A systematic property mapping using category hierarchy and data

Kalpa Gunaratna, Sarasi Lalithsena, Prateek Jain, Cory Henson, and Amit Sheth

Ohio Center of Excellence in Knowledge Enabled Computing (Kno.e.sis)
Wright State University, Dayton, Ohio 45435
{kalpa,sarasi,prateek,cory,amit}@knoesis.org

Abstract. *Relationships play a key role in Semantic Web to connect the dots between entities (concepts or instances) in a way that enables to absorb the real sense of the entities. Even though relationships are important, it is difficult to categorize or identify them because they consist of complex knowledge in the schema. Therefore systematically identifying relationships yield many advantages and open doors for new research avenues. In this work, we try to identify a specific type of relationship (part of) in a multi-domain dataset with a shallow schema in Linked Open Data. We used DBpedia dataset and devised an algorithm using Wikipedia to identify patterns of part of relationships in the dataset. This paper is based on some in progress initial work based on identifying part of relationships.*

Keywords: Property Matching, Linked Open Data

1 Introduction

The most significant aspect of Semantic Web is how things are related to each other. To express this relatedness, we use properties in triples to connect subject and object which leads to a meaningful representation of knowledge in a more structured way. Hence properties take a major part in Semantic Web and Ontologies. By being the most important fact of Semantic Web properties/relationships still there is very few work carried out based on the relationships compared to concepts. Finding facts about relationships by itself can lead to interesting research and development based questions [8].This paper is based on some in progress work based on identifying part-of relationships from ontologies.

Linked Open Data (LOD), also called as web of data, is a growing cloud of datasets in the web. Important fact in LOD is, datasets in LOD are interconnected with each other by forming a giant graph that consists of large and rich information. By being a giant data graph LOD have the capability to exchange information from different data sources and also integrate data in a meaningful way. In order to achieve that ontology alignment and matching systems is a must. In almost every system of these kinds only concentrate on matching concepts (or classes in the schema). In the recent past, many efforts have been taken to match and align concepts in different ontologies. BLOOMS [6] is one such successful effort with respect to LOD. But there is no effort taken yet (to the best of our knowledge) to map properties to more richer relationships like part-of, causality and etc. There exists some limited manual mappings of properties: DBpedia to Pronto, etc.

Motivation

There is no systematic way of mapping properties in an ontology instead of using equivalence or sub property mappings. For many applications and research, concept mapping and equivalent or sub property mapping alone is not sufficient. Therefore there is a need to have a property mapping system for ontologies. If a property mapping could be achieved for an ontology then it provides more coherent connection between the ontologies than having just a concept mapping. Property is the connection path between a subject and object in an ontology. Therefore property is a function of mapping a subject to an object. If we can map a property in one ontology in to another ontology then it means that the same meaning is present for the connection of subject and object in the second ontology. Hence whenever the mapped property occurs in the second ontology, it infers some additional information or knowledge that we possess on second ontology using first ontology. For example, if we know a property is mapped to part-of relationship then we know that the property's subject actually participates in a part to whole relationship with the object.

Mapping properties in to more richer relationships like part-of, causality, membership and etc would be more useful in aligning ontologies to upper level ontologies like SUMO, OpenCyc, UMBEL and Proton. With that kind of a mapping these upper level ontologies can serve as an unifying schema for the LOD data sets. It can lead to applications such as querying across LOD data set that have numerous different schemas. Furthermore identifying part-of relationships would lead to come up with better interference mechanism as well.

Objective

The goal of this work is to find a way to identify relationships that has the property of part of relationship. In other words, mapping relationships to part-of property.

2 Background

To best of our knowledge, there is no any known mechanism in automatically identifying part of relationship in a systematic way with or without using background knowledge. There are various discussions going on in order to identify part-of relationships and also to define the part-of relationship due to the lack of a formal definition to the part-of relationship. But all these discussions are theoretical but still unimplemented. [10] classified part-of relation in to six relation types named component-integral object, member-collection, portion-mass, stuff-object, feature-activity and place-area. [9] came up with a definition to understand the part-of relationship for the purpose of supporting development of ontologies in scientific research especially for biomedical ontologies. It defines part-of relationship between two concepts say A, B by considering the instance-level part relationships of instances of A and B.

In the past years many methods have been used in mapping ontologies and most of these mappings/alignments have been done based on the concepts of the ontologies. Most of these systems are based on content based similarities and structure-based similarities. In addition to that some approaches use machine learning and rule based approaches as well [5] [3] [4]. Some ontology alignment systems follow semi-automatic approach for mapping on ontologies by utilizing the user's feedback for the alignment process [1][7]. Some of the above approaches like [1] focus their attention to mapping properties but they just use equivalence properties and sub property relationships in mapping. Ontology mapping becomes becomes non trivial when consider the shallow schemas in Linked Open Data. BLOOMS [6] is a system designed to map schemas in LOD by utilizing the background knowledge from Wikipedia and they also showed that none of the existing ontology matching mechanisms works well with the schema information in LOD. In case of BLOOMS it also map the concepts in two ontologies using equivalence and subsumption relationships and does not focus on mapping properties.

