# DWS at the 2016 Open Knowledge Extraction Challenge: A Hearst-like Pattern-Based Approach to Hypernym Extraction and Class Induction

Stefano Faralli and Simone Paolo Ponzetto

Research Group Data and Web Science, University of Mannheim {stefano,simone}@informatik.uni-mannheim.de

Abstract. In this paper we present a system for the 2016 edition of the Open Knowledge Extraction (OKE) Challenge. The OKE challenge promotes research in automatic extraction of structured content from textual data and its representation and publication as Linked Data. The proposed system addresses the second task of the challenge, namely "Class Induction and entity typing for Vocabulary and Knowledge Base enrichment" and combines state of the art lexical based Natural Language Processing techniques with lexical and semantic knowledge bases to first extract hypernyms from definitional sentences and second select the most suitable class of the extracted hypernyms from those available in the DOLCE foundational ontology.

**Keywords:** Linked Open Data, Hearst patterns, Hypernym extraction, Class induction

## 1 Introduction

Open Knowledge Extraction (OKE) is a recently introduced paradigm [23] that involves algorithms for Open Information Extraction (OIE) and automated taxonomy learning, which were shown able to extract information on a Web scale basis in an unsupervised manner. In OKE, lexicalized extractions are usually induced into a semantic model and linked to the Linked Open Data (LOD) cloud to serve as enriched resource for state-of-the-art Natural Language Processing (NLP) applications. Nuzzolese et al. [13] organized in 2015 the first OKE challenge. The OKE challenge promotes the research in automatic extraction of structured content from textual data and its representation and publication as Linked Data. In this paper we present our system for the second edition of the challenge, in particular for the task number 2 namely "Class Induction and entity typing for Vocabulary and Knowledge Base enrichment". Our approach combines three main aspects:

1. Natural Language Processing (NLP) state of the art methodologies of hypernym extraction and class induction;

- 2. A fresh new dataset<sup>1</sup> [18] of lexical "isa" relations extracted using Hearst's like pattern from the widest publicly available crawl of the Web [11];
- 3. Existing Linked Data resources such as WordNet [4] and OntoWordnet [5] to target the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)[10].

The rest of the paper is organized as follows: in Section 2 we give an overview of the literature on taxonomic relation extraction and induction. In Section 3 we describe our approaches to hypernym extraction (see Section 3.1) and class induction (see Section 3.2). In Section 4 we describe the results of a preliminary evaluation of the proposed algorithms on a gold standard dataset available from the challenge official Web-site; Finally we draw some conclusions on the presented work in Section 5.

#### 2 Related work

The extraction of taxonomic relations (also known as "isa" or hypernym relations) from text is a long-standing challenge for NLP and is a basic step for higher-end knowledge acquisition tasks such as ontology learning, see e.g. Biemann [2] for a survey. The manual construction of taxonomies is a very demanding task, requiring a large amount of time and effort. Ontology learning aims at reducing such cost of manual annotation by automatically or semi-automatically creating a lexicalized taxonomies using textual data from corpora or the Web [15, 2, 3, 9, 16]. As a result, the heavy requirements of manual ontology construction are drastically reduced.

The specific literature on hypernym extraction offers a high variability of methods, from simple lexical patterns [6, 14], similar to those used in our method, all the way through more complex statistical techniques [1, 17].

Snow et al. [20] perform the hypernym extraction by first collecting sentences that contain two terms that are known to be hypernyms. They parse the sentences, and extract patterns from the parse trees. Finally, they train a hypernym classifier based on these features and apply it to text corpora.

Yang and Callan [25] presented a semi-supervised taxonomy induction framework that integrates co-occurrence, syntactic dependencies, lexical-syntactic patterns and other features to learn an ontology metric, calculated in terms of the semantic distance for each pair of terms in a taxonomy. Terms are incrementally clustered on the basis of their ontology metric scores.

Snow et al. [19] perform incremental construction of taxonomies using a probabilistic model. They combine evidences from multiple supervised classifiers trained on large training datasets of hyponymy and co-hyponymy relations. The taxonomy learning task is defined as the problem of finding the taxonomy that maximizes the probability of individual relations extracted by the classifiers.

