Type Prediction for Entities in DBpedia by Aggregating Multilingual Resources

Thi-Nhu Nguyen[1,4], Hideaki Takeda[2],
Khai Nguyen[2,3], Ryutaro Ichise[2], Tuan-Dung Cao[4]

[1] Haiphong University, Vietnam
nhunt@dhhp.edu.vn
[2] National Institute of Informatics, Japan
{takeda,nhkhai,ichise}@nii.ac.jp
[3] University of Science, VNU-HCMC, Vietnam
nhkhai@fit.hcmus.edu.vn
[4] Hanoi University of Science and Technology, Vietnam
dungct@soict.hust.edu.vn

Abstract. The entity type is considered as very important in DBpedia. Since this information is inconsistently described in different languages, it is difficult to recognize the most suitable type of an entity. We propose a method to predict the entity type based on a novel conformity measure. We combine the consideration of the specific-level and the majority voting. The experiment result shows that our method can suggest informative types and outperforms the baselines.

Keywords: DBpedia, Ontology, Mappings, Conformity, Consistency

1 Introduction

DBpedia is built upon the community effort to extract the knowledge from Wikipedia [1]. Currently, contributors from many countries have joined the DBpedia mapping project, whose target is to map the Wikipedia templates into the types (e.g., `Species`, `Person`, and `Place`) in DBpedia ontology [2]. Despite the maturity of the DBpedia community, the lack of consensus between the contributors from different languages is still remaining as an issue.

In DBpedia, a real-world entity is represented by multiple instances. Each instance is described in a specific language and its type is based on the mappings constructed for that language. Because the mappings are manually created for different languages, the types of particular instances are different even when those instances describe the same entity. Concretely, considering an entity, some types may be different at the specific-level, correct, or incorrect. For example, the entity of Barack Obama is recognized as `Person`, `Politician`, `President`, `Artist` and `Book` in 29 languages. Here, there is an agreement between `Person`, `Politician`, and `President` but still different at the specific levels. Meanwhile, `Artist` and `Book`

Table 1. The statistics of the agreement of types between two languages

Languages		# instances have type in both languages	% instances have the same type
nl	sv	308462	21.86%
en	nl	201248	53.28%
en	sv	158842	8.47%
en	es	149234	30.92%
it	en	144815	10.59%
en	sv	158842	8.47%
nl	es	143634	77.53%
pt	en	132773	68.55%
nl	it	130938	8.83%
pl	it	118935	8.75%

are incorrect. In this situation, choosing the most suitable type of an entity is necessary to guarantee the consolidation of DBpedia but it becomes a difficult task.

According to a preliminary analysis, the agreement of type assignment among different languages is low, even if only comparing two particular languages. Table 1 illustrates the percentage of instances sharing the same type in 10 language pairs, in which the number of instances had a type in both languages are the most among all 476 language pairs. In general, only 37% pairs have more than 50% of instances assigned with the same type.

Recently, entity type prediction is considered as an important problem. It is helpful for the utility of DBpedia versions whose mapping community is immature. In addition, it is also the core of automatic mapping creation [3].

The simple ideas of type prediction are majority voting and most specific ancestor. The disadvantage of majority voting is that the suggested type is not specific enough for an entity. The most specific ancestor even returns more general types.

In this paper, we propose a new method to predict the most suitable type of an entity. Our method is the improvement of majority voting. In detail, we focus on how to retrieve more specific types.

2 Type suggestion

In this section, we describe how to predict the most suitable type for an entity. Our idea is based on the combination of the specific-level and the majority voting. The input is a set of the most specific-level types assigned by different languages. We define the conformity $Con(x)$ of the most specific-level type x. The conformity is a recursive value taking the sum of the frequency of x and the conformity of its parent.

$$Con(x) = frequency(x) + Con\big(parent(x)\big) \qquad (1)$$

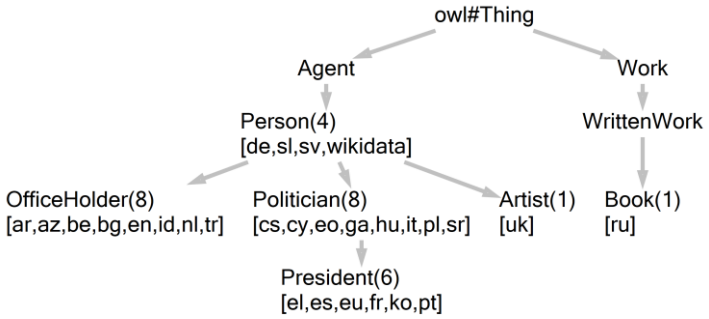

Fig. 1. All types of Barack Obama entity and their frequency.

