Natural Language Interaction with Semantic Web Knowledge Bases and LOD

Mariana Damova*, Dana Dannélls†,
Ramona Enache†, Maria Mateva*, Aarne Ranta†

*Ontotext, AD, Sofia, Bulgaria
†Chalmers University and GU, Sweden
`{mariana.damova,maria.mateva}@ontotext.com`
`{dana.dannells,ramona.enache,aarne.ranta}@chalmers.se`

Abstract. Cultural heritage appears to be a very useful use case for Semantic Web technologies. The domain provides with plenty of circumstances where linkages between different knowledge sources are required to ensure access to rich information and respond to the needs of professionals dealing with cultural heritage content. Semantic Web technologies offer the technological backbone to meet the requirement of integrating heterogeneous data easily, but they are still more adapted to be consumed by computers than by humans, especially non-engineers or developers. This chapter is about a technique which allows interaction in natural language with semantic knowledge bases. The proposed technique offers a method that allows querying a semantic repository in natural language and obtaining results from it as a coherent text. This unique solution includes several steps of transition from natural language to SPARQL and from RDF to coherent multilingual descriptions, using the Grammatical Framework, GF. The approach builds on a semantic knowledge infrastructure in RDF, it is based on OWLIM-SE and the data integration method Reason-able View supplied with an ontological reference layer. The latter is connected via formal rules with abstract representations derived from the syntactic trees of natural language input using the GF resource grammar library.

Key words: CIDOC-CRM, Cultural Heritage, Functional programming, Grammatical Framework, Information Retrieval, Natural Language Generation, Natural Language Understanding, Ontology, Ontology Mapping, OWLIM, Question Answering, Reason-able View, Semantic Web, SPARQL.

1 Introduction

Cultural heritage appears to be a very useful use case for Semantic Web technologies. The domain provides with plenty of circumstances where linkages between different knowledge sources are required to ensure access to rich information and respond to the needs of professionals dealing with cultural heritage content. Semantic Web technologies offer the technological backbone to meet the requirement of integrating heterogeneous data easily, but they are still more adapted to be consumed by computers than by humans, especially non-engineers or developers. The main obstacle for this is the fact that in order to retrieve information, it is necessary to master SPARQL, a query language for RDF (Resource Description Framework) [1] and the schemata of each integrated dataset in the knowledge base of interest.

This chapter is about a technique which allows interaction in natural language (NL) with semantic knowledge bases. The proposed technique adapts various approaches from the fields of Question Answering (QA), Information Retrieval (IR) and Natural Language Generation (NLG). It offers a method which allows querying a semantic repository in natural language and obtaining results from it as a coherent text. This unique solution includes several steps of transition from natural language to SPARQL and from RDF [2] to coherent multilingual descriptions, using the Grammatical Framework, GF [3]. The approach builds on a semantic knowledge infrastructure in RDF, it is based on OWLIM-SE [4] and the data integration method Reason-able View [5] supplied with an ontological reference layer [6]. The latter is connected via formal rules with abstract representations derived from the syntactic trees of natural language input using the GF resource grammar library [7,3].

The highlights of the approach and its realization are presented in the following order. Section 2 describes the technological infrastructure which provides the data pool for querying, retrieval and text generation. Section 3 outlines the method of producing SPARQL queries from a natural language input. Section 4 presents the multilingual generation results from the Museum Reason-able View and describes how well the system performs. Section 5 comments on related work. Section 6 concludes with remarks about the approach and its novelty.

2 The Knowledge Representation Infrastructure

To allow experimentation with the natural language interface described in this chapter, it is necessary to have access to a semantic knowledge pool which allows to query and retrieve answers from it. This section presents the principles and the construction of such a knowledge pool. We call this pool the knowledge representation infrastructure. We use it to showcase the natural language to ontology interoperability.

The knowledge representation infrastructure adopted in our approach is designed as a Reason-able View of the Web of Data. This method lies on the assumptions about the Semantic Web [8] addressing issues of access to structured data, availability of structured models, and reasoning over data instances. These are well known problems arising from the design principles of the Semantic Web and the Linked Data.[1] The former being an extension of the standard Web, allows to encode and express the relationships between web pages,

---

[1] `http://linkeddata.org`

letting machines to understand the meaning of some content of the pages. Linked data, on the other hand, abides principles of publishing open data in RDF, where each data element (alias resource) is represented as a web link and is supplied with useful information coming from linking to other web resources. The architecture of the Semantic Web and the Linked Data that enables access to data placed on different servers do not allow reasoning across them so far and cannot guarantee 100% availability of the resources. The Reason-able View approach addresses and aims to solve these problems. It enables reasoning across datasets and circumvents the availability of the data for HTTP (Hypertext Transfer Protocol Secure) requests.

