Using wikipedia and supersense tagging for semi-automatic complex taxonomy construction

Davide Picca
University of Lausanne
Dorigny
Switzerland
davide.picca@unil.ch

Adrian Popescu
CEA LIST
18, route du Panorama
Fontenay aux Roses
France
adrian.popescu@cea.fr

Abstract

In this paper we propose an unsupervised approach for acquiring domain related conceptual hierarchies from open-domain text. Super Sense Tagging (SST) is used to extract up-level terms and Wikipedia categories and WordNet are employed to construct the rest of taxonomic hierarchy. The result is a complete top-bottom taxonomy for every formal context. We describe both the method we implemented and some encoruaging initial experimental results.

Keywords

Ontology, Wikipedia, Taxonomy, Wordnet, Formal concept analysis

1 Introduction

In the Semantic Web paradigm it is required to provide a structured view of the unstructured information expressed in texts. For instance, ontologies, an explicit representation of the knowledge shared by a community, represent a way of organizing domain related knowledge.

When one seeks to define a well-structured taxonomy, it is necessary to specify a set of classes and subclasses in an unambiguous way. In order to be more formal, we use the FCA formalism [12] to explain. In fact, as well formalized in [12] a triple

$$\mathbb{K} := (G, M, I) \tag{1}$$

consisting of two sets G and M and a binary relation $I \in GXM$ is called *formal context*. The elements in G are called objects and those in M attributes and I is the incidence of the context. So a *formal concept* in a *formal context* $\mathbb{K}$ is a duality relationship between a subset of objects denoted by A and a subset of attributes denoted by B where all attributes common to objects in A (*intent*) and all objects common to attributes in B (*extent*) must be the same. This duality relationship is given by: A taxonomy is often arranged in a *hierarchical order* based on a $\leq$ relation between concepts. A concept (A_1, B_1) is a *subconcept* of a concept (A_2, B_2) if $A_1 \subseteq A_2$ or $B_2 \subseteq B_1$. Correspondingly, (A_2, B_2) is a *superconcept* of $(A_1, B_1$, hence $(A_1, B_1) \leq (A_2, B_2)$. Taxonomy maps an ordering from the most general to the most specific concept, top to bottom. The top most concept is called the *supremum* and the bottom most concept is called the *infimum*. First we try to automatically populate the "class" category which defines a group of individuals sharing some high-level properties (*intent*). For example, the class *Person* includes all human entities without specific distinctions. In order to be more precise, "classes" can be organized in a more specific hierarchy using "subClassOf"[18]. So a subclass is a specification of the more general category "class". Hence, we try to define a partial order lattice. For example, the class *Professor* could be stated to be a subclass of the class *Person*. Finally the category "individual" defines instances of classes. For example, *Albert Einstein* is an instance of the subclass *Professor*. In this paper we explore the possibility of combining the use of a semi-structured resource, Wikipedia, and an open-domain supersense tagger for automating the acquisition of a conceptual hierarchy following this structure. Subsequently these tagged categories can be easily exploited for semi-automatically building a domain-oriented ontology (section 8). Our method is based on a combination of two basic approaches: (i) Super Sense Tagging (SST) and (ii) Wikipedia categories exploiting. We adopt SST as a preprocessing step (see Section 4) so as to assign a "supersense" category (e.g. `person`, `act`) to contextual term meanings. SST is the problem to identify terms in texts, assigning a "supersense" category (e.g. `person`, `act`) to their senses in context and apply it to recognize concepts and instances in large scale textual collections." of texts. Then we perform a distributional analysis of the occurrences of such terms

in the corpus, with the goal of finding domain relations among them as defined in [16]. This step permits to extract and tag terms that will be assembled in a specific domain ontology. We choose three different domains, `Music`, `Sport`, `Religion` for experimentation and use Wikipedia categories for distinguishing between subclasses and individuals in each domain (see Section 6). As illustrated in Section 7, the proposed approach achieves good results when compared to similar methods and constitutes an innovative approach to the ontology learning field and taxonomic construction.

