Improving Multilingual DBpedia: Interlingual Property Mapping

Mart Busger op Vollenbroek, Olivier Louwaars, Xianchao Zeng
Information Science, Faculty of Arts
University of Groningen

Abstract

In this project the target is to improve multilingual DBpedia. The approach we are using is developing a multilingual mapping tool that automatically maps properties in different languages into the same ontology. For this research, the Dutch DBpedia was taken as sample (nl.dbpedia.org). There were two main problems found concerning the attributes at a Dutch DBpedia page. In the current situation, it is not possible to automatically update a value under a same property in different languages as the attribute names for the two values are not linked together in different languages. Also, searching DBpedia with a SPARQL query can result in empty answers because the specified name is not known. In this research it will be attempted to use semantic web knowledge to build up a mapping method between multilingual properties. The result shows a high feasibility on the attempt in both mapping the multilingual properties and improving the SPARQL search quality.

The software referred to in this article can be found at: https://github.com/Obipls/SWT-ass1

1 Introduction

DBpedia is a crowd-sourced community effort to extract structured information from Wikipedia and make this information available on the Web. DBpedia allows users to ask sophisticated queries against Wikipedia, and to link the different data sets on the Web to Wikipedia data. This work may make it easier for the huge amount of information in Wikipedia to be used in some new interesting ways. Furthermore, it might inspire new mechanisms for navigating, linking, and improving the encyclopedia itself.

The English version of the DBpedia knowledge base describes 4.58 million entities, out of which 4.22 million are classified in a consistent ontology, with 1.445.000 persons, 735.000 places (including 478.000 populated places), 411.000 creative works (including 123.000 music albums, 87.000 films and 19.000 video games), 241.000 organizations (including 58.000 companies and 49.000 educational institutions), 251.000 species and 6.000 diseases. In addition, they also provide localized versions of DBpedia in 125 languages. All these versions together describe 38.3 million entities, out of which 23.8 million are localized descriptions of things that also exist in the English version of DBpedia. Altogether the DBpedia 2014 release consists of 3 billion pieces of information (RDF triples) out of which 580 million were extracted from the English edition of Wikipedia, 2.46 billion were extracted from other language editions. (DBpedia Blog)

For the Dutch DBpedia, there has been significant progress on the mapping templates since its launch in 2013. So far, DBpedia has

been updated to version 4.0, which stems from September 2014.

1.1 DBpedia Ontology

The DBpedia Ontology is a shallow, cross-domain ontology, which has been manually created based on the most commonly used infoboxes within Wikipedia. The ontology currently covers 685 classes which form a subsumption hierarchy and are described by 2,795 different properties. Since the DBpedia 3.7 release, the ontology is a directed-acyclic graph, not a tree. Classes may have multiple superclasses, which was important for the mappings. A taxonomy can still be constructed by ignoring all superclasses except the one that is specified first in the list and is considered the most important. (DBpedia Ontology)

1.2 Property Mapping

DBpedia properties also have properties of their own and while those sub-properties are already mapped in the English DBpedia ontology, in other languages they are poorly mapped or sometimes not mapped at all. The problem with this is that the property itself exists in multiple languages, but does not 'know' that of itself. Hence, changing the value in one language does not change it in the rest. Therefore, it is desired to extend and map these sub-properties in DBpedia for other languages than English.

There are two main problems regarding a Dutch DBpedia page. The first problem is that properties of an entity are not annotated/linked in an ontology, while they are linked in the English version. The second problem is that different spelling of a same property exists in Dutch version, making the data inconsistent. In this situation, it is not applicable to use a certain language for searching using a SPARQL query, and be sure that all retrieved results are correct and complete.

With respect to these problems, the following research questions will be answered with this research:

1. How can ontologies and their methods in different languages be tied together (mapped)?
2. How is DBpedia being mapped, and how can these methods be used to contribute
3. How to conduct and analyze the evaluation for mapping the multilingual properties for a DBpedia ontology?

