Enriching an Academic Knowledge base using Linked Open Data

Chetana Gavankar[1,2] *Ashish Kulkarni*[1]
Yuan − Fang Li[3] *Ganesh Ramakrishnan*[1]
(1) IIT Bombay, Mumbai, India
(2) IITB-Monash Research Academy, Mumbai, India
(3) Monash University, Melbourne, Australia
chetana@cse.iitb.ac.in, kulashish@gmail.com, yuanfang.li@monash.edu, ganesh@cse.iitb.ac.in

ABSTRACT
In this paper we present work done towards populating a domain ontology using a public knowledge base like DBpedia. Using an academic ontology as our target we identify mappings between a subset of its predicates and those in DBpedia and other linked datasets. In the semantic web context, ontology mapping allows linking of independently developed ontologies and inter-operation of heterogeneous resources. Linked open data is an initiative in this direction. We populate our ontology by querying the linked open datasets for extracting instances from these resources. We show how these along with semantic web standards and tools enable us to populate the academic ontology. Resulting instances could then be used as seeds in spirit of the typical bootstrapping paradigm.

KEYWORDS: Ontology Mapping, Academic knowledge base, linked open data, DBpedia.

Proceedings of the Workshop on Speech and Language Processing Tools in Education, pages 51–60,
COLING 2012, Mumbai, December 2012.

1 Introduction

Ontologies and knowledge bases play an important role in semantic web. This has led to an independent and distributed effort of developing several domain ontologies and public knowledge bases. In this context, ontology mapping enables interlinking of different ontologies and their population by exploiting similarities between predicates. Prior research discusses several approaches to ontology population ranging from automated to semi-supervised. Bootstrapping approach (Agichtein and Gravano, 2000; Mintz et al., 2009) to ontology population often makes use of a small number of seed examples for each predicate in an ontology. Generating these seeds could benefit from availability of a mapping between a domain ontology and a knowledge base like DBpedia. Using an academic ontology as our target, we study this approach to ontology population and propose a query based formulation to map nodes from the academic ontology to those of the DBpedia ontology. For *e.g.*, *Journal* node from the academic ontology could be mapped to the *Academic Journal* concept in DBpedia. Similarly *Software, Person, Programming Language* etc. have corresponding mappings to nodes in DBpedia.

We will now introduce some basic definitions for setting the context of our work. We will then list the important contributions of our work.

(Flahive et al., 2011) defines *Ontologies* as concepts and relationships used to describe and represent an area of knowledge. An ontology is made up of a set of concepts, properties, property mappings and relationships between the concepts. Concepts are the nodes or objects that identify something that exists. Set of relationships relate two concepts within an ontology. They can either link two concepts together or loop back and link to the same concept. Properties provide extra features used to identify the concept. The property mapping element is similar to a relationship element, but it links a property to a concept rather that one concept to another .

Ontology population primarily concerns itself with the identification of instances for classes in an ontology. It is a knowledge acquisition activity that relies on semi-automatic methods to transform unstructured, semi-structured and structured data sources into instance data.

(Zhang et al., 2012) define *Ontology mapping* as follows. Given two ontologies O_1 and O_2, mapping one ontology onto another means that for each entity *e1* (concept, relation, or instance) in an ontology O_1, we find a corresponding entity *e2*, which has the same intended meaning, in an ontology O_2. $map(e1_i) = e2_j$

The primary contributions of this paper are the following:

- Development of an academic domain ontology.
- Identification of nodes in an academic ontology to be populated using external data sources and mapping to DBpedia ontology nodes.
- Ontology population using SPARQL queries against the DBpedia and other linked datasets.

2 Related Work

We relate our work to the existing work in the area of ontology population to generate academic knowledge base.

