Web-based Ontology Learning with ISOLDE

Nicolas Weber, Paul Buitelaar

DFKI GmbH - Language Technology Lab
Saarbrücken, Germany

{paulb,nweber}@dfki.de

Abstract The last few years saw a continuing increase in the availability of open source web-based information resources such as Wikipedia and similar initiatives. As this kind of information is freely available, has some pre-defined format, covers many topic areas and is of an acceptable quality it is an ideal resource for bootstrapping or extending domain ontologies. In this paper we show how web resources such as Wikipedia, Wiktionary and a German online dictionary (DWDS) can be used in combination with a domain corpus and a general purpose named-entity tagger to derive a domain ontology. The ISOLDE (Information System for Ontology Learning and Domain Exploration) system we describe generates a domain ontology by extracting class candidates from the linguistic context of a given set of ontology instances and by deriving further knowledge on these class candidates from available web resources.

1 Introduction

In this paper we show how web resources such as Wikipedia and Wiktionary can be used in combination with a domain corpus, a general purpose named-entity tagger and a seed or 'base' ontology to derive a domain ontology. The ISOLDE (Information System for Ontology Learning and Domain Exploration) system we describe generates a domain ontology by extracting class candidates from the linguistic context of a given set of ontology instances and by deriving further knowledge on these class candidates from available web resources.

The ISOLDE approach is based on techniques for unsupervised named-entity recognition as developed among others by (Yangarber et al. 2000, Cimiano and Staab 2004, Cimiano et al. 2005, Etzioni et al. 2005) but the results are used in a different way. Whereas for these approaches the assignment of named-entities with extracted classes is the final goal, ISOLDE goes one step further by trying to find additional information on the extracted classes in order to organize them into a taxonomy or full ontology.

2 ISOLDE System Overview

The ISOLDE system can be defined by the following three analysis steps that can be defined as follows (see also Figure 2.):

1. Named-entity recognition (NER) for the extraction of instances for classes in the base ontology
2. Linguistic pattern analysis for the extraction of class candidates from the context of the instances extracted in step 1
3. Collecting web-based knowledge for the class candidates extracted in step 2 and integrating this into a new or extended taxonomy/ontology

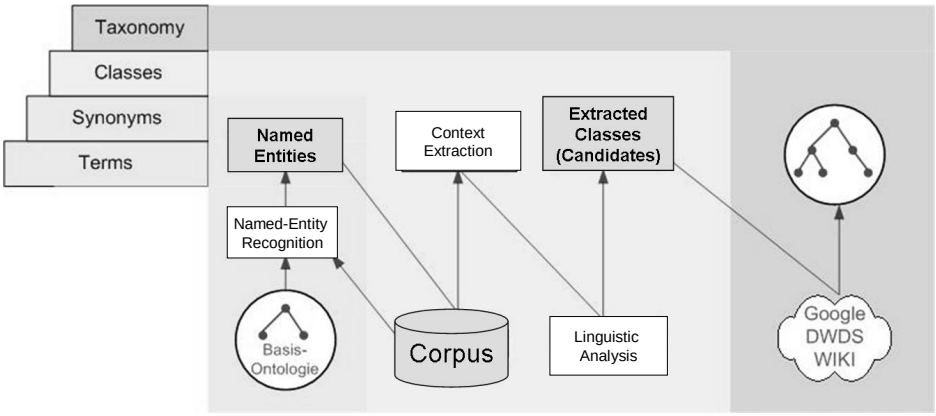


Figure 1: ISOLDE system overview

In step 1 we use a domain-specific corpus, a base ontology and a general purpose NER system (SproUT, see Drozdzynski et al. 2004) to find instances for the classes in the base ontology.