Kozareva and Hovy [8] start from a set of root terms and use Hearst-like lexico-syntactic patterns to harvest new terms from the Web, which results in

<sup>&</sup>lt;sup>1</sup> http://webdatacommons.org/isadb/

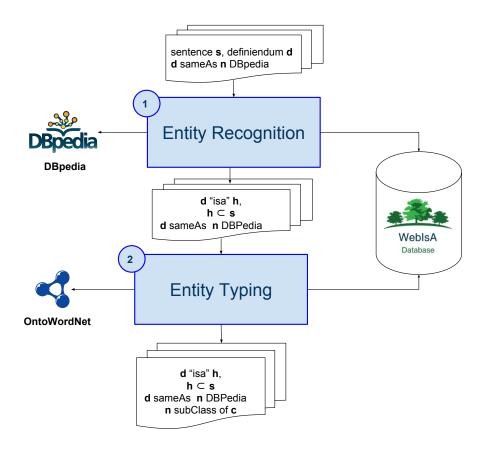


Fig. 1. Overview of our method for hypernym extraction and class induction.

a set of hypernym relations. Next, to induce taxonomic relations between intermediate concepts, the Web is searched again with these patterns. The last step consists of pruning of the resulting graph.

Velardi et al. [24] proposed a graph-based algorithm to learn a taxonomy from textual definitions, extracted from a corpus and the Web in an iterative fashion. The hypernym extraction is based on a set of Word-Class Lattices (WCLs) [12] classifiers trained on a domain-independent set of definitions from Wikipedia. An optimal branching algorithm is used to induce a taxonomy.

In this work we combine some of the above mentioned state-of-the-art NLP techniques of taxonomy induction - i.e. we use a collection of around 50 Hearst-like "isa" patterns collected and evaluated in [18] and similar scoring techniques to [24] - and also involve a large dataset namely the WebIsaDB [18], which consists of around 40 Million matches between these patterns and a corpus built from the Common Crawl [11].<sup>2</sup> Finally, in our lexical-based approach, we also

<sup>&</sup>lt;sup>2</sup> the largest publicly available crawl of the Web.

#### Algorithm 1 Entity Recognition

Require: - (s, d, n): s the definitional sentence for d, n the corresponding DBpedia node for d and  $h \subset s$  the hypernym of d; OWN: the OntoWordnet mapping Ensure: - (s, d, n, h)1: H = extractIsaTuple(s, d)2: s' = s + additionalContextWords(n)3: for  $t_i = \{d, h, Context, f\} \in H$  do

4:  $rank(t_i) = \frac{|getBoW(s') \cup getBoW(t_i.Context)|}{|getBoW(t_i.Context)|}$ 5:  $rank(t_i) = rank(t_i) * stringCoverage(t_i.h, s)$ 6:  $rank(t_i) = rank(t_i) * t_i.f$ 7:  $top = argmax_{t_i} rank(t_i)$ 8: return(s, d, n, top.h)

involve WordNet [4] and the OntoWordnet resource [5] to target the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [10].

# 3 Approach

In Figure 1 we represent the workflow of our system and how it interfaces with the specifications of the second task of the 2016 OKE Challenge, namely "Class Induction and entity typing for Vocabulary and Knowledge Base enrichment". The task input consists of a fragment of text s which is meant to be a definitional sentence for a given term d (in the rest of this paper we will refer to d as the definiendum) and the corresponding DBpedia node n for d. The expected output is a triple (s, h, c) where h is the hypernym of d, h is a substring of s and c is the most suitable class of d from the DOLCE target reference ontology.

Our methodology can be divided in two subsequent steps:

- (1) **Entity Recognition**: where we extract with Hearst like patterns from the sentence s the most suitable hypernym h in s for the definiendum d. In this phase, if no matches can be found in s then candidates hypernyms are obtained by consulting a large database of "isa" relations extracted from the Web, namely the WebIsaDB [18] (see Section 2 for more details about the involved patterns and WebIsaDB);
- (2) **Entity Typing:** where we collect different chains of "isa" relations rooted on the *definiendum* d and the extracted hypernym h (i.e. the pair (d,h)) and aimed at ending with a pair (i,c) in the OntoWordnet mapping and where i belongs to WordNet lexicon and c is a label for a DOLCE class.

In the following paragraphs we describe these two steps in more details.