Ontology population involves building and populating an ontology from structured, semistructured and unstructured text. There is a large body of work in ontology population (Brunzel,

2008; Poesio and Almuhareb, 2008; Maynard et al., 2008; De Boer et al., 2007) that uses frequency based term extraction along with shallow NLP techniques. However, enterprise data usually does not have the luxury of highly redundant data exploited in the above approaches. Ontology building and population from collaborative resources such as Wikipedia has developed a great interest in researchers across the world. There are various publicly available data sources built using these collaborative resources on the linked data cloud such as YAGO, DBpedia. YAGO2 (Hoffart et al., 2011, 2010; Suchanek et al., 2008; de Melo et al., 2008) is a Geo-spatial ontology built automatically from Wikipedia, GeoNames, and WordNet. It contains 80 million facts about 9.8 million entities. DBpedia data set (Bizer et al., 2009b; Auer et al., 2008; Morsey et al., 2012) consists of RDF triples extracted from the "infoboxes" commonly seen on the right hand side of Wikipedia articles, while Geonames [1] provides RDF descriptions of millions of geographical locations worldwide. These collaborative resources are also used for bootstrapping the ontology population process. Ontological smoothing (Zhang et al., 2012) uses a semi-supervised technique that learns extractors for a set of minimally-labeled relations. It uses the few examples to generate a mapping from the target relation to a database view over a background knowledge base, such as Freebase. It then queries the background knowledge base to retrieve many more instances that are deemed similar to those of the target relation and the system learns the extractor. Our work is influenced the most from this approach. However we choose to use DBpedia and the datasets linked to it as the source knowledge base due to its higher degree of overlap with the academic ontology. We also differ in the mapping technique and write several manual SPARQL queries against the DBpedia SPARQL endpoint[2] to extract instances to be used as seeds in populating the academic ontology.

3 Academic Ontology

Ontology building for a specific domain can start from scratch or by modifying an existing ontology. We built our academic ontology using existing *Benchmark* [3] and *Aisso*[4] ontologies. Ontologies are merged using the *Protege*[5] ontology editor and extended to include several classes like *award*, *project* etc. and attributes like *professor* has *research-area*, *course* has *prerequisite* etc. In addition, we scraped the glossary lists available in Wikipedia to populate class hierarchy rooted at the *concept* class. An ontology that we finally used consists of more than 190 classes, 150 object properties and 150 data properties. Please refer figure 1 for the snapshot of some nodes in an academic ontology.

4 Ontology Mapping

The preliminary step in extracting instances from external data sources is mapping of nodes in academic ontology with external ontologies. We use ontology mapping between academic ontology and DBpedia ontology. The DBpedia Ontology is hand-made with 205 ontology classes. We first identify the nodes in academic ontology such as *academic conferences* to be populated using an external knowledge bases. We then identify mappings between these nodes and its relational properties with those in DBpedia. The mapping between external data sources and academic ontology involves mapping between nodes, its data properties and object properties. Though the names of concepts in an ontology match, it may not be exact mapping due to

[1] http://www.geonames.org/ontology/
[2] http://dbpedia.org/sparql
[3] http://swat.cse.lehigh.edu/onto/univ-bench.owl
[4] http://vocab.org/aiiso/schema
[5] http://protege.stanford.edu

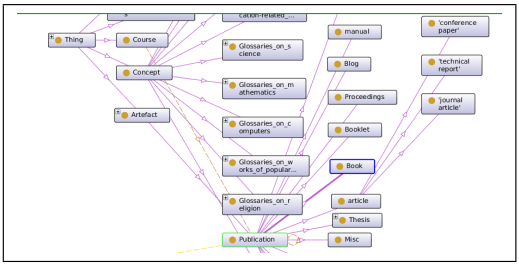

Figure 1: Academic Ontology snapshot of some nodes in ontology

different interpretation in the respective ontologies. Mapping can either be between nodes using equivalence class or subclass or super class relation. All the data properties in the academic ontology may not have corresponding mapping in DBpedia ontology. Conversely all data properties of DBpedia may not have corresponding mapping in Academic ontology like software programming language. There are other issues like the names of label for nodes or property may be different but they mean the same. Example: Label 'location' in DBpedia ontology is same as label 'venue' in academic ontology. Mapping between object property of academic ontology and DBpedia has an additional constraint of checking the domain and range of an object property in both ontologies. We used the above heuristics for ontology mapping between academic ontology and DBpedia ontology. Please refer table 1 for the mapping between academic domain and DBpedia ontology classes.