A Reason-able View is a compound dataset composed of several RDF datasets and their schemata. It allows reasoning to be performed on all the statements of all datasets all together, based on a given reasoning language. To query such a compound dataset, the user has to be intimately familiar with the schemata of each single composing dataset. That is why the Reason-able View approach is extended with the so called ontological reference layer, which introduces a unification ontology, mapped to the schemata of all single datasets from a given Reason-able View and thus provides a mechanism for efficient access and navigation of the data by allowing to formulate queries in terms of the unification ontology and retrieve data from all datasets of the Reason-able View. The knowledge representation infrastructure described in this chapter is the Reason-able View of the Web of Data with a reference layer. Its content is described in the following section.

2.1 Museum Reason-able View

The cultural heritage semantic knowledge representation infrastructure is a reason-able view of the web of data. We call it the Museum Reason-able View. The Museum Reason-able View is an assembly of RDF datasets from museum collections and from LOD,[2] which is loaded into OWLIM-SE with inference preformed on the data with respect to OWL Horst [9], thus extending the introduced explicit statements with implicit ones and increasing the available knowledge for querying with about 20%. As described in [10], the Museum Reason-able View gathers: (a) datasets from LOD, including DBpedia,[3] the RDF-ized version of Wikipedia, describing more than 3.5 million things and covers 97 languages; (b) a unification ontology which provides links to the LOD, i.e. PROTON,[4] an upper-level ontology, consisting of 542 classes and 183 properties; (c) cultural heritage specific ontologies, such as: (i) CIDOC-CRM,[5] an object oriented ontology developed by the International Council of Museum's Committee for Documentation (ICOM-CIDOC), with overall scope of curated

---

[2] `http://linkeddata.org`

[3] DBPedia, structured information from Wikipedia: `http://dbpedia.org`.

[4] PROTON, a lightweight upper-level ontology: `http://www.ontotext.com/proton-ontology`

[5] `http://www.cidoc-crm.org/`

knowledge of museums. It consists of about 90 classes and 148 properties. The CRM classes are integrated with PROTON classes using OWL constructs [11]; (ii) Museum Artifacts Ontology (MAO),[6] developed for mapping between museum data and the K-samsök schema.[7] The ontology includes classes reflecting the K-samsök schema to allow integrating the data from the Swedish museums. It has about 10 classes and about 20 new properties. (d) Painting ontology, developed to cover detailed information about painting objects in the framework of the Semantic Web.[8] It contains 197 classes and 107 properties of which 24 classes are equivalent to classes from the CIDOC-CRM and 17 properties are sub-properties of the CIDOC-CRM properties. It has been used as a reference unification ontology to support natural language to ontology and SPARQL interoperability and to allow unified access to the three cultural heritage datasets (see Section 2.2).

The Museum Reason-able View is accessible via SPARQL end point.[9] The top browser layer is provided by the Forest framework, developed at Ontotext. Through the Forest framework it is possible to retrieve knowledge by formulating queries like *Oil paintings from the GIM collection*, *Paintings with a Gothenburg motive*, *Portraits and their painters*, *Museum Objects from Swedish Museums*, *Museum objects of height more than 30 centimeter*, *Paintings given as a present to the Gothenburg City Museum* in SPARQL. This SPARQL end point is used to link the semantic knowledge representation infrastructure to the natural language interface, allowing to query the RDF knowledge representation infrastructure with natural language queries, and to generate coherent natural language texts from the query results. In Section 3 we describe the method in more detail.

2.2 Cultural Heritage Datasets

The cultural heritage datasets that we made available through the Museum Reason-able View are (a) The Gothenburg City Museum data preserves 8900 museum objects described in its database. These objects were extracted from two of the museum collections, namely GSM (Göteborg Stadmuseum) and GIM (Göteborg Industri Museum) each of which is describe in the museum database tables. 39 metadata fields display each museum object, including its identification, its type, e.g. painting, sculpture, its material, its measurements, its location, etc. All fields, both data and metadata, that are mainly given in Swedish were translated into English; (b) Painting objects from DBpedia, covering 15,350 entries, each is described with 8 metadata fields. These entries were retrieve by formulating a specific SPARQL query. This query, illustrated below, selects paintings based on the content of the RDF resources;

---

[6] It is just a coincidence that this ontology has the same name as the Finish MAO [12], which also describes museum artifacts for the Finish museums.

[7] K-samsök `http://www.ksamsok.se/in-english/`), the Swedish Open Cultural Heritage (SOCH), is a Web service for applications to retrieve data from cultural heritage institutions or associations with cultural heritage information.

[8] `http://spraakdata.gu.se/svedd/painting-ontology/painting.owl`

[9] http://museum.ontotext.com/sparql

```
select ?author ?painting where {
  ?author1 fb:visual_art.visual_artist.artworks ?painting .
  ?author1 ff:preferredLabel ?author .
  ?painting:comment ?comment .
  FILTER (regex(?comment, "painting"))
}
```

(c) Paintings from Europeana Semantic Data are pooled from the SPARQL end point,[10] also by formulating a query that matches the content of the description of the RDF resources, example of this query is shown below.

```
select ?autor ?title
{
  ?painting edm:type "IMAGE" ;
            ore:proxyIn ?proxy ;
            dc:title ?title ;
            dc:creator ?author ;
            dc:subject ?type ;
            dcterms:created ?year ;
            dc:description ?desc ;
            dc:source ?source .
  ?painting ore:proxyIn ?proxy .
  ?proxy edm:dataProvider ?museum .
  FILTER regex(?type, "painting", "i")
    }
```

To allow a unified access to each of the cultural heritage datasets, their schemata has been mapped to the Painting ontology .