2 Related work

An important research effort was directed toward the development of automatic or semi-automatic techniques for annotating classes, subclasses and individuals [15] [8]. Many authors exploit Name Entity Recognition (NER) techniques[15] [14] [1] [10], whereas others use statistical approaches based on the distribution of context in corpus [7] [17]. Usually, these approaches are focused on learning a single relation, i.e. *instanceOf*, without paying attention to the difference between "classes" and its subcategories [8] [3]. These methods are very efficient but they have the disadvantage to be less precise in defining the relational lattice $\mathbb{K}$ compromising the solidity of taxonomy. More precision can be obtained by exploiting Wikipedia categories, that structure the encyclopedic entries in a hierarchy. A number of recent works exploit Wikipedia in NER tasks and we discuss the most relevant of them hereafter. The approach that is the closest to ours is presented in [19]. The authors use Wikipedia entries and analyze the content of articles and WordNet nouns hierarchy to create dictionaries of proper names. WordNet knowledge is employed so as to determine whether a phrase is an instance or a class. Locations, persons and organization names are extracted from a set of 3517 Wikipedia entries. There are 236 person names that are guessed in the dataset and the reported precision is of 61% when only using Wikipedia and of 87% when a candidate matches as person name both in Wikipedia and WordNet. Note that in the latter situation, the recall drops from 77% to 30%. In [2] Wikipedia categories are extracted and used for NER. First, some categories (e.g.: *Cleanup from December*) are removed because they are considered not to be significant outside context and second, the remaining categories are ranked. This approach disregards pages containing disambiguation related categories and the authors acknowledge that, by doing so, they lose an important source of information. In [4], the authors exploit information in the collaborative encyclopedia to disambiguate person names. A machine learning technique (an SVM kernel) is used to create a taxonomy of people by occupation and the reported results are encouraging, with the accuracy around 75%. Although the datasets used in the experiments contain tens of thousands entries, it is underlined in the paper that an important limitation of the method in [4] comes from the utilization of SVMs as it is difficult to further scale-up the method. In [13], the authors propose a method that employs Wikipedia in named entity recognition tasks. The first sentence in a Wikipedia article is extracted and, via the use of machine learning technique, named entities (person names, places and organizations) are discriminated from other concepts. The precision in the person name recognition task is of 90.1%. These good results are partially explained by the fact that the information found on disambiguation pages, which constitute the hardest cases to analyze, is not parsed. The authors also note that, in their vision, Wikipedia categories are not all hypernyms of the analyzed concept (which is a source of noise) and present their utilization as future work.
The common point between our approach and the ones cited in [19], [2], [4], [13] is the use of Wikipedia information for NER. The main difference with [19] arise from the way we aggregate Wikipedia and WordNet. In [2], the ambiguous classes are not considered whereas they are included in the scope of the present work. In [4], a machine learning technique is used to classify entities while we propose a lightweight architecture that addresses the same task. Finally, when compared to [13], our approach differs in that it exploits a different part of the Wikipedia pages (the categories), we use the information on the disambiguation pages and we do not employ any machine learning technique.

3 Our proposal

In this paper we explore the possibility of combining open-domain resources as Wikipedia and open-domain supersense tagger for automating the acquisition of conceptual hierarchy. Subsequently these tagged categories can be easily exploited for building a semi-automatic domain-oriented ontology.

The main contribution of this paper to the problem of taxonomy learning is a novel method that can be exploited for automatically acquiring and populating domain specific ontologies. In particular, our goal is to automatically detect and populate complex taxonomic structures. Our method is based on a combination of two basic approaches: (i) Super Sense Tagging (SST) and (ii) Wikipedia categories retrieval.

First, our system retrieves subconcepts of *supremum* concepts defined by the WordNet supersenses, such as "artifact", "act" and "person". Three different domains (`Music`, `Sport`, `Religion`) are chosen in order to guarantee a formal context and we use Wikipedia categories to discriminate between subclasses and individuals in each domain.