Academic ontology nodes	DBpedia ontology nodes	Academic ontology properties	DBpedia ontology properties
Book	Book	author, title,abstract, type, date,isbn, publisher	author, title, abstract, type, date, issn, publisher
Event	Event	date, title, location	date, title, venue, event period, committe
Conference	Conference	date, title, location	date, title, venue, event period, committe
Workshop	Workshop	date, title, location	date, title, venue, event period, committe
Journal	AcademicJournal	date, title	publication date, title, ranking
Software	Software	Programming Language, computing platform	Programming Language, computing platform
Programming Language	Programming Language	name	name
Glossaries_on_mathematics	Mathematical_terminology	terms	terms
Glossary_of_graph_theory	Glossary_of_graph_theory	terms	terms
Glossary_of_education-related_terms	Glossary_of_education-related_terms	terms	terms

Table 1: Ontology mapping for sample nodes from Academic ontology to DBpedia ontology

5 Ontology Population

We then search the required entities on linked open data to locate the relevant data source. Due to the openness of this LOD data sources, it is difficult to know data sources relevant for query answering. We use web interface, *open link software* [6] to ease the task of finding relevant data source. The results for a sample search for *glossary of mathematics* are displayed in the figure refer figure 2.

Subsequent to data source searching , we query these resources to extract the relevant instances. Data on the linked open data cloud (Bizer et al., 2009a) are expressed using resource description framework (RDF) or web ontology language OWL. RDF is a directed, labeled graph data format for representing information in the Web. SPARQL protocol and RDF query language (SPARQL)

[6] http://dbpedia.org/fct/

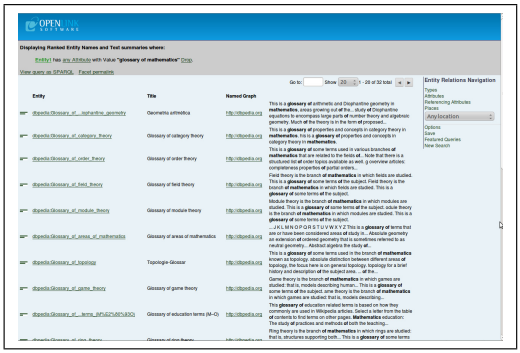

Figure 2: Link open data search results for glossary of mathematics

can be used to express queries across diverse data sources, whether the data is stored natively as RDF or viewed as RDF via middleware. We populate our ontology by querying the linked open datasets using SPARQL for extracting the instances from these RDF resources on the LOD cloud. We wrote and executed SPARQL queries through DBpedia SPARQL endpoint [7]. Refer figure 3 for the results from a sample SPARQL query. The SPARQL queries return a set of instances to populate nodes in academic ontology. Refer Appendix A for set of sample SPARQL queries for extracting instances from linked open data. Resulting instances could then be used as seeds in spirit of the typical bootstrapping paradigm.

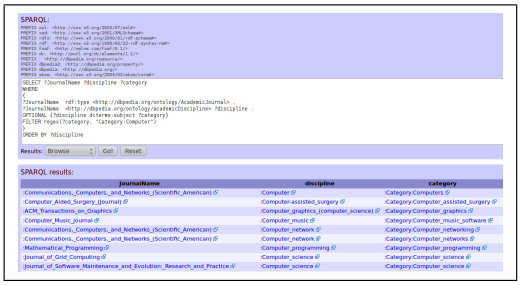

Figure 3: Sample SPARQL query execution

6 Evaluation

The purpose of evaluation was to ascertain the correctness of instances extracted from the linked open data for ontology population. We indexed corpus of three major universities obtained by crawling their pages.We then queried this index for each instance obtained from the linked open data and recorded the top 10 results. We scanned these results to check support for that instance in context of the category being populated. Table 2 summarizes the results of our evaluation for a subset of nodes in our academic ontology [8].