2.3 OWLIM: Semantic Data Storage

The datasets described in the previous section (Section 2.2) were loaded into OWLIM-SE semantic repository and full materialization with respect to OWL Horst [13] is performed during loading. OWL [11] is an ontology language which supports more complex logical descriptions than RDFs [14] class equivalence construct *owl:equivalentClass*. It provides the reasoning mechanisms ensuring class construction through property restrictions, property types definitions, like transitive, inverse, symmetric, implying the generation of certain implicit statements. OWL is based on Description Logic (DL) [13], and has three versions: OWL Lite, OWL DL, and OWL Full. The first one being the less expressive and the last one being so complex, that is considered computationally undecidable. To bridge the gap of expressiveness, and logical decidability, while allowing scalable reasoning, other dialects have been created. They are positioned between RDFs and OWL Lite. OWL Horst is an extension of RDFs.

---

[10] http://europeana.ontotext.com

It is based on ter Horst [9], where he defines RDFs extensions toward rule support as a dialect of OWL, which makes use of rule entailment (R-entailment) of RDF graphs. OWL Horst language has the following characteristics:

- It is a proper (backward compatible) extension of RDFs, which allows to use the classes of RDFs with OWL Horst reasoning
- It allows rule extensions without DL-related constraints because it is based on R-entailment formalism
- Its complexity allows greater scalability compared to other approaches combining DL ontologies with rules.

OWLIM supports OWL Horst. It is used for full materialization during loading. As a result of this, the Museum Reason-able View contains 1,779,944 explicit triples, and 413,157 implicit triples, providing 16% more retrievable triples, e.g. 230,982. These statistics reflect the number of triples formed by a selection of the paintings from DBpedia, and not the entire dataset. This selection was provided to make the dataset smaller to reduce the processing time while experimenting with the data.

3 Interoperability between GF and Ontologies

Most people who are using Web-based search engines usually formulate their queries with the help of keywords. However, ontology based RDF data allow for more complex semantic-based queries that are accessed through the ontology classes and semantic relations between them, i.e. class and property assertions. To take as an example, a SRARQL query formulated for retrieving *all oil painting objects that belong to the Gothenburg City Museum* contains the semantic classes: *Oil painting*, *Museum* and the properties, i.e. *belong to*. Thus, the ontology restricts the number of semantic queries that can be run against it, as it represents a closed world bound by the concepts (ontology classes) and relations (ontology properties) that are included in it. Therefore, the number of possible semantic queries is finite. Our approach relies on this assumption.

In addition, an ontology has a logically organized structure that semantically characterises the domain. This allows formulating a controlled language that will exhaustively cover all possible conceptual semantic queries. They can be easily translated into SPARQL queries consequently. For example, the *Painting ontology* which is part of the knowledge representation infrastructure will allow to formulate queries about the different types of paintings such as *portraits, oil paintings, water colour paintings*, etc.

Furthermore, one semantic query can be expressed in multiple ways in natural language. For example, the query *oil paintings from the GIM collection* has the same semantic interpretation as the queries *what oil paintings belong to the GIM collection*, *show all oil paintings from the GIM collection*. All these natural language sentences, both imperative and wh-questions, can be formulated with a single semantic SPARQL query as shown below:

```
select ?painting ?collection where {

?painting rdf:type painting:Painting ;
painting:belongsTo ?collection ;
painting:hasMaterial painting:OilPaint .
?collection rdfs:label "GIM" .
}
```

Regardless the syntactic form of the sentence, it will map to same SPARQL query and the results retrieved from such a query will be identical for each sentence.

3.1 The Grammatical Framework, GF

The grammar formalism we employ is the Grammatical Framework (GF) [15]. It is a grammar formalism, based on Martin Löf's type theory [16]. The key feature of the grammar is the division between the abstract syntax, i.e the semantic representation of the domain and concrete syntaxes, representing linearizations in various target languages either natural or formal. This division has been proved advantageous in the context of multilingual natural language generation from ontologies [17,18,19], and also in performing similar tasks such as multilingual controlled natural languages [20,21].

GF comes with a resource library [3], covering the syntax of 30 languages. The resource library aids the development of new grammars for specific domains by providing the operations for basic grammatical constructions, and thus making it possible to produce correct natural language analysis for all the languages that are covered in its library. We take advantage of this power to cover 15 languages.