The problem tries to address from this paper is different from other mapping approaches discussed above. While other ontology mapping systems try to map two ontologies, this approach would try to map properties in the ontology to part-of relationships. Such kind of a mapping would be useful when mapping properties in ontology to an upper level ontology like SUMO, OpenCyc, UMBEL or Proton. Out of these upper level ontologies it is only Proton that maintains manually mapped upper level mappings to DBpedia. This mapping contains 27 mappings at the relationship level only with two part-of relationships location and foundationPlace[2].

3 Problem Description

Linked Open Data cloud has many data sets in domain specific and multi-domain categories. DBpedia dataset can be considered as a multi-domain dataset in LOD. Identifying part-of relationship in a multi-domain dataset (in DBpedia) is the focus of this work which will be extended to identify a set of other relationships as well.

Idea

in a part-whole relationship, there should be a matching concept between the part and the whole. If we take wheel and car, both participate in a part-whole relationship that wheel is essential for the car to function in a normal way. Therefore we can take the same basic idea to devise a process to identify part-whole relationships automatically. That is to take the subject and also the object and see whether to have some common concept which provides evidence for a part-whole relationship. If we further simply the concept, when we generalize the subject at some point it will be nearly equal to the object. If that is the case, we take this evidence as a strong heuristic for the availability of part-whole relationship between the subject and object.

3.1 Datasets and issues

Linked open data cloud has many datasets and we chose DBpedia [1] for this work because it contains information about many different topics and titles (domains). Also DBpedia does not have a well structured schema compared to other ontologies in LOD.

Shallow schema

Shallow schema are schema that do not have ontology rules defined in the schema. For example, DBpedia schema does not have property restrictions specified in the ontology like transitive, reflexive, etc. Even though DBpedia schema does not mention about transitive rule for properties, we have found out that in the data level there exists instances for transitive relationships. The relationship called *http://dbpedia.org/ontology/partOfWineRegion* has many transitive instances in the data level but it is not mentioned in the schema.

Furthermore domain or/and range are also not mentioned for the many properties in the dataset. For example, *http://dbpedia.org/ontology/alliance* property does not have a range specified. It is extremely difficult to work with a shallow schema since it does not provide much information that it could have but yet the dataset may be very useful like DBpedia.

Patterns in the dataset

When schema does not provide evidence for particular property rules in the dataset, we can still approximate them as mentioned above which helps in getting useful information. For example, we can find patterns for transitive relationships using data instances by checking whether a relationship exists between an instance A and B, B and C and also A and C.

Even though we are able to approximate certain relationships using data instances it is depends on the availability of data level instances for a specific pattern. Hence there are examples when we expect a particular relationship to possess some characteristics by its name but unable to find any supporting evidence from data instances. The property called *http://dbpedia.org/ontology/isPartOf* is expected to be a part of relationship and also a partial order relationship but

[1] http://wiki.dbpedia.org/Datasets

lacks at least one transitive pattern in the data level. A partial order relationship has reflexive, anti-symmetric and transitive characteristics.

4 Design and Implementation

Identifying part-whole relationships results in identifying whether the object comprises of the subject. For that we need to figure out that the subject is in some kind of semantic relationship with the object which conveys that the whole has the part. To identify this semantic relationship we have used somewhat similar approach that [6] follows.

For a property P,domain D and range R of P, we build two trees considering D and R as two roots.

- To build the trees(tree T_D built from Domain D and T_R built from range R), we used Wikipedia category hierarchy. Start from domain D or range R and search for categories of the domain or range for the first time and add them as its children of the tree. Then for each of its children, categories are searched and added as subsequent children. This process is followed until it reaches the specified depth of the tree.
- After two trees are built, we start from T_D and search for a common category in T_R less than the current level of T_D (always by difference of 2).
- If at least one common category is found then we decide that the domain D participates in a part-whole relationship with range R.

We have decided that the depth d of each tree at most should be 5 and corresponding search on T_R should not go beyond 3 and should be less than current level i of T_D in all cases.

Omitting patterns

We have seen a pattern in dataset that whenever a superlative is included in a property (relationship). For example, *http://dbpedia.org/ontology/fastestDriver* property can never be a part-whole relationship because the property itself conveys the message that the object participated in the triple is going to be the extreme end which will most probably not the whole and may be more specific than the subject. So we maintain a list of superlative keywords and if any of them appear in a property, we decide that it is not going to be a part-whole relationship.

If a property has the meaning of some human behaviour then the property is mentioning something other than part-whole relationship (like member of, etc). For example, *http://dbpedia.org/ontology/vicePresident* is about representing a vice president and most probably not to do anything about part-whole relationship.

In this DBpedia dataset properties with sub class of person class in subject and/or object is omitted because the dataset does not have such information. For example, DBpedia dataset does not have triples explaining human hand is a part of human (person) but nerve is a part of human anatomy.