Where the $frequency$ of x is the number of languages treating the entity as x. For an entity, we select the most suitable type by picking the one with the highest conformity. Obviously, this chosen type will meet the condition that it is used in the most languages and also enough specific. If there are many types that have the same highest conformity, we rank the type based on the conformity of their parent type. Let's consider the example in Fig.1. In this figure, the entity of Barack Obama is assigned to 6 types in 29 languages. The conformity of the type `President` is the highest (Con(`President`) =18). Therefore, it is selected as the prediction result.

3 Experiment and evaluation

We compare our method with a manually crafted dataset and two other baselines: (1) majority voting and (2) most specific ancestor. We build an entity database from all available language versions of DBpedia. An entity is the compilation of the instances interconnected via `owl:sameAs` links. The difficulty of type suggestion is the diversity of types. Therefore, we select the entities with high inconsistency. Concretely, we first randomly select 500 entities whose type is available in at least 5 languages. Then, we pick up the 100 most inconsistent ones. Here, the inconsistency is estimated by the entropy of types' frequency. Different from the conformity in Eq. 1, in order to guarantee the hierarchical relations, transitive types are counted. In which, transitive types are a set of ancestor types. After the selection, an expert is asked to assign the most suitable type among available of the entity (3). Finally, we compare the results of our method, (1), (2) against (3).

Table 2 implies that our method gives the best result. As entities of high inconsistency are selected, the most specific ancestor method always chooses the `owl:Thing`, which is the root of the DBpedia ontology. Although majority voting is better than the most specific ancestor, in general, its result is not specific enough. This experiment demonstrates our prediction method is good but still 45% of the predicted types are different from human's opinion. Most of them belong to types of place entity because among countries, the definitions of administrative region and residential area different. DBpedia ontology currently lacks types to represent all

Table 2. The accuracy of type suggestion methods

Our method	Majority voting	Most specific ancestor
55%	44%	0%

these dissimilarities. For example, Voultegon is a commune in France but there is no type for commune. Therefore, this entity should be mapped to `Settlement` type. However, our method returns the inaccurate type `City` because this type is more specific than `Settlement`.

4 The demo

We build a tool to visualize the entity types. A user can input the keywords in any language or a URI to query the entity. The database contains 86,290,758 entities, which are constructed from 128,866,644 instances of all languages. We use Lucene[1] to have the entities indexed by all labels (i.e., `rdfs:label`) provided in all languages. We build a tool named MLDQ[2] to visualize hierarchically types in different languages, the suggested types of our method and other baselines, and some general information of the entity (e.g., the entropy of inconsistency).

5 Conclusion and future work

In this work, we proposed a new method that combines the consideration of the specific-level and the majority voting to suggest the most suitable type of an entity. Three methods were evaluated and the results show that our method is the most promising one although it remains some weaknesses. For future work, we will evaluate our method with deeper analyses, including comparisons to more baselines. We also aim to improve our method by considering the conformity of transitive types in order to give more accurate predictions.

References

1. Lehmann, J.; Isele, R.; Jakob, M; et al.: DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. The Journal Semantic Web – Interoperability, Usability, Applicability. vol. 6, no. 2, pp. 167-195. (2015)
2. Mendes, PN.; Jakob, M.; Bizer, C.;. DBpedia: A Multilingual Cross-Domain Knowledge Base. In Proceedings of the Eighth International Conference on Language Resources and Evaluation, pp. 1813-1817. (2012).
3. Palmero, A.; Giuliano, C.; Lavelli, A.: Automatic Mapping of Wikipedia Templates for Fast Deployment of Localised DBpedia datasets. In Proceedings of the 13th International Conference on Knowledge Management and Knowledge Technologies, pp. {1:1-1:8} (2013).

[1] https://lucenenet.apache.org/

[2] https://sites.google.com/site/iswc2016demo/