

Type inference in Dbpedia from free text

Ondřej Zamazal
Tomáš Kliegr
Václav Zeman

Motivation

- ▶ New DBpedia resource types extraction
- ▶ Type/hypernym discovering from resources without infobox or mapping rules

```
dbpedia:Dublin_City_Cup    rdf:type    dbpedia-owl:Event    #new type
```

- ▶ Find more specific DBpedia types

```
dbpedia:Angela_Merkel    rdf:type    dbpedia-owl:Person    #mapping-based type  
dbpedia:Angela_Merkel    rdf:type    dbpedia-owl:Chancellor    #more specific discovered type
```

- ▶ Extract types/hypernyms from plain text
- ▶ Provide datasets with newly discovered DBpedia types and evaluation
- ▶ Provide a hypernym discovery tool for local DBpedia chapters to acquire new types

Outline

- ▶ LHD 1.0: identifies the hypernym word from the first sentence of the short abstract and maps it to a DBpedia resource
 - Language dependent process
 - POS Tagger and user-defined grammar are required
- ▶ LHD 2.0 – STI classifier: inferring DBpedia ontology types based on co-occurrence of LHD 1.0 and DBpedia types
- ▶ hSVM classifier: inferring types based on machine learning over bag of words
 - Language independent approach
- ▶ Evaluation on a large crowdsourced dataset

STI classifier

- ▶ Extract a hypernym word from the first sentence
 - TreeTagger + Gate

Havel was a Czech playwright, essayist, poet, dissident and politician.

- ▶ Map the hypernym word to a DBpedia resource
 - dbp:Playwright
- ▶ Get their DBpedia types along with frequency
- ▶ Remove the supertypes, balancing specificity and reliability
- ▶ Select the type with highest support
 - dbpedia-owl:Writer

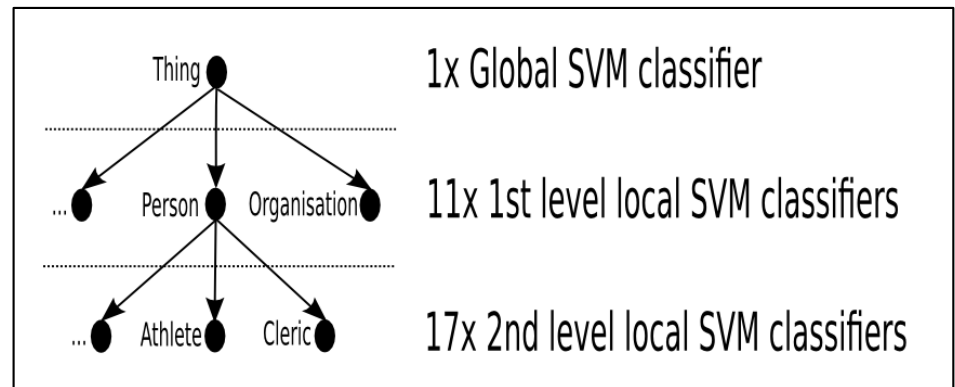
Comedian:1, ... OfficeHolder:7, ..., Writer:266, Artist:277, Person:521

Comedian:1, ... OfficeHolder:7, ..., Writer:266, Artist:277, ~~Person:521~~

Language dependent approach !!!

hSVM classifier

- ▶ Each resource is represented with two **bag-of-words** models
 - BOW1 based on **short abstract**
 - BOW2 based on article **categories**
- ▶ Classification models are trained by SVM with mapping-based resources
- ▶ 3 layers of SVM classifiers = 58 SVM classifiers



hSVM classifier

- ▶ Each SVM model returns an assignment confidence
- ▶ Supertype confidence is propagated to its subtypes as well
- ▶ We define 3 strategies to determine the final type
 - α strategy assigns with maximum confidence from all specific types (leaf types)
 - β strategy assigns type with maximum confidence from all types
 - γ strategy combines α and β strategies

Language independent approach !!!

Evaluation

- ▶ Evaluation on a crowdsourced dataset
- ▶ Each entity was typically annotated by three to four annotators
- ▶ 1021 entities with a DBpedia type in the gold standard
- ▶ Evaluation of
 1. STI – language dependent approach
 2. hSVM – language independent approach
 3. STI + hSVM fusion
 4. Mapping-based types dataset
 5. SDType – Mapping-based types (Heuristic) dataset
 - heuristic link-based type inference mechanism

Results

Classifier	Acc _{Exact}	Acc _{Supertypes}	Acc _{[Sub Super]types}
STI	.375	.456	.643
hSVM	.350	.725	.768
STI+hSVM	.468	.831	.856
SDType	.337	.706	.775
DBpedia (mapping-based)	.374	.856	.920

Conclusion

- ▶ STI/hSVM vs SDType
 - Type intersection is not so large
 - STI/hSVM is more specific
 - Use both STI/hSVM and SDType to discover new DBpedia types
- ▶ STI – language dependent (now only for English, German and Dutch)
- ▶ hSVM – language independent (available for all DBpedia chapters)
- ▶ The newest tool and datasets will be available very soon; publication is under review
- ▶ Watch our website
 - <http://ner.vse.cz/datasets/linkedhypernyms/>

Thank you for your attention

<http://ner.vse.cz/datasets/linkedhypernyms/>

vaclav.zeman@vse.cz