We investigate the hypothesis that supersenses provide a stable and finite set of *supremum* concepts and, when paired with Wikipedia categories, can produce quite precise representations of complex taxonomies up to *infimum* concepts. As illustrated in Section 7, the proposed method achieves good results, offering an innovative approach to the ontology learning field.

4 Supersense Tagging

WordNet [11] defines 41 lexicographer's categories, also called *supersenses* [6], used by lexicographers to provide an initial broad classification for the lexicon

entries[1]. The supersense ontology has several attractive features for NLP purposes. First, concepts, although fairly general, are easily recognizable. Second, the small number of classes of *supremum* concepts and the tendency of similar word senses to be merged together, make this feature really important in order to keep taxonomies as compacted as possible avoiding all possible redundancies in the top level classification. For instance the noun *folk* has four fine-grained senses, at the supersense level it only has two as illustrated below:

1. people in general (noun.group)
2. a social division of (usually preliterate) people (noun.group)
3. people descended from a common ancestor (noun.group)
4. the traditional and typically anonymous music that is an expression of the life of people in a community (noun.communication)

Thanks to this feature, we are able to guarantee a certain coherence among classes and its members. In this example we can assign the first three senses of the noun *folk* to the same class *noun.group* and only the last one to a different class Using the Semcor corpus, a fraction of the Brown corpus annotated with WordNet word senses, a supersense tagger has been implemented [5] which can be used for annotating large collections of English text [2]. The tagger implements a Hidden Markov Model, trained with the perceptron algorithm introduced in [9] and it achieves a recall of 77.71% and a precision of 76.65% . The tagset used by the tagger defines 26 supersense labels for nouns and 15 supersense labels for verbs. The tagger outputs class label information, but also covers other relevant categories and attempts lexical disambiguation at the supersense level. The following is a sample output of the tagger:

(2) $\text{Guns}_{B-noun.group}$ $\text{and}_{I-noun.group}$ $\text{Roses}_{I-noun.group}$ $\text{plays}_{B-verb.communication}$ at_{O} the_{O} $\text{stadium}_{B-noun.location}$

Compared to other semantic tagsets, supersenses have the advantage of being designed to cover all possible open class words. Thus, in principle, there is a supersense category for each word, known or novel. Additionally, no distinction is made between proper and common nouns, whereas the named entity tag set tends to be biased towards the former. In order to tackle this fundamental distinction we developed an algorithm using Wikipedia as resource for accomplishing this task.

5 Wikipedia

Wikipedia is a collaborative effort and its content is partially organized, a property that renders the articles in the encyclopedia interesting in information retrieval tasks. The Wikipedia pages are designated by unique string identifiers and, as underlined in [13], there are different parts in the structure of an article that can be isolated using the syntax of the source files and used for knowledge extraction. One can choose to exploit the headings, the lists, the tables, the internal links or the categories in the articles. Wikipedia categories [3] shape the encyclopedia into a tangled conceptual hierarchy. For example, the entry corresponding the the Italian composer Ferruccio Busoni belongs to the following parent categories: *1866 births*, *1924 deaths*, *People from the Province of Florence*, *20th century classical composers*, *Italian classical pianists*, *Italian composers*, *Italian conductors*, *Italian opera composers*, *Romantic composers*, *Neoclassical composers*, *Jadassohn students*. We observe that the categories include both general information about Busoni (e.g.: *1866 births* or *People from the Province of Florence*) and specific domain information (e.g.: *Italian classical pianists* or *Romantic composers*). As we describe below, in our tentative to separate person names from other classes, we use both types of information but favour the second type against the first. We propose an example meant to give an idea about the depth of the conceptual hierarchy in Wikipedia: the *20th century classical composers* category. One line of parents is: *Classical composers*, *Composers by genre*, *Composers*, *Musicians*, *People by occupation*, *People and self*. For the moment, we only consider the categories on the initial page.