In step 2 we collect the linguistic contexts of the instances derived in step 1 and extract class candidates from this by use of lexico-syntactic patterns (Hearst 1992):

NE „*ist ein*“ {NP}	*Jürgen Klinsmann ist ein Trainer*
NE „ , “ {NP}	*Jürgen Klinsmann, Trainer des...*
{NP} NE	*Trainer Jürgen Klinsmann*

If one of these patterns matches, the head of the NP is taken as a new class candidate. For instance, we can extract the class candidate TRAINER from the context of *Jürgen Klinsmann* – which was instantiated for the class PERSON in the base ontology – in the following sentence:

Jürgen Klinsmann, Trainer der Nationalmannschaft

(Jürgen Klinsmann, trainer of the national team)

The result of step 2 is a list with extracted class candidates for each named-entity. For every class candidate we determine its statistical relevance by use of X^2, which provides a measure over frequency in the domain specific corpus with that in a balanced corpus (Manning and Schütze 1999).

In step 3 we collect information on and between extracted class candidates from online resources: DWDS, Wikipedia and Wiktionary. *Das Digitale Wörterbuch der Deutschen Sprache – DWDS* (the digital dictionary of German) is a public online dictionary. It consists of a corpus that currently comprises 80.000 texts and a corresponding dictionary. Wikipedia is a free multilingual encyclopaedia that anyone can edit. As of today there are 345.000 articles in German in various domains. The articles consist of free (i.e. unstructured) text and semi-structured data. Wiktionary, a multilingual dictionary and thesaurus, is a sister project from Wikipedia. The German Wiktionary is still a small project (15.000 articles) but it is the fastest growing Wiki-project.

Whereas Wikipedia provides general knowledge on various topics, DWDS and Wiktionary additionally provide linguistic knowledge on lexical semantic relations (synonymy, hyponymy), morphology and part of speech.

In the context of ISOLDE, we aim at deriving the following information from these resources:

- taxonomic: establishing if two class candidates are in a hierarchical relation (compare the RDF/OWL property *SubClassOf*)
- non-taxonomic: establishing if two class candidates are equivalent (compare the OWL property *SameAs*)

A problem occurs if extracted relations are in conflict to each other, as in *SubClassOf(Defender, Goalkeeper)* and *SubClassOf(Goalkeeper,Defender).* To avoid this, the relations are ranked. If a relation is more frequent than another, it is more likely that this relation is correct. Are two conflicting relations equally frequent, then the two classes could be equivalent as in *SameAs(Goalkeeper, Defender).*

3 Experiment

In order to test the ISOLDE system we defined an experiment in generating a domain ontology for football (soccer) by use of a domain-specific corpus and the web resources discussed above. The corpus consisted of around 3000 match reports and other news articles on soccer that were downloaded from the web[1].

In step 1, all named-entities of class PERSON were collected. As presented in Figure 2., there is a correlation between the number of extracted (different) instances of class PERSON and the number of documents these occur in. The higher the threshold for the occurrence of the PERSON instance, the less different PERSON instances are processed. We therefore decided on taking into account only those PERSON instances that occurred at least 40 times.

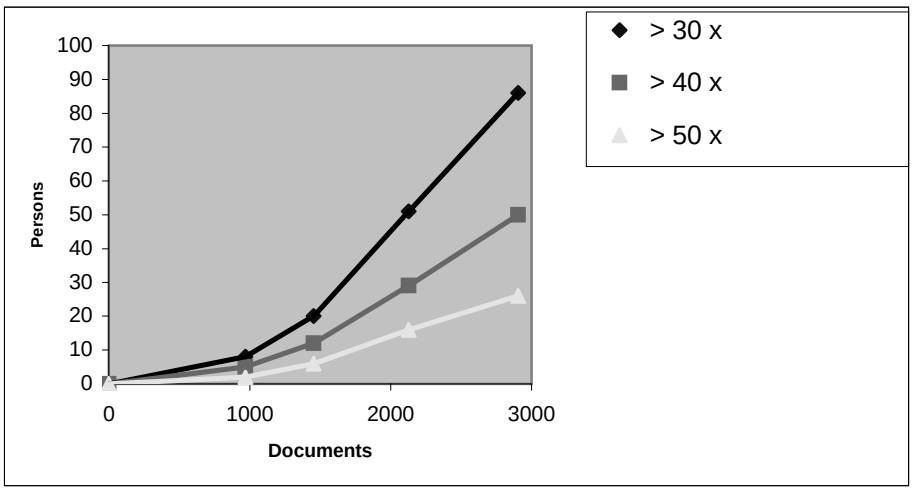


Figure 2: Correlation between the number of different instances of class PERSON and the number of documents they occur in