[7] http://dbpedia.org/snorql/

[8] rough estimate for *Softwares* based only on number of search results

Academic ontology node	Extracted	In Corpus	In-context	Precision
AIConferences	7	7	6	0.86
BotanyBooks	32	4	3	0.75
ChemistryJournals	152	94	92	0.98
EngineeringJournals	61	48	42	0.88
ProgramingLanguages	288	183	142	0.78
ComputerScienceConferences	34	26	26	1
ComputerScienceBooks	68	33	18	0.55
GlossaryofMathematics	111	73	58	0.80
GlossaryofMathematicalConcepts	22	18	17	0.95
GlossaryofPredicateLogic	28	11	11	1
Softwares	27947	4386	4326	1

Table 2: Number of instances found in corpus out of the total number of instances obtained by querying linked open data for a subset of nodes in the academic ontology

Conclusion and Future work

In this work, we showed the feasibility of exploiting overlaps between a domain ontology and public knowledge bases using a query based mapping formulation. In particular, we wrote several SPARQL queries against the DBpedia datasets to extract instances for the predicates in our academic ontology. In the process, we studied different types of mappings between ontology predicates ranging from mapping one concept to a combination of many others to mapping different types of predicates. The instances thus extracted could serve as seeds in bootstrapping the ontology population process. That forms the direction of our future research. Such a populated academic knowledge base could be leveraged in information extraction and retrieval applications built over academic corpora.

References

Agichtein, E. and Gravano, L. (2000). Snowball: extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*, DL '00, pages 85–94, New York, NY, USA. ACM.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2008). DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference (ISWC)*, volume 4825 of *Lecture Notes in Computer Science*, pages 722–735. Springer.

Bizer, C., Heath, T., and Berners-Lee, T. (2009a). Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009b). Dbpedia - a crystallization point for the web of data. *Web Semant.*, 7(3):154–165.

Brunzel, M. (2008). The xtreem methods for ontology learning from web documents. In *Proceedings of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pages 3–26, Amsterdam, The Netherlands, The Netherlands. IOS Press.

De Boer, V., Van Someren, M., and Wielinga, B. J. (2007). Relation instantiation for ontology population using the web. In *Proceedings of the 29th annual German conference on Artificial intelligence*, KI'06, pages 202–213, Berlin, Heidelberg. Springer-Verlag.

de Melo, G., Suchanek, F. M., and Pease, A. (2008). Integrating YAGO into the Suggested Upper Merged Ontology. In *20th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2008)*.

Flahive, A., Taniar, D., Rahayu, J. W., and Apduhan, B. O. (2011). Ontology expansion: appending with extracted sub-ontology. *Logic Journal of the IGPL*, 19(5):618–647.

Hoffart, J., Suchanek, F. M., Berberich, K., Lewis-Kelham, E., de Melo, G., and Weikum, G. (2011). Yago2: exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 229–232, New York, NY, USA. ACM.

Hoffart, J., Suchanek, F. M., Berberich, K., and Weikum, G. (2010). YAGO2: a spatially and temporally enhanced knowledge base from Wikipedia. Research Report MPI-I-2010-5-007, Max-Planck-Institut für Informatik, Stuhlsatzenhausweg 85, 66123 Saarbrücken, Germany.

Maynard, D., Li, Y., and Peters, W. (2008). Nlp techniques for term extraction and ontology population. In *Proceedings of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pages 107–127, Amsterdam, The Netherlands, The Netherlands. IOS Press.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA. Association for Computational Linguistics.

Morsey, M., Lehmann, J., Auer, S., Stadler, C., and Hellmann, S. (2012). Dbpedia and the live extraction of structured data from wikipedia. *Program: electronic library and information systems*, 46.

Poesio, M. and Almuhareb, A. (2008). Extracting concept descriptions from the web: the importance of attributes and values. In *Proceedings of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pages 29–44, Amsterdam, The Netherlands, The Netherlands. IOS Press.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2008). YAGO: A Large Ontology from Wikipedia and WordNet. *Elsevier Journal of Web Semantics*.