5.1 Exploiting Wikipedia Categories

In this section we describe the way for exploiting categories in a class-instance separation task. When they are created by the contributors, the Wikipedia entries are necessarily assigned to at least one category [4] and thus, belong to the tree of concepts in the encyclopedia. This ensures that categories can be exploited for each concept that is represented in the encyclopedia. The categorical information is already structured and relevant for the annotated pages. We empirically determined that most of pages describing people have attached categories that describe the semantic role of that person and use domain concepts as subclasses that will be searched among the categories presented on a page. For example in the `music` domain, semantic roles as artist, musician, composer, guitarist are employed. When defining relevant subclasses for a domain, we look first at the WordNet nouns hierarchy that defines the concepts in that particular domain. For `music`, concepts under *musician* are retained building domain related subclasses. They are more interesting than general categories because they provide a finer conceptual description. The domain description provided by WordNet is not complete and it is necessary to manually define some other relevant terms. In the case of `religion`, we add *philosopher* because there are philosophers that write about religion. For `music` we also use *journalist* as a descriptor for people that analyze the musical world. Some examples of subclasses used to detect person names are presented in table 1. If one or more of the terms cor-

[1] Throughout the paper we intend WordNet version 2.0.

[2] The tagger is publicly available at: http://sourceforge.net/projects/supersensetag/.

[3] http://en.wikipedia.org/wiki/Wikipedia:Categorical_index

responding to an entry in table 1 are found in the category list extracted from the Wikipedia page of a term, this last is considered as being a *infimum* concept (a person name). There are rare occurrences of pages in Wikipedia that are described only by general terms that allow a classifier to validate them as referring to person names and non-specific subclasses (e.g.: birth, death years are used in these situations).

Domain	Subclasses
Sport	player, coach, champion
Music	artist, musician, composer, guitarist
Religion	saint, prophet, philosopher, bishop
General	birth, death

Table 1: *Some examples patterns used to detect person names.*

6 Methods and algorithms

In our experiments we used the British National Corpus. Each text is splitted into sub-portions of 40 sentences, and each portion is regarded as a different document, collecting overall about 130,000 documents. Each document was annotated with the supersense tagger. A term by document matrix describing the whole corpus was extracted, where the terms adopted are in the form `term#supersense`, as for example `radio#artifact`. To filter out less reliable low-frequency terms, we considered only those terms occurring in more than 3 documents in the corpus, obtaining a vocabulary of about 450,000 terms. The singular value decomposition (SVD) process was performed by considering the first 100 dimensions. Three different queries were submitted to the system, each one describing a semantic domain: `Music`, `Religion` and `Sport`. In order to perform this step thresholds θ_d and θ_t have been empirically set to *0.6*, for terms, observing that these assignments provide good quality domain specific material for any query. Next, in order to guarantee the comparison with the related works taken as baselines for evaluation, the items labeled as *noun.person* are selected from the dataset furnished by the supersense tagger. For each one of these terms, the corresponding Wikipedia pages are downloaded and parsed so as to isolate the categories. At this point, the algorithm evaluates the patterns that are considered relevant for person names against the categorical information we extracted from the source pages and if one or more matches are found, the query is considered to be a *infimum* concept otherwise is classified as an *intermediate* concept. We provide a description of the employed procedure in figure 1. The algorithm is computationally very simple but efficient.

We mention that the three examined domains are disjoint and each particular individual in the entry set is already assigned to `music`, `sport` or `religion`. If none of the domain or general terms is found, the query is considered to be a something else and so discharged.

```
Input: items detected by the supersense tagger
Output: items classified as person name or other
Foreach (Ii in Ed)
{
  If (Ii is a noun.person)
  {
    If (Ii in M)
    {
      If ((Ii in Mp) or (Ii in Gc)) {Ii is a Pn}
      Else {Ii is a O}
    }
    Elseif (Ii in S)
    {
        If ((Ii in Sp) or (Ii in Gc)) {Ii is a Pn}
        Else {Ii is a O}
    }
    Elseif (Ii in R)
    {
        If ((Ii in Rp) or (Ii in Gc)) {Ii is a Pn}
        Else {Ii is a O}
    }
  }
}
```

Fig. 1: *An algorithmic overview of our person names discrimination procedure. Notations: Ii- input item; Ed - entry dataset; M, S, R - respectively music, sport or religion domain; Mp, Sp, Rp - respectively music, sport and religion related patterns for person name recognition; Gc - generic patterns for person name recognition; Pn - person name; O - class.*

7 Evaluation

In the following, an evaluation of the person names separation method that employs Wikipedia. Some relevant statistics about the dataset we employed are to be found in table 2.