In step 2 the linguistic context for each of the extracted named-entities is analyzed, e.g.:

Named-Entity:	*Michael Ballack*
Linguistic Context:	*Münchens Mittelfeldspieler Michael Ballack ist nach einer Entscheidung …*
Pattern:	*{NP} Ballack*
Extracted Class Candidate:	*Mittelfeldspieler*

[1] http://fifaworldcup.yahoo.com/06/en/

Named-Entity:	*Michael Ballack*
Linguistic Context:	*Michael Ballack, bester Mann auf dem Platz..*
Pattern:	*Ballack "," {NP}*
Extracted Class Candidate:	*Mann*

As discussed above, a frequency and statistical relevance measure is computed for each of the extracted class candidates. For instance, *Stürmer (striker)* occurs 334 times in our corpus (~0.5 mio tokens) and 316 times in the general corpus (~9 mio tokens), which may be represented as follows:

0	**1**	**2**
1	334	316
2	500000	9000000

Using the formula for $\boldsymbol{X^2}$ defined as:

$$\frac{N\,(O_{11}\,O_{22} - O_{12}\,O_{21})^2}{(O_{11} + O_{12})(O_{11} + O_{21})(O_{12} + O_{22})(O_{21} + O_{22})}$$

In this way, we are able to determine which class candidates are relevant to the domain. For instance, the class candidates for *Thierry Henry* are *Stürmer (striker), Trainer (trainer), Vater (father)* of which only the first two are over a certain threshold that determines their domain relevance – see the table below. In addition we want to decide which ones of these classes are really of importance relative to the extracted named-entity (NE). For this purpose we keep track of their co-occurrence, which in this case lets us decide to select *Stürmer* and not *Trainer* as a relevant class candidate.

Class Candidate	**X^2**	**Co-occurrence with NE**
Stürmer (striker)	*2771.27*	*4*
Trainer (trainer)	*19.78*	*1*
Vater (father)	*0.8*	*4*

In step 3 we extract information from Wikipedia, Wiktionary and DWDS for all of the relevant class candidates. For instance, the following information for *Torwart (goalkeeper)*:

Wikipedia

Der **Torwart** (Torhüter, Tormann, Keeper; Schweiz. Goalie) **ist ein Mitspieler** einer Mannschaftssportart. Er ist der defensivste Spieler seiner Mannschaft und seine Hauptaufgabe besteht darin zu verhindern, dass das Spielgerät (z.B. ein Ball) ins Tor der eigenen Mannschaft gelangt. Daher wird er auch Torhüter genannt….

Wiktionary

Bedeutungen:

Derjenige Fußballspieler, dessen Aufgabe es ist, gegnerische Tore zu vermeiden und der hierfür als einziger Spieler auch seine Hände einsetzen darf.

Herkunft:

aus Tor und Wart

Synonyme:

Tormann, Torhüter, Keeper

Oberbegriffe:

Sport, Fußball, Fußballspieler

Unterbegriffe:

Fliegenfänger (neg.)

DWDS

Torwart, der **1.** Ballspiele *Spieler im Tor, der den Ball fängt, abwehrt*: der T. warf den Ball zum Verteidiger **2.** hist. *Wachmann am Tor*;