2 Method

For achieving the goal of mapping Dutch properties to the corresponding English ontology member, it has to be established that these two are indeed each other's counterparts. There are several ways to do this, such as automatic translation or assuming they are the same when they have the same value. As both of these methods have their flaws and benefits, the second way is the most universal applicable without relying on external sources. Also, it is believed that an automatic mapping system could profit from large collections of <entity, attribute, value> triples harvested from Wikipedia templates (Bouma, et al., 2009).

To justify the assumption that the same value belongs to the same attribute in different languages, as much doubt or ambiguity has to be taken away first in order to make sure that

the matching values belong to the same entity. For example, it is not unimaginable that one person died on the day that another person was born. Therefore the descriptions of the values in this research will only be matched if the DBpedia URL corresponds. The risk that two equal values show up on the same page is considered naught, and is favored of the risk of getting wrong translations using the other method.

The matching process of the value in the two languages will involve a *Python* script that loops through the two documents and creates a dictionary with the value and page URL as key, and the English and Dutch name for it as value. As the attributes of entire DBpedia take up several Gigabytes, a fast and fluent approach is needed in order to keep the runtime manageable. The only way to do this is by keeping all the data in one single dictionary, as otherwise two large files have to be compared line by line. This results in reading one file line by line for every line in the other file, which results in exponential longer runtime if more data is added.

Due to the fact that Dutch properties can have multiple meanings in English, and only one can have the 'sameAs' relation that will be assigned, the type of the page also has to be included. For example, the word 'doop' can mean baptism for a child and christening of commission date of a ship. With the use of the type, both can be kept in the mappings file. This way the method is wider applicable, as it is not restricted to one single type of entity, such as a person.

All necessary information can be found in the files of the DBpedia dump, where three Dutch files contain all required input:

- *Mappingbased-properties_nl.nt*

Contains the Dutch entity URL, the ontology attribute and the value

- *Infobox-properties_nl.nt*

Contains the Dutch entity URL, the Dutch property attribute and the value

- *Instance-types_nl.nt*

Contains the Dutch entity URL, the RDF Type identifier and the ontology attribute

Combining all three of these files will create a symmetrical mapping file that links the Dutch property to the official English DBpedia ontology for a certain type of page. For the development of the system not all entries in these files are needed, as the full runtime will be too long for convenience. An arbitrary selection will be sufficient, for instance all attributes that contain a date.

The evaluation of the experiment will be done by comparing two methods: manually annotate how many cases are mapped right in the raw output files, versus filtering single cases out and then annotate. After all, it is reasonable to assume that if a mapping is wrong, it will be because the value was with the wrong attribute. These cases will be very rare mistakes, and will probably not happen more than once per page type.

3 Results

After running the two slightly different variations of the program (named system 1

and system 2), some results were gathered. These results can be found in table 1.

	System 1	System 2
Amount of triples	320	241
Accuracy	92,81%	94,19%
Runtime (in minutes)	25	0,10

Table 1: System results comparison

Aside from the difference in the amount of triples found by both variations, the biggest difference between the variations is the runtime. While the first system took 25 minutes after several improvements were already made, the second system only took 10 seconds. The difference between these runtimes can be explained by the methods being used. Both systems use dictionaries as datatypes for storing the information, but the first system has to make several more iterations for every triple in the file, which takes a long time. By altering system 1 into system 2 making use of optimization with several iterations, the runtime was drastically decreased.

However, by using this new method the amount of triples also decreased. The reason for this is that while reviewing the results from system 1, it became clear that there were many triples that only occurred once in the dataset. These triples were therefore excluded from the results.

One thing worth mentioning is that the results above are achieved using only a small portion of the total data dump provided by DBpedia - only the date properties were used. Seeing as the first system used 25 minutes for only those properties, it is unlikely that such a system is suited for mapping all the Dutch properties. The second system however, with higher accuracy and a lower runtime shows promise for mapping all the Dutch properties to their English equivalents. Annotating for those results has not been done because it was deemed too time consuming.