#### 3.1 Entity Recognition

With reference to Algorithm 1, initially we are given as input a triple (s,d,n) where s is a definitional sentence for the  $definition dum\ d$  and n is the DBpedia

node for d. To extract the most suitable hypernym h for d which occurs in the sentence s we proceed as follows:

- we create the set H of tuples representing the "isa" relation extracted from s and from the Web for the definiendum d (line 1). Each tuple in H includes also the number f of matches and the concatenation Context of a sample from all the matching sentences. The set H is created by i) first searching for pattern matches between the sentence s and the set of Hearst-like patterns defined in  $[18]^3$  and then, ii) in case of no matches in s, by collecting all the "isa" relations for the  $definiendum\ d$  from the WebIsaDB<sup>4</sup> which already holds all the matches between above Hearst-like patterns and a corpus made on top of the largest publicly available crawl of the Web [11];
- we create s' as the expanded textual context of s (line 2). To extend the extraction context s for d we create s' as the string concatenation of s with the following textual informations:
  - the string label associated to the DBpedia node n;
  - the string label associated to all the DBpedia ontology node  $n_h$ , where  $n_h$  is a direct superclass of n in the DBpedia ontology;
  - the glosses associated to all the above DBpedia ontology node  $n_h$ .
- we rank all the tuple  $t_i$  of H (lines 3-6) with a score that combines the following metrics:
  - BoW overlap: the normalized number of common content words<sup>5</sup> between the Bag of Words of s' and the extraction context of  $t_i.h$  (line 4);
  - Hypernym String Coverage: is the length of the string  $t_i.h$  if  $t_i.h \subset s$ , 0 otherwise (line 5);
  - Tuple Frequency: the number of pattern matches.
- we select the tuple *top* with the highest score in the rank (line 7);
- we return the quadruple (s, d, n, top.h) with the extracted hypernym  $top.h \subset s$  for the definiendum d (line 8);

To better understand our approach to hypernym extraction we now turn to a step-by-step example. Given the input triple (s,d,n) where: s= "A perennial plant or simply perennial (from Latin per, meaning through, and annus, meaning year) is a plant that lives for more than two years. ...";  $^6d =$  "perennial plant";  $^7n = \frac{1}{2}$  http://dbpedia.org/resource/Perennial\_plant.

Since no matches were produced by applying our set of Hearst-like patterns directly on the sentence s, we collected in H a total of 152 different "isa" relations

<sup>&</sup>lt;sup>3</sup> In this case f = 1 and Context = s

<sup>&</sup>lt;sup>4</sup> In this case f = |matchingsentences| and Context is the result of concatenation of a random sample of at most five matching sentences.

<sup>&</sup>lt;sup>5</sup> A standard set of english stop words is used to select only content words.

<sup>&</sup>lt;sup>6</sup> Corresponding to sentence http://www.ontologydesignpatterns.org/data/oke-challenge/task-2/sentence-117#char=0,381

<sup>&</sup>lt;sup>7</sup> Corresponding to definiendum http://www.ontologydesignpatterns.org/data/oke-challenge/task-2/sentence-117#char=2,17

**Table 1.** Example of tuples extracted for the definiendum "perennial plant"

```
t_1.d =
              "perennial plant"
t_1.h =
               "shrub
t_1.Context = "Forests where the dominant woody elements are shrubs i.e. perennial plants, gen-
              erally more than 0.5 m and less than 5 m high on maturity and without a definite
             crown. Changes in forest cover classes as interpreted from forest cover mapping. are
              shrubs i.e. woody perennial plants, generally of more than 0.5 m and less than 5 m in
              Shrub It refers to vegetation types where the dominant woody elements are shrubs
              i.e. woody perennial plants, generally more than 0.5 meters and less than 5 meters in
              height at maturity and without a definite crown.."
t_1.f =
              "perennial plant'
t_2.d =
t_2.h =
              "plant"
t_2.Context =
              "A perennial plant is a plant that lives longer than 2 years. A perennial plant or
              simply perennial is a plant that lives for more than two years. A perennial plant is
              a plant that lives more than 2 years, or a winter hardy plant. A perennial plant is a
              plant that lives for more than two years or produces in successive years. Such raw-
              materials can be derived from plants, such as annular or perennial plants, including
              straw, willow, energy hay, Miscantho s..'
t_2.f =
              "perennial plant
t_3.d =
t_3.h =
              "shrub about 13 tall"
t_3.Context = "perennial plant is a small much-branched shrub about 13' tall."
t_3.f =
```

for the  $definiendum\ d$ . In Table 1 we show an excerpt of three tuples from the 152 extracted for the definiendum "perennial plant".