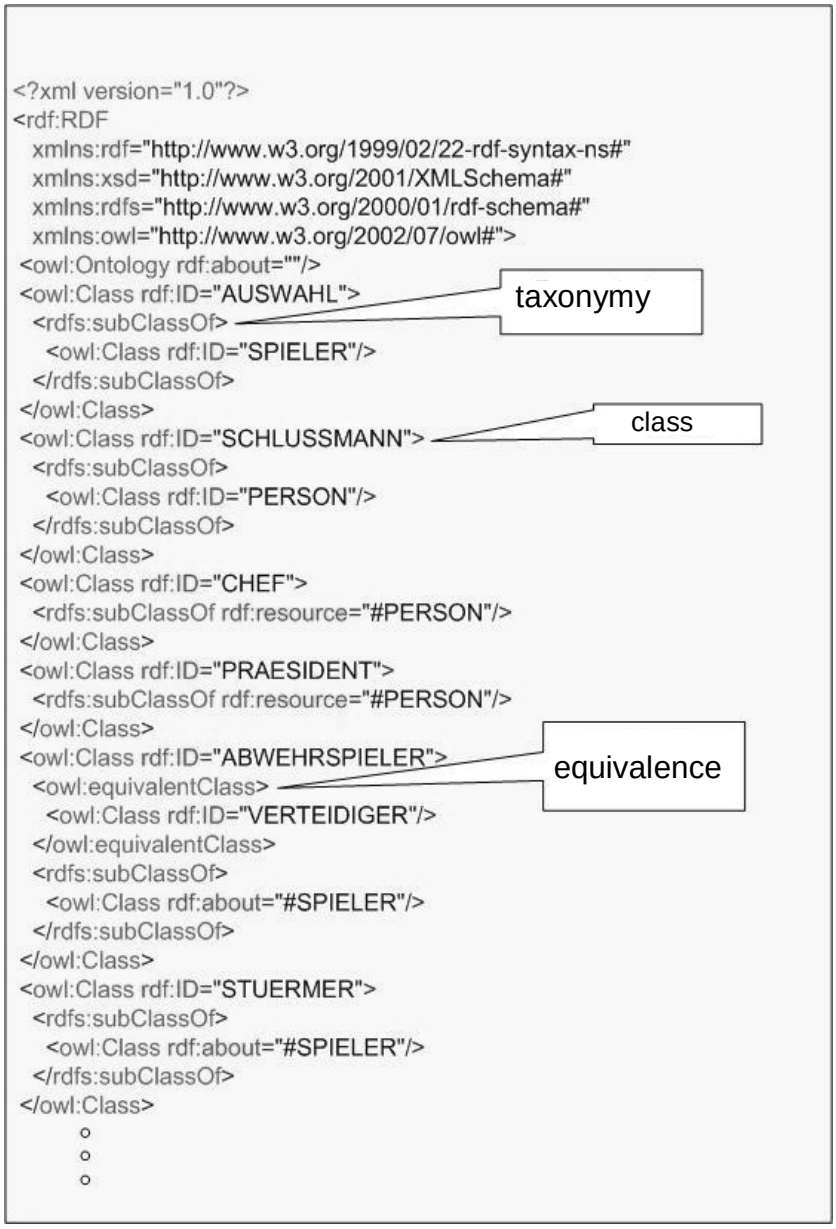
Hyperonyme

Spieler

Hyponyme

Abwehr Abwehrspieler

For each of the class candidates, information can be extracted and gathered in this way and merged into an ontology. The target format for the ontology is OWL as we aim to represent knowledge on equivalence with the OWL property SameAs. The soccer ontology we derived in the experiment looks as follows:




```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#">
 <owl:Ontology rdf:about=""/>
 <owl:Class rdf:ID="AUSWAHL">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="SPIELER"/>
  </rdfs:subClassOf>
 </owl:Class>
 <owl:Class rdf:ID="SCHLUSSMANN">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="PERSON"/>
  </rdfs:subClassOf>
 </owl:Class>
 <owl:Class rdf:ID="CHEF">
  <rdfs:subClassOf rdf:resource="#PERSON"/>
 </owl:Class>
 <owl:Class rdf:ID="PRAESIDENT">
  <rdfs:subClassOf rdf:resource="#PERSON"/>
 </owl:Class>
 <owl:Class rdf:ID="ABWEHRSPIELER">
  <owl:equivalentClass>
    <owl:Class rdf:ID="VERTEIDIGER"/>
  </owl:equivalentClass>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#SPIELER"/>
  </rdfs:subClassOf>
 </owl:Class>
 <owl:Class rdf:ID="STUERMER">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#SPIELER"/>
  </rdfs:subClassOf>
 </owl:Class>
        o
        o
        o
```

Figure 3: Soccer ontology derived from FIFA corpus and web resources

4 Evaluation

To evaluate the generated ontology, we compare it to a gold standard or reference ontology. For this purpose we used the manually created SportEvent ontology which was developed in the SmartWeb project[2]. For the evaluation we used however only a relevant sub-tree with root *SportiveRole*, which contains 50 classes and 49 direct (taxonomic) relations and 226 indirect relations.

