
Q7	*What [be] the versions of [vocab]?*	*What are the versions of adms?*
Q8	*What [be] the languages of [vocab]?*	*What are the languages of dcat?*
Q9	*Where to find [vocab] documentation?*	*Where to find foaf documentation?*
Q10	*How many vocabularies reuse [vocab]?*	*How many vocabularies reuse adms?*
Q11	*How many datasets use [vocab]?*	*How many datasets use adms?*
Q12	*What [be] the namespace of [vocab]?*	*What is the namespace of dcterms?*
Q13	*What [be] the title of [vocab]?*	*What is the title of foaf?*
Q14	*What [be] the category of [vocab]?*	*What is the category of dcterms?*

Figure 1: A sample view of metadata information of Dublin Core vocabulary in LOV.



`omn-lifecycle`, `elseweb-data`, `dbpedia-owl`, `geo-deling`, `sdmx-code`, `wf-invoc`, `iso-thes`, `eac-cpf`, `p-plan` and `ma-ont`. Moreover, 21 prefixes containing a number (e.g., `g50k`) and 3 special cases (`homeActivity`, `LiMo`, and `juso.kr`) are not currently covered by the system. Currently, the system handles almost 92,99% of the prefixes in LOV.

5. CONCLUSION AND FUTURE WORK

In this paper, we have presented a prototype system for answering a set of 14 questions in natural language backed by a vocabulary catalogue. The questions are targeted to retrieve metadata information in vocabularies. The approach is based on the identification of the vocabulary prefix combining POS and regular expressions. The implementation uses the LOV dataset in RDF and the Quepy tool. The first results show that the system handles 92,99% of the vocabularies in the LOV catalogue. We plan to extend the types of queries to more complex ones. Also, we intend to align the set of questions to the QALD challenges to make the system comparable to existing Question Answering systems.

Acknowledgments. Thanks to the LOV team and curators for maintaining the LOV catalogue, and Machinalis for the open source Quepy tool.

6. REFERENCES

[1] V. Lopez, C. Unger, P. Cimiano, and E. Motta. Evaluating question answering over linked data. *Web Semantics Science Services And Agents On The World Wide Web*, 21:3–13, 2013.

[2] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini.

Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330, 1993.

[3] Michanalis. Quepy Tool, 2012. `https://github.com/machinalis/quepy/`.

[4] P.-Y. Vandenbussche, G. A. Atemezing, M. Poveda-Villalónc, and B. Vatant. LOV: a gateway to reusable semantic vocabularies on the Web. *Semantic Web Journal*, 2015. `http://www.semantic-web-journal.net/system/files/swj974.pdf`.

[5] P.-Y. Vandenbussche and B. Vatant. Metadata Recommendations For Linked Open Vocabularies. OKFN, 2012. `http://lov.okfn.org/dataset/lov/Recommendations_Vocabulary_Design.pdf`.