To score all the extracted tuples we first create s' as an extension of s including some additional textual information we derive from n. In our example s' ="A perennial plant or simply perennial (from Latin per, meaning through, and annus, meaning year) is a plant that lives for more than two years. The term is often used to differentiate a plant from shorter-lived annuals and biennials. The term is also widely used to distinguish plants with little or no woody growth from trees and shrubs, which are also technically perennials." We then assign to each extracted tuple  $t_i$  a score which combines: i) the normalized number of common words between the bag of words of s' and of  $t_i$ ; the string support of  $t_i.h$  in s, and iii) the tuple frequency  $t_i.f$ . For the tuples enumerated in Table 1 we obtain:  $rank(t_1) = 10.71$ ,  $rank(t_2) = 65.62$  and  $rank(t_3) = 0.0$ . We finally identify the top ranking tuple  $t_2$  and output (s, d, n, "plant").

#### 3.2 Entity Typing

With reference to Algorithm 2, in this phase of the task we receive as input: the quadruple (s,d,n,h) where s is the definitional sentence for the definiendum d, n is the corresponding DBpedia node for d and h is the most suitable hypernym for d in s; and output the same tuple with an additional component c which is the most suitable DOLCE class for d in the context of s. To identify the most suitable DOLCE class c for d in the context of the sentence s we proceed as follows:

#### Algorithm 2 Entity Typing

**Require:** - (s, d, n, h): s the definitional sentence for d, n the corresponding DBpedia node for d and  $h \subset s$  the hypernym of d; OWN: the OntoWordnet mapping.

**Ensure:** - (s, d, n, h, c): s the definitional sentence for d, n the corresponding DBpedia node for d,  $h \subset s$  the hypernym of d and c the most suitable DOLCE class for d

```
1: I = OWN.qetAllInstances()
2: for wn = (synonyms, DolceClassAncestors) \in I, h right substring of at least one
    ws \in wn.synonyms do
 3:
      for dc \in wnDolceClassAncestors do
 4:
         rank(dc) = rank(dc) + 1
 5: if rank.isEmpty() then
      H' = extractWebIsaTuples(h)
 7:
      s' = s + additionalContextWords(n)
      for t_i = \{h, h', Context, f\} \in H' do
8:
9:
         for wn = \{synonyms, DolceClassAncestors\} \in I, h' right substring includes
         at least one word ws \in wn.synonyms do
            for dc \in wnDolceClassAncestors do
10:
              rank(dc) = \frac{|getBoW(s') \cup getBoW(t_i.Context)|}{|getBoW(t_i.Context)|}
11:
              rank(dc) = rank(dc) * stringCoverage(h,s)
12:
13:
              rank(dc) = rank(t_i) * t_i.f
14: top = \operatorname{argmax}_{dc} rank(dc)
15: return (s, d, n, h, top)
```

- we first create the set I of all the linked instances from OntoWordnet (line 1); Each element of I contains an "isa" pair (synonyms, DolceClassAncestors), where synonyms is a set of word expression for the same WordNet concept and DolceClassAncestors is a set of DOLCE classes mapping to the ancestors of w in WordNet;
- we search all the pairs  $wn \in I$  where at least one word  $\in wn.synonyms$  is right string included by the hypernym h, for all these pair wn we count all the corresponding classes  $dc \in wn.DolceClassAncestors$  (lines 2 4). The right string inclusion is an effective heuristic also adopted in the taxonomy induction process to project hypernyms into core ontology concepts [24].
- if no DOLCE classes were identified on the previous step (line 5) we proceed by creating the set H' containing all the hypernyms h' in the WebIsaDB for h (line 6);
- we follow the approach from hypernym ranking (Algorithm 1) and assign a score to all the DOLCE classes related to the instances  $wn \in I$  where at least one word  $\in wn.synonyms$  is right string included by the hypernym h'.
- finally, we return the quintuple (s, d, n, h, top) where top is the top-ranked DOLCE class in our rank (lines 14-15);

We now extend our running example from Section 3.1 with the application of our class induction approach. Let (s, d, n, h) be the output tuple from our Hypernym Extraction algorithm, the Class Induction phase then works as follows:

- we create the set I, which is composed by all the pair extracted from OntoWordNet of the kind (synonyms, DolceClassAncestors);
- we analyse all the pairs wn from I where at least one word in wn.synonyms is right-string included in h, for example the pairs:  $w_1$  = "(herb, herbaceous plant,Organism)",  $w_2$  = "(plant, flora, plant life,Organism)" and  $w_3$  = "(plant, works, industrial plant,PhysicalObject)" and create the following rank: rank(Organism) = 2 and rank(PhysicalObject) = 1;
- since a non-empty rank is obtained, the most suitable DOLCE class for the  $definiendum\ d=$  "perennial plant" is the class Organism;

When h does not "project" its string into any WordNet synset, we search for all the hypernyms of h in the WebIsaDB and apply for them the same right-string inclusion heuristic to candidate WordNet synsets and rank the corresponding DOLCE class as expressed in the OntoWordnet ontology.

#### 4 Evaluation

We follow the standard evaluation protocol of the OKE challenge. We developed a Web-service which is consumed by the General Entity Annotator Benchmark (GERBIL)<sup>8</sup> [22]. By calling our Web-service the above system send a series of requests in the NLP Interchange Format (NIF) [7] and our Web-service replies to the request using the same format. An example excerpt of such request follows:

```
@prefix nif: <http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#> .
@prefix d0: <http://ontologydesignpatterns.org/ont/wikipedia/d0.owl#>
@prefix dul: <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#>
@prefix dbpedia: <a href="http://dbpedia.org/resource/">

@prefix gerbil: <a href="http://www.aksw.org/gerbil/NifWebService/">http://www.aksw.org/gerbil/NifWebService/</a>>.
gerbil:request_1#char=0,150
                            nif:RFC5147String , nif:String , nif:Context ;
     a
                            "Brian Banner is a fictional villain from the Marvel Comics
     nif:isString
                            Universe created by Bill Mantlo and Mike Mignola and first
                            appearing in print in late 1985.";
                            "0"^^xsd:int ;
     nif:beginIndex
                            "150"^^xsd:int
     nif:endIndex
gerbil:request_1#char=0,12
                            nif:RFC5147String , nif:String ;
     a
     nif:anchorOf
                            "Brian Banner"@en :
     nif:referenceContext gerbil:request_1#char=0,150 ;
                             "0"^^xsd:int ;
    nif:beginIndex
                            "12"^^xsd:int;
     nif:endIndex
     itsrdf:taIdentRef
                            dbpedia:Brian_Banner .
The expected answer for the above request is:
@prefix oke: <http://www.ontologydesignpatterns.org/data/oke-challenge/task-2/> .
oke:sentence-1#char=0,150
                            nif:RFC5147String , nif:String , nif:Context ;
     nif:isString
                            "Brian Banner is a fictional villain from the Marvel Comics
                            Universe created by Bill Mantlo and Mike Mignola and first
```

<sup>&</sup>lt;sup>8</sup> http://aksw.org/Projects/GERBIL.html

**Table 2.** Performances on Entity Recognition and Entity Typing on the gold-standard sample from the 2015 edition of the challenge.

OKE 2015 Task 2 gold standard sample						
Sub-task	Micro			Macro		
	F1	Р	R	F1	Р	R
Entity Recognition (weak)						
Entity Recognition (strong)	30.3	42.4	23.6	24.9	25.2	24.7
Entity Typing	29.5	29.0	29.9	28.4	28.6	29.5

```
appearing in print in late 1985.";
                            "0"^xsd:int;
"150"^xsd:int.
     nif:beginIndex
     nif:endIndex
oke:sentence-1#char=0.12
                            nif:RFC5147String , nif:String ;
     nif:anchorOf
                            "Brian Banner"@en :
     nif:referenceContext oke:entence-1#char=0,150 ;
nif:beginIndex "0"^^xsd:int ;
     nif:beginIndex
                            "12"^^xsd:int ;
     nif:endIndex
     itsrdf:taIdentRef
                            dbpedia:Brian_Banner .
dbpedia:Brian_Banner
                           oke:FictionalVillain;
     rdfs:label
                           "Brian Banner"@en
oke:FictionalVillain
                            owl:Class ;
     rdfs:label
                            "fictional villain"@en ;
     rdfs:subClassOf
                            dul:Personification .
oke:sentence-1#char=18,35
                            nif:RFC5147String , nif:String ;
     nif:anchorOf
                            "fictional villain"@en ;
     nif:referenceContext oke:entence-1#char=0,150 ;
                            "18"^xsd:int;
"35"^xsd:int;
     nif:beginIndex
     nif:endIndex
     itsrdf:taIdentRef
                            oke:FictionalVillain
```