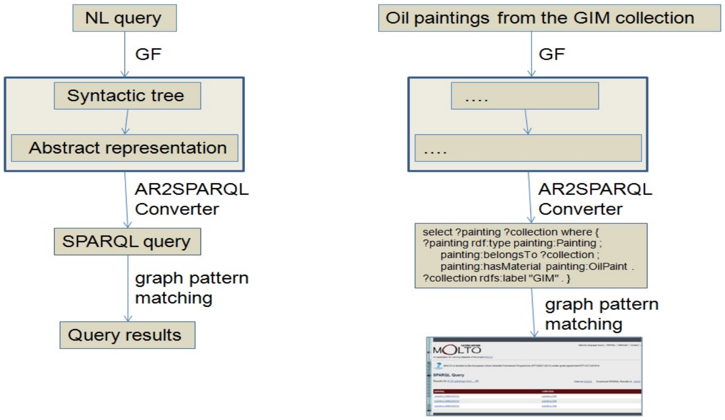
3.2 GF – SPARQL – GF

In the context of the Semantic Web, semantic data is accessible via the SPARQL endpoint as in our Museum Reason-able View of Linked Open Data (LOD), shown in Figure 4. One of the bottlenecks of SPARQL is that formulating a query requires knowledge of the query language and of the schemata underlying the datasets in the knowledge representation infrastructure. To avoid this, natural language/controlled natural language mechanism could be used to help the user formulate queries by suggesting the valid words. These words are in fact the lexicalizations of the classes and properties that are available in the knowledge representation infrastructure.

We implemented a system that generates SPARQL queries from natural language, and from a set of RDFs to a coherent natural description. The system consists of five modules: (1) Query; (2) Answer; (3) Text; (4) Lexicon; and (5) Data. The Data and the Lexicon modules are shared by the remaining three modules. The *Answer* module is a top module, its task is to generate either a yes/no answer or a coherent text as a response to a query, using both the *Query* and the *Text* modules .

The approach to queries and texts is that the abstract syntax is driven by the ontology and the concrete syntax by the resource grammars. Part of the abstract syntax is generic (such as wh-questions and quantifiers), the other part, the predicates are domain-dependent. In the same way, part of the concrete syntax is language dependent and language independent.

Query Module: NL to SPARQL to NL Generating descriptions of ontology objects as a response to a SPARQL query starts off by formulating a question in natural language instead of in SPARQL syntax.



```
select ?painting ?collection where {
?painting rdf:type painting:Painting ;
     painting:belongsTo ?collection ;
     painting:hasMaterial  painting:OilPaint .
?collection rdfs:label "GIM" . }
```

Fig. 1. NL to SPARQL query results processing flow

Figure 1 shows the processing flow of querying the knowledge representation infrastructure in natural language. The natural language query is analyzed by GF, then an AR2SPARQL converter translates from the GF analysis abstract representation into a SPARQL query, which in turn is run against the RDF knowledge base and retrieves the query results. Some examples of the queries that are supported by the grammar are: *who painted x, what is the material of x, All x painted by y, Show everything about all x that are painted on y*. Where x and y are either ontology classes or instances that have been defined and linearized in the lexicon.

To allow translation to SPARQL, some strings were linearized (lin) with appropriated SPARQL sub-strings. For example, in the following grammar extract we show how the function *MQuery* that has been defined in the grammar to generate questions like: *what is the material of x, what are the colours of x* is constructed.

```
lin MQuery q = "PREFIX painting:
<http://spraakbanken.gu.se/rdf/owl/painting.owl#> $n
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> $n
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> $n
SELECT distinct"++ q.wh1 ++ "$n WHERE { $n
?painting rdf:type painting:Painting; $n
rdfs:label ?title; $n " ++ q.wh2 ++ "$n" ++ q.prop++"}" ;
```

The category *q* has three parameters: *wh1,wh2* and *prop*. *wh1* carries information about the ontology classes, i.e. material, museum, painter; *wh2* carries information about the ontology classes and properties assertions; and *prop* carries the same information as in *wh1* and *wh2* but has a different syntax.

With these grammars it is possible to generate the following outputs from one semantic representation that is available in the domain ontology: An example of a SPARQL query that is generated in response to the NL query *show everything about all miniatures* is specified below.[11]

```
SELECT distinct ?painting ?title ?material ?author ?year
 ?length ?height ?color ?museum $n
WHERE $ { ?painting rdf:type painting:MiniaturePainting . $n
 rdfs:label ?title ; $n painting:createdBy ?author; $n
 painting:hasMaterial ?material; $n
 painting:hasCurrentLocation ?museum; $n
 painting:hasCreationDate ?date; $n
 painting:hasDimension ?dim; $n
 painting:hasColor ?color; $n
 ?author rdfs:label ?painter . $n
 ?date painting:toTimePeriodValue ?year .
 ?dim painting:lengthValue
 ?length ; $n painting:heightValue ?height .
 ? museum rdfs:label ?loc. $n
  FILTER (lang(?title) = ’en’) } LIMIT 200
```

As opposed to the template based query mechanism, where NL sentences are just a short-cut to formulate SRPARQL sentences for non-expert users [22,23], the natural language query approach follows the WYSIWYM (what you see is what you meant) mechanism [24]; the user can formulate queries by clicking on a proposed feedback text. The interpretation of the sentence will derive a single semantic representation, which includes information about the intention of the sentence, its structure, the classes represented in it, and the parts that are to be looked for, and then translated into a SPARQL query which retrieves the results from the RDF knowledge base.