Total number of noun.person items	529
Number of items processed with Wikipedia	424
Number of person names	214
Number of other concepts	210

Table 2: *Supersenses categories for nouns.*

The initial dataset contained 529 items labeled as *noun.person*, out of which it was possible to analyze 424 using the collaborative encyclopedia. The phrases belong to three domains: `sport`, `music` and `religion`. Our method could not be applied to around 20% of the phrases, that is items which do not have Wikipedia entries or the relevance of the returned pages is too low (we imposed a threshold at 10%). In order to assess the quality of the obtained results, we manually labeled the components of the remaining dataset discriminating between names (214 individuals) and others (210). After the analysis, 220 items were considered as being person names, with 193 hits and 27 errors. The errors are of two types: either the person names are mistaken as concepts (20 times) or inversely (7 times).
The method provides a good match in 87.4% of the cases analyzed with Wikipedia. Hereafter, we present three failure examples of where person names were misclassified as concepts. These examples illustrate different problems we encountered:

1. *man Jesus* - a phrase that was wrongly segmented and has no directly corresponding Wikipedia page

2. *Abimelech* - none of the patterns the method looks for is to be found on the page

3. *Dean Garcia* - there is an automatic redirection towards the entry corresponding to *The Curve*, a music band *Dean Garcia* was in

The segmentation errors are the hardest to deal with because it is highly probable that the algorithm matches an incorrect Wikipedia page with the query. In table 3, we synthesize the precision results obtained when using only Wikipedia.

	Precision
Overall	87.4%
Sport domain	100%
Music sport	92.2%
Religion domain	80.6%

Table 3: *Precision results when using only Wikipedia to discriminate person names from intermediate concepts.*

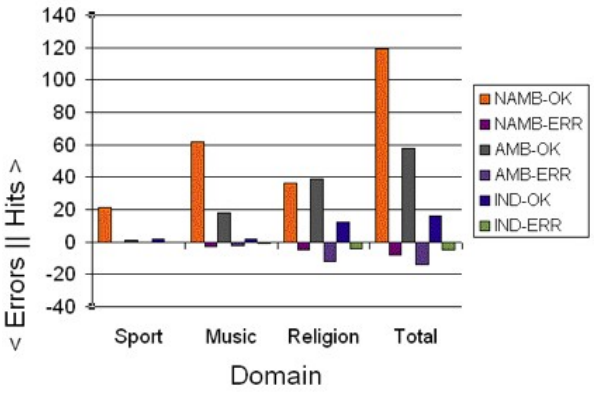

Fig. 2: *Distribution of the number hits and errors with respect to the type of the analyzed Wikipedia pages for the three analyzed domains. Notations: NAMB - non-ambiguous page; AMB - ambiguous page; IND - page without an exact match. OK stands for a hit and ERR for an error.*

The hits were distributed as follows: 24 for sports (100% accuracy), 82 for music (92.2% precision) and 87 for religion (80.6% accuracy). These values are consistent with the fact that the entries in the encyclopedia were more ambiguous for religion (62%) than for music (26.1%) and sport (12.5%). In figure 2, we analyze the hits and the errors for each one of the three domains for the three types of responses we obtained from Wikipedia: non-ambiguous pages, ambiguous ones and pages without an exact match. As one would expect, the highest accuracy is obtained in situations when there is no ambiguity (93.7%) and lower values appear for ambiguous pages (80.6%) or no exact match (76.2%). In the first case, the categories are directly analyzable, while in the latter situations, the algorithm must choose among several options and the probability of failure is increased.