In Table 2 we show the results on evaluating our approach with the 2015 challenge gold standard sample dataset. For the Entity Recognition step we report two kinds of experiment:

- 1. weak: which evaluates the extracted hypernyms by means of a partial string match:
- 2. *strong*: where, instead, extracted hypernyms are compared against gold standard hypernyms with a perfect string match.

GERBIL computes both micro and macro precisions, recalls and F-measures. Error analysis on the 2015 dataset shows that most of the errors of our system are due to the following issues:

 errors derived from the part-of-speech tagging system [21] we used during the application of lexical-syntactic patterns (see Algorithm 1, line 1), in particular errors come from Verb/Noun mismatches and proper nouns from different languages;

- presence of erroneous "isa" relations extracted by lexical-syntactic patterns, which, in turn, may introduce wrong taxonomic relations;
- complex definitional sentence structures that can not be matched using Hearst-like pattern, and challenging, highly polysemous definiendums like, for instance, "god", which is a concept hard to classify also for human;
- the low recall of Entity Typing may be improved by extending/replacing the OntoWordNet mapping between WordNet and the foundational DOLCE with a more rich sense inventory, i.e. DBpedia.

#### 5 Conclusion

In this paper we presented a system aimed for the task number 2 of the 2016 edition of the Open Knowledge Extraction (OKE) Challenge. In line with the general aims of the challenge and with the Open Knowledge Extraction paradigm, our system:

- performs hypernym extraction by applying Hearst-like patterns on the input definitional contexts and harvesting "isa" relations from a large collection of Hearst-like patterns matches extracted from the Web;
- in order to discover the most suitable class for the extracted hypernyms, it combines WordNet and OntoWordNet to target the DOLCE ontology model.

We also performed an evaluation on the 2015 gold standard sample dataset, and analysed the nature of the errors.

Our contributions indicate that: i) Hearst-like patterns are an optimal solution for the hypernym extraction; ii) but potentially lead to "noisy/erroneous" relations during the class induction phase; iii) richer mappings than OntoWordNet should be involved to improve the coverage of the Entity Typing sub-task. We also consider the Web scale of the proposed approach an initial step to investigate a multilingual settings of the Open Knowledge Extraction paradigm.

Further work will focus on the implementation of more reliable "isa" relation filtering techniques aimed at removing noisy taxonomical relations extracted from the Web. Performance may benefit from a mapping between common senses and foundational ontologies with a higher sense covering.

## Acknowledgement

This work was partially funded by the Junior-professor funding programme of the Ministry of Science, Research and the Arts of the state of Baden-Württemberg, Germany (project "Deep semantic models for high-end NLP applications").

#### References

 Agirre, E., Ansa, O., Hovy, E.H., Martínez, D.: Enriching very large ontologies using the WWW. In: ECAI Workshop on Ontology Learning. pp. 28–33. Berlin, Germany (2000)

