Web Semantics: Science, Services and Agents on the World Wide Web 39 (2016) 47–61

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

LHD 2.0: A text mining approach to typing entities in knowledge graphs

Tomáš Kliegr [a,b,*,1], Ondřej Zamazal [a,1]

[a] Department of Information and Knowledge Engineering, Faculty of Informatics and Statistics, University of Economics, Prague, nám. W Churchilla 4, 13067, Prague, Czech Republic

[b] Multimedia and Vision Research Group, Queen Mary, University of London, 327 Mile End Road, London E1 4NS, United Kingdom

ARTICLE INFO

Article history:
Received 23 July 2015
Received in revised form
10 May 2016
Accepted 10 May 2016
Available online 2 June 2016

Keywords:
Type inference
Support Vector Machines
Entity classification
DBpedia

ABSTRACT

The type of the entity being described is one of the key pieces of information in linked data knowledge graphs. In this article, we introduce a novel technique for type inference that extracts types from the free text description of the entity combining lexico-syntactic pattern analysis with supervised classification. For lexico-syntactic (Hearst) pattern-based extraction we use our previously published Linked Hypernyms Dataset Framework. Its output is mapped to the DBpedia Ontology with exact string matching complemented with a novel co-occurrence-based algorithm STI. This algorithm maps classes appearing in one knowledge graph to a different set of classes appearing in another knowledge graph provided that the two graphs contain common set of typed instances. The supervised results are obtained from a hierarchy of Support Vector Machines classifiers (hSVM) trained on the bag-of-words representation of short abstracts and categories of Wikipedia articles. The results of both approaches are probabilistically fused. For evaluation we created a gold-standard dataset covering over 2000 DBpedia entities using a commercial crowdsourcing service. The hierarchical precision of our hSVM and STI approaches is comparable to SDType, the current state-of-the-art type inference algorithm, while the set of applicable instances is largely complementary to SDType as our algorithms do not require semantic properties in the knowledge graph to type an instance. The paper also provides a comprehensive evaluation of type assignment in DBpedia in terms of hierarchical precision, recall and exact match with the gold standard. Dataset generated by a version of the presented approach is included in DBpedia 2015.

1. Introduction

One of the most important pieces of information in linked data knowledge graphs is the *type* of the entities described. The next generation linked open data enabled applications, such as entity classification systems, require complete, accurate and specific type information. However, many entities in the most commonly used semantic knowledge graphs miss a type. For example, DBpedia 3.9 is estimated to have at least 2.7 million missing types with the percentage of entities without any type being estimated at 20% [1]. Type inference has thus received increased attention in the recent years, with the approaches proposed taking either of the two principal paths: statistical processing of information that is already present in the knowledge graph, or extraction of additional types from the free text. In this article we introduce a novel technique for type inference which combines lexico-syntactic analysis of the free text and machine learning. This combined approach can complete types for about 70% of Wikipedia articles without a type in DBpedia.

Our previously published Linked Hypernyms Dataset (LHD) framework [2] extracts types from the first sentence of Wikipedia articles using lexico-syntactic patterns. In this work we extend it with Statistical Type Inference (STI) which helps to map LHD results to the DBpedia Ontology used by the native DBpedia solution. STI algorithm is a generic co-occurrence-based algorithm for mapping classes appearing in one knowledge graph to a different set of classes appearing in another knowledge graph provided that the two knowledge graphs contain common set of instances. In our setup, our target knowledge graph is DBpedia, and the source knowledge graph is LHD.

* Corresponding author at: Department of Information and Knowledge Engineering, Faculty of Informatics and Statistics, University of Economics, Prague, nám. W Churchilla 4, 13067, Prague, Czech Republic.

E-mail addresses: tomas.kliegr@vse.cz (T. Kliegr), ondrej.zamazal@vse.cz (O. Zamazal).

[1] Both authors contributed equally.

http://dx.doi.org/10.1016/j.websem.2016.05.001
1570-8268/

There are many articles for which lexico-syntactic patterns fail to extract any type. To address this, we employ Support Vector Machines (SVMs) trained on the bag-of-words representation of short abstracts and categories of Wikipedia articles. This supervised machine learning approach gives us a second set of entity type assignments.

In order to exploit the complementary character of the co-occurrence based STI algorithm and the supervised SVM models, we implement an ontology-aware fusion approach based on the multiplicative scoring rule proposed for hierarchical SVM classification. The hSVM algorithm can also be used separately as a language independent way to assign types since it uses abstract or categories as input feature set and it does not require language-specific preprocessing.

We validate our work on DBpedia 2014 [3], one of the most widely used Wikipedia-based knowledge graphs, the algorithmic approach is applicable also to the YAGO knowledge base [4], as well as to other semantic resources which contain instances (entities) that are (a) classified according to a taxonomy, and (b) described with a free text definition.

The evaluation of our algorithms is performed on DBpedia using a gold standard dataset comprising more than 2000 entities annotated with types from the DBpedia ontology using a crowdsourcing service.