Seeing as both systems do not achieve a 100% accuracy score, a closer look at the errors is required. While doing that, two different categories of errors have been discovered. The first category contains errors of triples in which the two properties have nothing to do with each other. Some examples of this kind of error are:

SoccerPlayer <http://nl.dbpedia.org/property/clubupdate> dbpedia-owl:sameAs <http://dbpedia.org/ontology/deathDate>

Aircraft <http://nl.dbpedia.org/property/eerstevlucht> dbpedia-owl:sameAs <http://dbpedia.org/ontology/retired>

The second category contains errors for which the property is simply to vaguely translated. These errors occur more often than errors from the first category. Examples from this category are:

Settlement <http://nl.dbpedia.org/property/datum> dbpedia-owl:sameAs <http://dbpedia.org/ontology/populationAsOf>

SubMunicipality <http://nl.dbpedia.org/property/datum> dbpedia-owl:sameAs <http://dbpedia.org/ontology/populationAsOf>

4 Conclusion

To summarize, we have tried to map Dutch DBpedia properties to their English equivalents using a sameAs property to link them. This has

been done with two slightly different approaches to see which approach worked better. Because it became clear that the first method had a relatively high runtime, we chose to focus on a particular subset of all the properties in the Dutch DBpedia. Using this approach we achieved a runtime of 25 minutes using one approach, while the other could do it in merely 10 seconds. Both systems achieved a relatively high accuracy, but the system does not work perfectly. The main reason for not scoring 100% accuracy is the vaguely defined Dutch properties, which are not clear enough to map them by simply using a sameAs property.

Using this approach, different language properties can be mapped to their English equivalent. Our approach has been tested with the Dutch DBpedia, but because our approach is language independent it can be extended to other languages. The main advantage of doing this is making it possible to use queries in native languages instead of having to use English properties for the queries.

5 Related Work

Our approach gained inspiration from the previous introduction of the DBpedia mapping method (Jens Lehmann et al. 2012). In the original DBpedia working ontology, the structured data (presented in RDF triples) is extracted and mapped with an extraction framework. Similar work on automatically mapping templates has also been conducted (Alessio et al.2012).

The mapping-based infobox extraction uses manually written mappings that relate infoboxes in Wikipedia to terms in the DBpedia ontology. The mappings also specify a date-type for each infobox property and thus help the extraction framework to produce high quality data. A mapping template is developed to fulfill the needs when mapping the infobox data values. A mapping assigns a type from the DBpedia ontology to the entities that are described by the corresponding infobox. In addition, attributes in the infobox are mapped to properties in the DBpedia ontology.

By looking through the previous work, this seems still workable in practice. In this project, the focus was more on the detailed mapping between different named properties which contain the same value in the same format. It can be seen as relation extraction task. Because a same data value for multilingual properties has been formatted in the first mapping-based extraction. Through the improvement in the experiment, the results turn out to be acceptable.

References

[1] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N.Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, So¨ ren Auer, Christian Bizer. *In proceeding of Semantic Web 1 (2012) 1–5, DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia.*

[2] Gosse Bouma and Sergio Duarte. *In proceeding of ACL, 2009*. Wikipedia entity retrieval for Dutch and Spanish

[3] Alessio Palmero Aprosio, Claudio Giuliano, Alberto Lavelli. *Automatic Mapping of Wikipedia Templates*
for Fast Deployment of Localised DBpedia datasets.

[4] DBpedia Blog.
DBpedia Version 2014 released. [Online]
http://blog.dbpedia.org/?p=77.

[5] DBpedia.
DBpedia. [Online] http://dbpedia.org/about.

[6] DBpedia Ontology.
DBpedia Ontology. [Online]
http://wiki.dbpedia.org/services-resources/ontology.