Because GF supports both parsing and generation, it is possible to generate one single SPARQL query from natural language or linearize natural language from a single SPARQL.

---

[11] The *$n* stands for new line identifier for the backend to post-process.

Text Module: SPARQL Results to NL The knowledge representation infrastructure returns RDF triples as results from a SPRAQL query. A coherent natural language description is generated from these triples. The *Text* module has been designed to generate a coherent natural language descriptions from a selected set of the returned triples. More specifically, our grammar covers eight classes that are most commonly used to describe a painting, including: *Title, Painter, Painting Type, Material, Colour, Year, Museum* and *Size*. Each of these classes is defined as category and is captured in one function *DPainting* which has the following representation in the abstract syntax.

```
DPainting :
   Painting -> Painter -> PaintingType ->
   OptColours -> OptSize -> OptMaterial ->
   OptYear  -> OptMuseum  -> Description ;
```

The function *DPainting* takes eight arguments of which five are optional, i.e. *OptColour, OptSize, OptMaterial, OptYear* and *OptMuseum*. Each of these categories can be left out in a text. The advantage is that with one function we are able to generate different descriptions depending on the information that is available about the retrieved painting. This approach allows for efficient multilingual linearizations, as opposed to the previous one [19] , where semantic patterns were defined with different functions and thus required an extensive linguistic effort to linearize. Below follow some examples of texts generated in English to exemplify the different descriptions we are able to generate from one single function call with a varying number of instantiated parameters.

- Interior was painted on canvas by Edgar Degas in 1868. It measures 81 by 114 cm and it is painted in red and white. This painting is displayed at the Philadelphia Museum of Art.
- Interior was painted by Edgar Degas in 1868. It measures 81 by 114 cm. This painting is displayed at the Philadelphia Museum of Art.
- Interior was painted on canvas by Edgar Degas in 1868. It is painted in red and white. This painting is displayed at the Philadelphia Museum of Art.
- Interior was painted by Edgar Degas. It measures 81 by 114 cm and it is painted in red and white. This painting is displayed at the Philadelphia Museum of Art.
- Interior was painted on canvas by Edgar Degas. It measures 81 by 114 cm and it is painted in red and white.
- Interior was painted by Edgar Degas in 1868. This painting is displayed at the Philadelphia Museum of Art.
- Interior was painted by Edgar Degas.

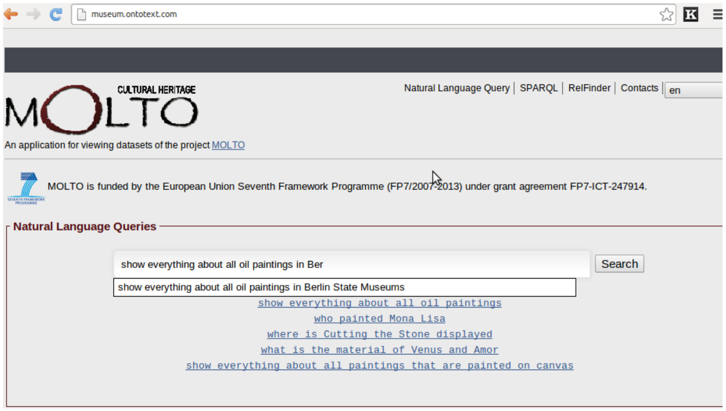
4 Museum Reason-able View Presentation

The Museum Reason-able view is presented by means of the Forest framework developed by Ontotext.[12] It has several features, such as SPARQL end point,

---

[12] `http://museum.ontotext.com`

keyword search with autocomplete, and relation finder. Figure 2 shows the initial page of the interface where exemplary natural language requests are provided.



museum.ontotext.com

An application for viewing datasets of the project MOLTO

MOLTO is funded by the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement FP7-ICT-247914.

Natural Language Queries

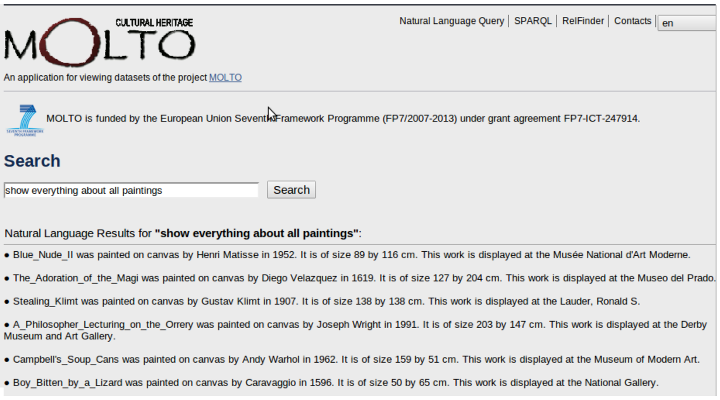
show everything about all oil paintings in Ber

show everything about all oil paintings in Berlin State Museums

`show everything about all oil paintings`
`who painted Mona Lisa`
`where is Cutting the Stone displayed`
`what is the material of Venus and Amor`
`show everything about all paintings that are painted on canvas`

Fig. 2. Initial page of the language to ontology showcase

Once placed in the search box with one click and executed, the results will appear in both the form of natural language text as shown in Figure 3, and RDF triples as shown in Figure 4.[13]



An application for viewing datasets of the project MOLTO

MOLTO is funded by the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement FP7-ICT-247914.