Zhang, C., Hoffmann, R., and Weld, D. S. (2012). Ontological smoothing for relation extraction with minimal supervision. In *AAAI*.

A Categories of SPARQL queries

List of computer science conferences. Similar query can be written for other area conferences

```
SELECT ?Conferences
WHERE
```

```
{
?Conferences rdf:type <http://dbpedia.org/class/yago/ComputerScienceConferences> .
}
```

Query for getting list of softwares and details

```
SELECT ?Software ?programmingLanguage ?latestVersion
WHERE
{
?Software rdf:type <http://dbpedia.org/ontology/Software> .
?Software <http://dbpedia.org/ontology/programmingLanguage> ?programmingLanguage .
OPTIONAL {?Software <http://dbpedia.org/property/latestReleaseVersion> ?latestVersion}
}
LIMIT 40
```

Discipline and categories of journal filter using computer. Similarly can be obtained for other categories like physics

```
SELECT ?s ?discipline ?category
WHERE
{
?s rdf:type <http://dbpedia.org/ontology/AcademicJournal> .
?s <http://dbpedia.org/ontology/academicDiscipline> ?discipline .
OPTIONAL ?discipline dcterms:subject ?category
FILTER regex(?category, "Category:Computer")
}
ORDER BY ?discipline
```

Query for list of programming languages

```
SELECT ?ProgrammingLanguage ?version ?OperatingSystem
WHERE
{
?ProgrammingLanguage rdf:type <http://dbpedia.org/ontology/ProgrammingLanguage> .
?ProgrammingLanguage <http://dbpedia.org/ontology/latestReleaseVersion> ?version .
?ProgrammingLanguage <http://dbpedia.org/property/operatingSystem> ?OperatingSystem .
}
LIMIT 100
```

Query for topics along with category and its related category

```
SELECT ?topic ?category ?relatedCategory
WHERE
{
?wikipage foaf:primaryTopic ?topic .
?topic dcterms:subject ?category .
OPTIONAL?category skos:related ?relatedCategory
```

```
FILTER regex(?category, "Category:Computer")
} LIMIT 20
```

To find available category and their broader category

```
SELECT ?isValueOf ?broaderCategory
WHERE
{
?isValueOf rdf:type <http://www.w3.org/2004/02/skos/core#Concept> .
?isValueOf skos:broader ?broaderCategory
}
LIMIT 20
```

Query of list of Academicjournals along with discipline and impact factor

```
SELECT ?s ?d ?i
WHERE
{
?s rdf:type <http://dbpedia.org/ontology/AcademicJournal> .
?s <http://dbpedia.org/ontology/academicDiscipline> ?d
?s <http://dbpedia.org/ontology/impactFactor> ?i
}
ORDER BY ?d
LIMIT 20
```

Query for list of projects

```
SELECT ?projectName ?objective ?keyword ?start ?end ?fundedBy
WHERE
{
?projectName rdf:type <http://dbpedia.org/ontology/Project> .
OPTIONAL{?projectName dbpedia-owl:projectObjective ?objective .}
OPTIONAL{?projectName dbpedia-owl:projectKeyword ?keyword .}
OPTIONAL?projectName dbpedia-owl:projectStartDate ?start .}
OPTIONAL{?projectName dbpedia-owl:projectEndDate ?end.}
OPTIONAL{?projectName dbpedia-owl:fundedBy ?fundedBy .}
}
```

To find all the available disciplines

```
SELECT DISTINCT ?discipline
WHERE
{
?s <http://dbpedia.org/ontology/academicDiscipline> ?discipline .
}
LIMIT 100
```

Finding wikipedia outlinks and redirects

```
SELECT ?resource ?redirects ?outLinks
WHERE { ?resource <http://dbpedia.org/ontology/wikiPageRedirects> ?redirects.
?resource <http://dbpedia.org/ontology/wikiPageExternalLink> ?outLinks
}
LIMIT 20
```