The dataset generated with an earlier version of our approach is part of the DBpedia 2015-04 release as *Inferred Types LHD* dataset.

Parts of the work presented in this article have been published within the conference paper "Towards Linked Hypernyms Dataset 2.0: complementing DBpedia with hypernym discovery and statistical type inference (Kliegr and Zamazal, 2014)" [5]. This article extends the conference paper by introducing the hierarchical SVM approach and by performing extensive evaluation on the contributed gold standard dataset allowing the community to track progress in accuracy and coverage of entity typing and extraction tools. Also, the review of related work was substantially expanded.

The article is organized as follows. Section 2 gives an overview of related work, focusing on approaches for inference of entity types in DBpedia. Section 3 gives an overview of our approach. Section 4 describes how our LHD framework extracts types from the first sentence of Wikipedia articles and disambiguates them to DBpedia concepts. Section 5 presents the proposed algorithm for statistical type inference. Section 6 introduces the hierarchical support vector machines classifier. Section 7 describes the fusion algorithm. Section 8 presents the evaluation on the crowdsourced content and comparison with the state-of-the-art SDType algorithm and the DBpedia infobox-based extraction framework. The conclusions provide a summary of the results and an outlook for future work.

2. Related work

Completing missing types based on statistical processing of the information already present in the knowledge graph is in current research approached from several directions: (a) RDFS reasoning, (b) obtaining types through the analysis of the unstructured content with patterns, (c) machine learning models trained on labeled data, (d) unsupervised models that perform inference from statistical distributions of types, instances and the relations between them.

The four approaches listed above are covered in Sections 2.1–2.4. Section 2.5 covers the comparison of our STI/hSVM with SDType, which is a state-of-the-art unsupervised algorithm actually used for type inference in DBpedia 3.9 and DBpedia 2014. Section 2.6 motivates our choice of hSVM as a suitable machine learning classifier. Since we perceive the crowdsourced gold standard as an important element of our contribution, Section 2.7 reviews methods and resources for evaluation of algorithms that assign types to DBpedia entities. Table 1 gives an overview of selected related algorithms in terms of the methods and input features used and provides a comparison with our solution described in this article. A recent broader overview of approaches for knowledge graph refinement is present in [6].

2.1. RDFS reasoning

The standard approach to the inference of new types in semantic web knowledge graphs is RDFS reasoning. There are two general requirements enabling RDFS reasoning. First, these graphs need to have *domain* and *range* for properties specified and, second, they need to contain the corresponding *RDF facts* employing the defined properties. However, since according to common ontology design best practices (e.g. in Noy et al. [11]), domain and range should be defined in a rather general way, the inferred types tend not to be very specific. Also, *type propagation* goes upward along the taxonomy as a result of interaction of the subsumption knowledge from the ontology with the RDF facts from a dataset. Hence, RDFS reasoning usually cannot infer a specific type (i.e. type low in the hierarchy).

Furthermore, it is well known that RDFS reasoning approach will not correctly work for problems where the knowledge graph contains false statements (which is the case for DBpedia), since the errors are amplified in the reasoning process. Additional discussion on unsuitability of reasoners for type inference in DBpedia has been presented by Paulheim and Bizer in [8].

2.2. Pattern-based analysis of unstructured content

Major semantic knowledge graphs DBpedia and YAGO are populated from the *semistructured data* in Wikipedia—infoboxes and article categories using extraction framework that primarily relies on hand-crafted patterns. Approaches that extract types from the *free text* of Wikipedia articles can be used to assign types to articles for which the semistructured data are either not available, or the extraction for some reason failed.

The analysis of the unstructured (free text) content also often involves hand-crafted patterns. Tipalo, presented by Gangemi et al. in [7], covers the complete process of generating types for DBpedia entities from the free text of Wikipedia articles using a set of heuristics based on graph patterns. The algorithm starts with identifying the first sentence in the abstract which contains the definition of the entity. In case a coreference is detected, a concatenation of two sentences from the article abstract is returned. The resulting natural language fragment is deep parsed for entity definitions.

Our STI component uses as input types that were extracted from the free text with lexico-syntactic patterns with the Linked Hypernyms Dataset extraction framework presented in [12]. This framework proceeds similarly with Tipalo in that it extracts the hypernym directly from the POS-tagged first sentence and then links it to a DBpedia entity.

The accuracy of LHD matches the results for Tipalo algorithm – as reported by its authors in [7] – for the type selection subtask (0.93 precision and 0.90 recall). A detailed comparison between LHD and Tipalo is presented in [2] as well as a more extensive literature review on pattern-based extraction.

A conceptual disadvantage of pattern-based approaches is that they require relatively complex natural language processing pipeline, which is costly to adapt for a particular language. In contrast, the hSVM approach that we introduce in this article has essentially no language-specific dependencies, apart from basic tokenization, which makes its portability to another language comparatively straightforward.