WordNet provides a distinction between concepts and instances and it is possible to use it so as to ameliorate our separation method. The rationale for using the lexical hierarchy in a post-processing phase is that it is not very developed on the instance side (in all there are less than 20000 items catalogued as instances). Once the person names are completed using Wikipedia we propose a post-processing of the results employing WordNet. The total number of errors drops from 27 to 17 and the overall precision of the method passes from 87.4% to 92.1%.

	Precision
Overall	92.1%
Sport domain	100%
Music sport	96.6%
Religion domain	83.3%

Table 4: *Precision results when combing Wikipedia and WordNet to discriminate person names from intermediate concepts.*

If we look at the individual domains (see table 4), the precision for music is now at 96.6% (from 92.2%) and at 83.3% for religion (from 80.6%). This phase is based on the separation between instances that are ranged under person or deity in the lexical hierarchy and the other concepts and it allows the correction of such errors like Rama, who is now recognized as an avatar of Vishnu. Allah who is now recognized as an individual entity or rappers, a term that is correctly ranged among classes.

In comparison, the dataset for person names in [19] is similar in size with the one we evaluate here and precision results are reported for the use of Wikipedia and Wikipedia and WordNet combined together. The precision of person name detection in [19] is of 61%, while we obtain 87% of correct classifications. If we compare the results when adding WordNet in the framework, it passes to 86% in [19] and at 92% in the present work. A noteworthy difference is that, with the introduction of the post processing step, the recall is unaffected here while it passes from 77% to 30% in [19]. This is a consequence of the fact that we only use WordNet for those items having an entry in the lexical hierarchy, while the authors of [19] employ it to evaluate the entire person name space. When comparing our results with those in [13], we obtain slightly better results when analyzing unambiguous pages (93.7% vs 90.1%). When using all the pages, the precision for our method lowers at 87.4% because we analyze equally ambiguous pages (where errors appear more often), which is not the case in [13]. When adding WordNet in a postprocessing step, our method outperforms that in [13] by 2% even if ambiguous pages are included.

7.1 Discussion

We present hereafter some of the benefits obtained from introducing the semiautomatic taxonomy construction using SST and Wikipedia. In figure 3 we present an excerpt of the result from the musical domain hierarchy. Second, as one individual can be representative for one or more classes, an instance can be

assigned to a single category (*Nick Cave*) or to several (*Busoni*, *Stevie Wonder*). Third, this method provides a mean to deal with synonmy within domains. For example, in music domain Busoni is synonym with *Ferruccio Busoni* and these two phrases will compose a single instance in the conceptual hierarchy. A similar example is that of *Edberg* and *Stefan Edberg*, both terms appear in the initial dataset and, after comparing the Wikipedia categories, the two are considered as synonyms is the *sport* hierarchy. This grouping is allowed by the fact that, after disambiguation both terms point toward the same page in the encyclopedia.

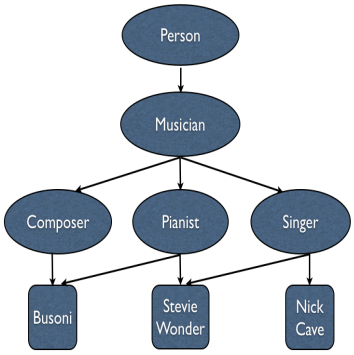

Fig. 3: *A excerpt from the* music *hierarchy.*

The segmentation of the terms provided by the tagger can be incomplete (e.g.: *Cool J* or *W Tozer* - stop signs are encountered). With the use of Wikipedia, these phrases are completed and the hierarchy will contain their correct spelling, that is *L.L. Cool J* and *A. W. Tozer* (but also *Aiden Wilson Tozer*, the complete spelling of *A. W. Tozer*). This feature is an effect of processing Wikipedia answers even when there is no exact match for a given query. Here we exploit the relevance function provided when searching the encyclopedia. The approximate answers are parsed if the relevance returned by Wikipedia is higher than 10% (empirical threshold) and if the strings in the query belong to the title of the page. By doing this, we avoid treating pages with to small relevance and those that are only mention the terms we queried, without being dedicated to its description.