A manually generated ontology has always a deeper (hierarchy-) structure than an automatically generated ontology. One reason for this is the fact that for structuring purposes additional classes are defined. Classes like *PositionalMatchFootballPlayer* or *SituationFootballPlayer* are not used in common language but serve only for structuring of the ontology. For this reason the automatic generated ontology is evaluated against the complete SportEvent ontology as well as to a reduced version of this ontology without the classes used for structuring only: SportEvent (adjusted).

The experiment described in the previous section presented us with 45 classes and 37 relations between these classes which were extracted from the domain corpus and the discussed web resources as follows:

Relations	**all**	**taxonomy**	**equivalence**
	37	*31*	*6*
Wikipedia	2	2	0
Wiktionary	12	8	4
DWDS	19	17	2
GOOGLE	4	4	0

As presented in this table, 85% of relations are obtained from DWDS and Wiktionary and only 15% from Wikipedia and Google. The reason for this is the different type of structure of these documents. DWDS and Wiktionary are semi-structured, i.e. the extraction of the pre-processed relations occurs by the position in text. Wikipedia and Google in contrast only provide unstructured text.

Precision and recall results are shown in the table below. As may be expected, results on the acquisition of ontology classes are much better than on relations, as there are many more possible combinations on the relational level than on the class level, i.e. the system will be able to ‘get it wrong’ more often.

[2] http://www.smartweb-projekt.de

	total	true positives	RECALL	PRECISION
Classes				
SportEvent	50	23	46,0%	31,9%
SportEvent (adjusted)	43	23	53.4%	35,3%
Relations				
SportEvent	226	24	10,6%	10,4%
SportEvent (adjusted)	107	24	22,4%	21,6%

5 Conclusions

We presented ISOLDE, a system for web based ontology learning that uses web resources such as Wikipedia and Wiktionary in combination with a domain corpus, a general purpose named-entity tagger and a seed or 'base' ontology to derive a domain ontology. As recent years saw a continuing increase in the availability of open source web-based information resources such as Wikipedia and similar initiatives, the use of such information seems worthwhile. In this paper we presented some methods for using these resources in bootstrapping or extending domain ontologies. The experiment shows that the best results may be obtained from semi-structured data resources (e.g. web dictionaries).

Acknowledgements

This research has been supported in part by the SmartWeb project, which is funded by the German Ministry of Education and Research under grant 01 IMD01.

References

P. Cimiano, S. Staab. 2004. *Learning by Googling*. In: SIGKDD Explorations Vol. 6, No. 2.

P. Cimiano, G. Ladwig, S. Staab. 2005. *Gimme´ The Context: Context-driven Automatic Semantic Annotation with C-PANKOW*. In A. Ellis, T. Hagino (eds.) Proceedings of the 14th World Wide Web Conference, Japan. ACM Press.

W. Drozdzynski, H-U. Krieger, J. Piskorski, U. Schäfer, and F. Xu. 2004. *Shallow processing with unification and typed feature structures – foundations and applications.* Künstliche Intelligenz, 1:17-23.

O. Etzioni, M. Cafarella, D. Downey, S. Kok, A. Popescu, T. Shaked, S. Soderland, D. Weld, and A. Yates. 2005. *Unsupervised named-entity extraction from the web: An experimental study*. Artificial Intelligence, 165(1):91–134.

M. Hearst. 1992. *Automatic Acquisition of Hyponyms from Large Text Corpora.* In: Proceedings of the 14th International Conference on Computational linguistics, Nantes.

Ch. D. Manning, H. Schütze. 1999. *Foundations of statistical natural language processing.* MIT Press. Cambridge, MA.

R. Yangarber, R. Grishman, P. Tapanainen and S. Huttunen. 2000. *Automatic Acquisition of Domain Knowledge for Information Extraction* In: Proceedings of COLING 2000, 18th International Conference on Computational Linguistics, Saarbrücken.