Search

show everything about all paintings

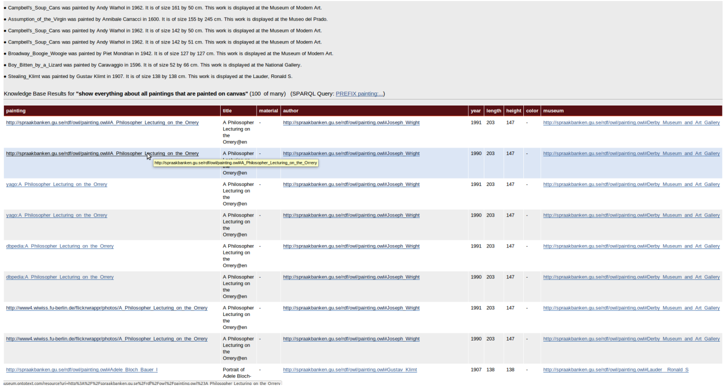
Natural Language Results for **"show everything about all paintings"**:

- Blue_Nude_II was painted on canvas by Henri Matisse in 1952. It is of size 89 by 116 cm. This work is displayed at the Musée National d'Art Moderne.
- The_Adoration_of_the_Magi was painted on canvas by Diego Velazquez in 1619. It is of size 127 by 204 cm. This work is displayed at the Museo del Prado.
- Stealing_Klimt was painted on canvas by Gustav Klimt in 1907. It is of size 138 by 138 cm. This work is displayed at the Lauder, Ronald S.
- A_Philosopher_Lecturing_on_the_Orrery was painted on canvas by Joseph Wright in 1991. It is of size 203 by 147 cm. This work is displayed at the Derby Museum and Art Gallery.
- Campbell's_Soup_Cans was painted on canvas by Andy Warhol in 1962. It is of size 159 by 51 cm. This work is displayed at the Museum of Modern Art.
- Boy_Bitten_by_a_Lizard was painted on canvas by Caravaggio in 1596. It is of size 50 by 65 cm. This work is displayed at the National Gallery.

Fig. 3. Natural language results

---

[13] The semantic data can be also extracted in JSON and XML format.



- Campbell's_Soup_Cans was painted by Andy Warhol in 1962. It is of size 161 by 50 cm. This work is displayed at the Museum of Modern Art.
- Assumption_of_the_Virgin was painted by Annibale Carracci in 1600. It is of size 155 by 245 cm. This work is displayed at the Museo del Prado.
- Campbell's_Soup_Cans was painted by Andy Warhol in 1962. It is of size 142 by 50 cm. This work is displayed at the Museum of Modern Art.
- Campbell's_Soup_Cans was painted by Andy Warhol in 1962. It is of size 142 by 51 cm. This work is displayed at the Museum of Modern Art.
- Broadway_Boogie_Woogie was painted by Piet Mondrian in 1942. It is of size 127 by 127 cm. This work is displayed at the Museum of Modern Art.
- Boy_Bitten_by_a_Lizard was painted by Caravaggio in 1596. It is of size 52 by 66 cm. This work is displayed at the National Gallery.
- Stealing_Klimt was painted by Gustav Klimt in 1907. It is of size 138 by 138 cm. This work is displayed at the Lauder, Ronald S.

Knowledge Base Results for **"show everything about all paintings that are painted on canvas"** (100 of many) (SPARQL Query: PREFIX painting:...)

painting	title	material	author	year	length	height	color	museum
http://spraakbanken.gu.se/rdf/owl/painting.owl#A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1991	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
http://spraakbanken.gu.se/rdf/owl/painting.owl#A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1990	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
yago:A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1991	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
yago:A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1990	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
dbpedia:A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1991	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
dbpedia:A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1990	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
http://www4.wiwiss.fu-berlin.de/flickrwrappr/photos/A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1991	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
http://www4.wiwiss.fu-berlin.de/flickrwrappr/photos/A_Philosopher_Lecturing_on_the_Orrery	A Philosopher Lecturing on the Orrery@en	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Joseph_Wright	1990	203	147	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Derby_Museum_and_Art_Gallery
http://spraakbanken.gu.se/rdf/owl/painting.owl#Adele_Bloch_Bauer_I	Portrait of Adele Bloch-	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Gustav_Klimt	1907	138	138	-	http://spraakbanken.gu.se/rdf/owl/painting.owl#Lauder__Ronald_S

http://spraakbanken.gu.se/rdf/owl/painting.owl#A_Philosopher_Lecturing_on_the_Orrery

useum.ontotext.com/resource?uri=http%3A%2F%2Fspraakbanken.gu.se%2Frdf%2Fowl%2Fpainting.owl%23A_Philosopher_Lecturing_on_the_Orrery

Fig. 4. Semantic repository results

Similar to any other regular SPARQL end point, SPARQL queries can be formulated and executed via a RESTFul API,[14] using the SPARQL query interface.[15] The interface is supplied with a set of exemplary queries, which are predefined, and serve as models and hints about the kind of queries that can be formulated.