8 Conclusion and Future Work

We presented a taxonomy construction method which is based on the utilization of SST and Wikipedia categories. The method is useful for automatically structuring domain related concept hierarchies based on semistructured information in the encyclopedia. Initial evaluations for the person name recognition were presented and the results encourage us to pursuit the work and extend the separation method to other named entities. In the future, we intend to add ontological relations in the domain hierarchies we extract. Wikipedia pages contain other structured or semistructured information aside categories which can be exploited to enrich the created semantic structure. If we take the case of a musician, we can extract knowledge about the musical genres she plays, her collaborators, nationality or activity years. These information can be parsed from tables in the Wikipedia entries or from the full text and it will be used to feed relations like "playsMusicType", "hasCollaborators", "hasNationality", "hasActivityYears".

References

[1] E. Alfonseca, M. Ruiz-Casado, M. Okumura, and P. Castells. Towards large-scale non-taxonomic relation extraction: Estimating the precision of rote extractors. pages 49–56, July 2006.

[2] A. Beneti, W. Hammoumi, E. Hielscher, M. Mller, and D. Persons. Automatic generation of fine-grained named entity classifications. Technical report, University of Amsterdam, February 2006.

[3] P. Buitelaar, D. Olejnik, and M. Sintek. A protégé plug-in for ontology extraction from text based on linguistic analysis. In *Proceedings of the 1st European Semantic Web Symposium (ESWS)*, 2004.

[4] R. C. Bunescu and M. Pasca. Using encyclopedic knowledge for named entity disambiguation. In *EACL*, 2006.

[5] M. Ciaramita and Y. Altun. Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of EMNLP-06*, pages 594–602, Sydney, Australia, 2006.

[6] M. Ciaramita and M. Johnson. Supersense tagging of unknown nouns in wordnet. In *Proceedings of EMNLP-03*, pages 168–175, Sapporo, Japan, 2003.

[7] P. Cimiano, A. Pivk, L. Schmidt-Thieme, and S. Staab. Learning taxonomic relations from heterogeneous evidence, 2004.

[8] P. Cimiano and J. Vlker. Towards large-scale, open-domain and ontology-based named entity classification. In G. Angelova, K. Bontcheva, R. Mitkov, and N. Nicolov, editors, *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 166–172, Borovets, Bulgaria, SEP 2005. INCOMA Ltd.

[9] M. Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of EMNLP-02*, 2002.

[10] K. Deschacht and M.-F. Moens. Efficient hierarchical entity classifier using conditional random fields. *Proceedings of the 2nd Workshop on Ontology Learning and Population, Sydney, 22 July.*, pages 33–40, July 2006.

[11] C. Fellbaum. *WordNet. An Electronic Lexical Database*. MIT Press, 1998.

[12] B. Ganter, G. Stumme, and R. Wille, editors. *Formal Concept Analsysis:Foundations and Applications*. Springer, 2005.

[13] J. Kazama and K. Torisawa. Exploiting wikipedia as external knowledge for named entity recognition. In *proceedings of EMNLP-07*, 2007.

[14] B. Magnini, E. Pianta, O. Popescu, and M. Speranza. Ontology population from textual mentions: Task definition and benchmark. *Proceedings of the 2nd Workshop on Ontology Learning and Population, Sydney, 22 July.*, pages 26–32, July 2006.

[15] R. Navigli and P. Velardi. Ontology enrichment through automatic semantic annotation of on-line glossaries. In *EKAW*, pages 126–140, 2006.

[16] D. Picca, A. Gliozzo, and M. Ciaramita. Semantic domains and supersens tagging for domain-specific ontology learning. In *proceedings RIAO 2007*, May 2007.

[17] M. Sanderson and B. Croft. Deriving concept hierarchies from text. pages 206–213, 1999.

[18] M. K. Smith, C. Welty, and D. L. McGuinness. Owl web ontology language guide.

[19] A. Toral and R. Muoz. A proposal to automatically build and maintain gazetteers for named entity recognition by using wikipedia. In *Proceedings of the workshop on NEW TEXT Wikis and blogs and other dynamic text sources*, 2006.