Using patterns

As mentioned earlier partial order properties are a super set of part of properties. If partial order properties can be identified correctly, then part of properties are definitely a sub set of that. Partial order theory can be used to identify partial order properties which can be specified as, for a property p and instances a,b,c

- reflexive (a p a)
- anti-symmetric (a p b and not b p a)
- transitive (if a p b and b p c then a p c exists)

Since DBpedia schema is shallow and can not guarantee the availability of the patterns in the data instances, we used a relaxed transitivity to provide evidence for part of relationship. That is we only look for a pattern *a p b and b p c* and take that as a supporting evidence/confidence for a part of relationship.

4.1 Process of steps

for the system being implemented, we follow the following steps in the order given as at now,

1. For each object type property of DBpedia, check for an existence of a human behaviour pattern and superlative word in the property name and if so discard the property. Then check for sub class of reasoning for domain and range with class person and if it is positive then discard.
2. If a property is neither of the above then build a Wikipedia category tree for domain and range. If a common category is found in the range tree for the tree of domain and also the property has the relaxed transitive characteristic in the data level then identify the property as a part of relationship.
3. All other properties are not part-whole relationships.

5 Evaluation

For the DBpedia dataset, there are 585 object properties and out of those we have manually identified 30 of them as part of relationships.

6 Results and Discussion

When the system is run for the DBpedia object properties it was able to identify 7 of them out of 10. We have ignored the other 20 (which were manually identified) because either they did not have domain and/or range specified or domain and/or range specified as the owl:Thing. Hence for the initial system implementation we only considered the properly specified properties.

precision : 7/36 = 0.19 recall : 7/10 = 0.70 f measure : 0.29

Recall is quite high but precision is low. (here we have to claim that getting recall high is important as then we can manually identify and filter out incorrect results from a reasonably short list or properties than manually identifying all).

Discussion

In the result set we were unable to get the most promising properties like *http://dbpedia.org/ontology/partOf* and *http://dbpedia.org/ontology/partOfWineRegion* for having domain and range the same class.

The first intention was to get the partial order properties using partial order theory and then add a further filtering to get part of relationships only from that. Getting partial order properties confronted with issues of not having transitive instances in the dataset and existence of few symmetric instances. Therefore identifying partial order properties was not very successful with the partial order theory. It identified very few instances like 2 or 3 of them.

Using WordNet to analyse the meaning of the property name whether it contains a part-holonym word to categorize as a part-whole relationship alone was not sufficient since it gives more incorrect results than correct results. It identified partOf and partOfWineRegion properties as mentioned above correctly which can not be processed using the category hierarchy because of their same domain and range specifications. But a property like *http://dbpedia.org/ontology/sisterCollege* was also identified as a part of relationship because of the word college in sister-College (we normalised the string and tokenized for WordNet processing). Hence more work needs to be done if WordNet to be used as a filter.

7 Conclusion and Future work

In this initial work the results seem promising in identifying part of relationships from building a relationship between the domain and range of a property using Wikipedia resources. The work is in-progress in building more filters to support part of relationships and filtering out other properties. Approximating domain and range for a property when those details are not present is required. It should also be done when the domain and range are the same. Even though the domain and range are the same, we may be able to approximate concepts suiting its hidden characteristics using data instances.

We wish to extend this process to identify many other properties as member of, causal, etc relationships which are useful in inference algorithms. The advantage of this system becomes more apparent when we are to identify relationship categories in a huge dataset (having many properties) like DBpedia.

References

1. Cruz, I., Antonelli, F., Stroe, C.: Agreementmaker: efficient matching for large real-world schemas and ontologies. Proceedings of the VLDB Endowment 2(2), 1586–1589 (2009)
2. Damova, M., Kiryakov, A., Simov, K., Petrov, S.: Mapping the central lod ontologies to proton upper-level ontology. Ontology Matching p. 61 (2010)
3. David, J., Guillet, F., Briand, H.: Matching directories and owl ontologies with aroma. In: Proceedings of the 15th ACM international conference on Information and knowledge management. pp. 830–831. ACM (2006)

4. Doan, A., Madhavan, J., Domingos, P., Halevy, A.: Ontology matching: A machine learning approach. Handbook on Ontologies pp. 385–516 (2004)
5. Giunchiglia, F., Shvaiko, P., Yatskevich, M.: S-match: an algorithm and an implementation of semantic matching. The semantic web: research and applications pp. 61–75 (2004)
6. Jain, P., Hitzler, P., Sheth, A., Verma, K., Yeh, P.: Ontology alignment for linked open data. The Semantic Web–ISWC 2010 pp. 402–417 (2010)
7. Noy, N., Musen, M.: Algorithm and tool for automated ontology merging and alignment. In: Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-00). Available as SMI technical report SMI-2000-0831 (2000)
8. Sheth, A., Arpinar, I., Kashyap, V.: Relationships at the heart of semantic web: Modeling, discovering, and exploiting complex semantic relationships. Enhancing the Power of the Internet (2003)
9. Smith, B.: Beyond concepts: ontology as reality representation. In: Formal ontology in information systems: proceedings of the third conference (FOIS-2004). p. 73. Ios Pr Inc (2004)
10. Winston, M., Chaffin, R., Herrmann, D.: A taxonomy of part-whole relations**. Cognitive science 11(4), 417–444 (1987)