5 Related Work

Natural language to ontology interoperability is a rather new research area. It has several aspects which reflect the different ways it can be used, e.g. building natural language interfaces which allow to link the word meanings, inherit the relationships based on the existing structure and deal with ambiguities more effectively, making use of reasoning capabilities and efficient production of lexicons, powering question answering tasks and improving user experience. Semantic technologies present another aspect and challenge in this last mentioned field, which is the focus of interest in this chapter. They require not only mapping from formal language representations to ontological classes and properties, but also constructing valid SPARQL queries that are intended to provide answers to questions that are executed on a semantic database. The advances of ontology to SPARQL interoperability have been tested in three consequent QALD challenges.[16] [17] [18]

In these challenges the different approaches to handle the natural language to SPARQL interoperability differ from each other in the way they interpret the

---

[14] `http://museum.ontotext.com/owlim/repositories`

[15] http://museum.ontotext.com/sparql

[16] `http://www.sc.cit-ec.uni-bielefeld.de/qald-1`

[17] `http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/index.php?x=challenge&q=2`

[18] `http://greententacle.techfak.uni-bielefeld.de/~cunger/qald/`

natural language input and in the way they produce the SPARQL query to provide the query results. For instance, [25] present a 5 step question answering architecture, e.g. 1 question parsing and query template generation; 2 lookup in an inverted index; 3 string similarity computation; 4 lookup in a lexical database; and 5 semantic similarity. In their approach, the SPARQL template is generated within the first step during parsing, and the layered approach, as the authors claim, helps to identify the effectiveness and the efficiency of each consecutive step. The string similarity steps make use of vector spaces, and are used to match the strings from the natural language input to DBpedia Uniform Resource Identifier (URI) candidates.

The approach taken by [23] is based on translating natural language questions to RDF triple patterns using the dependency tree of the question text, and relational patterns extracted from the Web. Their system relies on processing the RDF predicates in a form that is comparable with the syntactic output, which makes it data source dependent.

Many authors rely on multi-layered ontology approach for generating multilingual descriptions [26,27,28,29,30]. These approaches require extensive linguistic data associated with the ontology classes and properties. There is no attempt to generate descriptions in real time from a large set of ontologies. In the context of cultural heritage there have also been some attempts to generate natural language from ontologies using controlled natural language mechanism [31].

Our approach differs from the above approaches as it offers abstract semantic representations to SPARQL interoperability by enabling cross-language interaction using GF. In addition, it constructs answers in the form of coherent texts, by contrast to other approaches which generate at most single grammatical sentences.

6 Conclusions

This chapter presented an approach for natural language to ontology interoperability that is employed for multilingual interaction with Semantic Web knowledge bases and Linked Open Data. It is based on the assumption that ontologies restricts the semantic queries that can be formulated over them. The grammar formalism chosen, GF provides with the means to cover nearly 30 languages, which makes the transition from natural language expressions in multiple languages to their interoperability with the semantic web data seamless.

The division between the abstract and the concrete syntaxes provided by GF has been exploited to convert Semantic Web based representations to multilingual natural language. The grammar is successfully used by the cross-language retrieval system and offers natural language to ontology interoperability.

The chapter explains the approach on a use case from the cultural heritage domain, and shows a full cycle of natural language interaction, both querying and results description, over semantic web knowledge infrastructure.

Acknowledgment

This work is supported by MOLTO European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement FP7-ICT-247914.