- 2. Biemann, C.: Ontology Learning from Text: A Survey of Methods. LDV Forum 20(2), 75–93 (2005)
- 3. Buitelaar, P., Cimiano, P., Frank, A., Hartung, M., Racioppa, S.: Ontology-based information extraction and integration from heterogeneous data sources. Int. J. Hum.-Comput. Stud. 66(11), 759–788 (2008)
- 4. Fellbaum, C.: WordNet: An Electronic Lexical Database. Bradford Books (1998)
- 5. Gangemi, A., Navigli, R., Velardi, P.: On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: Proceedings of the OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, chap. The OntoWordNet Project: Extension and Axiomatization of Conceptual Relations in WordNet, pp. 820–838. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
- Hearst, M.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th conference on Computational linguistics. pp. 539–545. Association for Computational Linguistics (1992)
- Hellmann, S., Lehmann, J., Auer, S., Brümmer, M.: The Semantic Web ISWC 2013: 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II, chap. Integrating NLP Using Linked Data, pp. 98–113. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
- 8. Kozareva, Z., Hovy, E.: Learning arguments and supertypes of semantic relations using recursive patterns. In: Proc. of ACL-10. pp. 1482–1491. Uppsala, Sweden (2010)
- 9. Maedche, A., Staab, S.: Ontology learning. In: Handbook on Ontologies, pp. 245–268. Springer (2009)
- Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: WonderWeb deliverable D18 ontology library (final). Tech. rep., IST Project 2001-33052 Wonder-Web: Ontology Infrastructure for the Semantic Web (2003)
- 11. Meusel, R., Petrovski, P., Bizer, C.: The webdatacommons microdata, rdfa and microformat dataset series. In: The semantic Web ISWC 2014, pp. 277–292. Springer (2014)
- Navigli, R., Velardi, P.: Learning word-class lattices for definition and hypernym extraction. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. pp. 1318–1327. Association for Computational Linguistics, Uppsala, Sweden (2010)
- Nuzzolese, A.G., Gentile, A.L., Presutti, V., Gangemi, A., Garigliotti, D., Navigli, R.: Semantic Web Evaluation Challenges: Second SemWebEval Challenge at ESWC 2015, Portorož, Slovenia, May 31 June 4, 2015, Revised Selected Papers, chap. Open Knowledge Extraction Challenge, pp. 3–15. Springer International Publishing (2015)
- 14. Oakes, M.P.: Using Hearst's rules for the automatic acquisition of hyponyms for mining a pharmaceutical corpus. In: RANLP Text Mining Workshop '05. pp. 63–67. Borovets, Bulgaria (2005)
- 15. Perez, G.A., Mancho, M.D.: A Survey of Ontology Learning Methods and Techniques. OntoWeb Deliverable 1.5 (2003)
- Petasis, G., Karkaletsis, V., Paliouras, G., Krithara, A., Zavitsanos, E.: Ontology population and enrichment: State of the art. In: Paliouras, G., Spyropoulos, C., Tsatsaronis, G. (eds.) Knowledge-Driven Multimedia Information Extraction and Ontology Evolution, Lecture Notes in Computer Science, vol. 6050, pp. 134–166. Springer Berlin / Heidelberg (2011)

- 17. Ritter, A., Soderland, S., Etzioni, O.: What is this, anyway: Automatic hypernym discovery. In: Proceedings of the 2009 AAAI Spring Symposium on Learning by Reading and Learning to Read. pp. 88–93. Palo Alto, California (2009)
- 18. Seitner, J., Bizer, C., Eckert, K., Faralli, S., Meusel, R., Paulheim, H., Ponzetto, S.: A large database of hypernymy relations extracted from the web. In: Proceedings of 10th edition of the Language Resources and Evaluation Conference. Portorož, Slovenia (2016)
- 19. Snow, R., Jurafsky, D., Ng, A.: Semantic taxonomy induction from heterogeneous evidence. In: Proc. of ACL-COLING-06. pp. 801–808. Sydney, Australia (2006)
- Snow, R., Jurafsky, D., Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Proc. of NIPS-04. pp. 1297–1304. MIT Press, Cambridge, Mass. (2004)
- Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich Part-of-speech Tagging with a Cyclic Dependency Network. In: Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '03). pp. 173–180. ACL (2003)
- 22. Usbeck, R., Röder, M., Ngonga Ngomo, A.C., Baron, C., Both, A., Brümmer, M., Ceccarelli, D., Cornolti, M., Cherix, D., Eickmann, B., Ferragina, P., Lemke, C., Moro, A., Navigli, R., Piccinno, F., Rizzo, G., Sack, H., Speck, R., Troncy, R., Waitelonis, J., Wesemann, L.: GERBIL general entity annotation benchmark framework. In: 24th WWW conference (2015)
- Velardi, P., D'Antonio, F., Cucchiarelli, A.: Open domain knowledge extraction: Inference on a web scale. In: Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics. pp. 35:1–35:8. WIMS '13, ACM, New York, NY, USA (2013)
- 24. Velardi, P., Faralli, S., Navigli, R.: Ontolearn reloaded: A graph-based algorithm for taxonomy induction. Computational Linguistics 39(3), 665–707 (2013)
- 25. Yang, H., Callan, J.: A metric-based framework for automatic taxonomy induction. In: Proc. of ACL-09. pp. 271–279. Suntec, Singapore (2009)