References

1. Eric, P., Andy, S.: SPARQL. the query language for RDF (2008) `http://www.w3.org/TR/rdf-sparql-query/`.
2. Lassila, O., Swick, R.R.: Resource Description Framework (RDF). Model and Syntax Specification. (1999) `http://www.w3.org/TR/REC-rdf-syntax`.
3. Ranta, A.: The GF resource grammar library. The on-line journal Linguistics in Language Technology (LiLT) **2** (2009) `http://elanguage.net/journals/index.php/lilt/article/viewFile/214/158`.
4. Bishop, B., Kiryakov, A., Ognyanoff, D., Peikov, I., Tashev, Z., Velkov, R.: OWLIM: A family of scalable semantic repositories. Semantic Web Journal, Special Issue: Real-World Applications of OWL (2011)
5. Kiryakov, A., Ognyanoff, D., Velkov, R., Tashev, Z., Peikov, I.: Ldsr: Materialized reason-able view to the web of linked data. In (Ed.), I.R.H.P.S., ed.: Proceedings of OWLED 2009, Chantilly, USA (2009)
6. Damova, M., Kiryakov, A., Grinberg, M., Bergman, M.K., Giasson, F., Simov, K.: Creation and integration of reference ontologies for efficient LOD management. In: Semi-Automatic Ontology Development: Processes and Resources, IGI Global, Hershey PA, USA (2012)
7. Ranta, A.: Grammatical Framework, a type-theoretical grammar formalism. Journal of Functional Programming **14** (2004) 145–189
8. Berners-Lee, Tim and Hendler, James and Lassila, Ora: The Semantic Web. Scientific American (2001) 30–37
9. ter Horst, H.J.: Combining RDF and Part of OWL with Rules: Semantics, Decidability, Complexity. In: Proceedings of The Semantic Web ISWC. Volume 3729 of LNCS., Heidelberg, Springer Berlin (2005) 668–684
10. Damova, M., Dannélls, D.: Reason-able View of Linked Data for cultural heritage. In: Proceedings of the third International Conference on Software, Services and Semantic Technologies (S3T). (2011)
11. W3C OWL Working Group: OWL Web Ontology Language Overview. (2012) `http://www.w3.org/TR/owl2-overview/`.
12. Hyvyonen, E., Maekelae, E., Salminen, M., Valo, A., Viljanen, K., Saarela, S., Junnila, M., Kettula, S.: Museum finland. In: Finnihs Museum on the Semantic Web. (2008)
13. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F., eds.: The Description Logic Handbook: Theory, Implementation, and Applications. In Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F., eds.: Description Logic Handbook, Cambridge University Press (2003)
14. Brickley, D., Guha, R.: RDF Vocabulary Description Language 1.0: RDF Schema. W3C. (2004) `http://www.w3.org/TR/rdf-schema/`.
15. Ranta, A.: Grammatical Framework: Programming with Multilingual Grammars. CSLI Studies in Computational Linguistics. CSLI, Stanford (2011)
16. Martin-Löf, P.: Intuitionistic type theory. Bibliopolis, Napoli (1984)

17. Dannélls, D., Damova, M., Enache, R., Chechev, M.: A Framework for Improved Access to Museum Databases in the Semantic Web. In: Recent Advances in Natural Language Processing (RANLP). Language Technologies for Digital Humanities and Cultural Heritage (LaTeCH). (2011)
18. Dannélls, D.: On generating coherent multilingual descriptions of museum objects from semantic web ontologies. In: Proceedings of the Seventh International Natural Language Generation Conference (INLG 2012), Utica, IL, Association for Computational Linguistics (2012) 76–84
19. Dannélls, D., Damova, M., Enache, R., Chechev, M.: Multilingual online generation from semantic web ontologies. In: Proceedings of the World Wide Web Conference (WWW2012), Lyon, France (2012)
20. Ranta, A., Angelov, K.: Implementing controlled languages in gf. In: Lecture notes in computer science. CNL-2009 Controlled Natural Languages. (2010) 82–101
21. Kaljurand, K., Kuhn, T.: A multilingual semantic wiki based on Attempto Controlled English and Grammatical Framework. In: Proceedings of the 10th Extended Semantic Web Conference (ESWC 2013), Springer (2013)
22. Unger, C., Buehmann, L., Lehmann, J., Ngomo, A.C.N., Gerber, D., Cimiano, P.: Template-based question answering over rdf data. In: Proceedings of the 21st international conference on WWW'2012 - Ontology Representation and Querying: RDF and SPARQL, Genoa (2012)
23. Hakimov, S., Tunk, H., Akimaliev, M., Doglu, E.: Semantic question answering system over linked data using relational patterns. In: Proceedings of EDBT/ICDT, Genoa (2013)
24. Power, R., Scott, D., Evans, R.: What You See Is What You Meant: direct knowledge editings with natural language feedback. In: 13th European Conference on Artificial Intelligence (ECAI 1998), John Wiley and Sons (1998) 677–681
25. Walter, S., Unger, C., Cimiano, P., Baer, D.: Evaluation of a layered approach to question answering over linked data. In: Proceedings of ISWC 2012, Boston (2012)
26. Androutsopoulos, I., Kokkinaki, V., Dimitromanolaki, A., Calder, J., Oberl, J., Not, E.: Generating multilingual personalized descriptions of museum exhibits: The M-PIRO project. In: Proceedings of the International Conference on Computer Applications and Quantitative Methods in Archaeology. (2001)
27. O'Donnell, M.J., Mellish, C., Oberlander, J., Knott, A.: ILEX: An architecture for a dynamic hypertext generation system. Natural Language Engineering **7** (2001) 225–250
28. Androutsopoulos, S.K., Karkaletsis, V.: Exploiting OWL ontologies in the multilingual generation of object descriptions. In: The 10th European Workshop on NLG, Aberdeen, UK (2005) 150–155
29. Androutsopoulos, J.O., Karkaletsis, V.: Source authoring for multilingual generation of personalised object descriptions. Natural Language Engineering **13** (2007) 191–233
30. Bouayad-Agha, N., Casamayor, G., Mille, S., Rospocher, M., Saggion, H., Serafini, L., Wanner, L.: From Ontology to NL: Generation of multilingual user-oriented environmental reports. Lecture Notes in Computer Science **7337** (2012)
31. Damljanovic, D., Bontcheva, K.: Enhanced semantic access to software artefacts. In: Workshop on Semantic Web Enabled Software Engineering (SWESE) held in conjunction with ISWC. (2008)