LT3: A Multi-modular Approach to Automatic Taxonomy Construction

Els Lefever

LT³, Language and Translation Technology Team
Department of Translation, Interpreting and Communication – Ghent University
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
`els.lefever@UGent.be`

Abstract

This paper describes our contribution to the SemEval-2015 Task 17: Taxonomy Extraction Evaluation. We propose a hypernym detection system combining three modules: a lexico-syntactic pattern matcher, a morpho-syntactic analyzer and a module retrieving hypernym relations from structured lexical resources. Our system ranked first in the competition when considering the gold standard and manual evaluation, and second in the overall ranking. In addition, the experimental results show that all modules contribute to finding hypernym relations between terms.

1 Introduction

Because of globalization and rapid technological evolution, it is no longer feasible to manually create and manage taxonomies for the large variety of scientific and technological (sub)domains. In addition to domain-specific terminology, companies desire to build their own mono- or bilingual taxonomies containing the relevant sector and company specific terminology. This clear need for automation has encouraged researchers to investigate how terminological and semantically structured resources such as taxonomies or ontologies can be automatically constructed from text (Biemann, 2005).

Different approaches have been proposed to automatically detect hierarchical relations between terms: pattern-based approaches (Hearst, 1992; Pantel and Ravichandran, 2004), statistical and machine learning techniques (Ritter et al., 2009), distributional approaches (Caraballo, 1999; van der Plas and Bouma, 2005; Lenci and Benotto, 2012), morpho-syntactic approaches (Tjong Kim Sang et al., 2011) and word class latices (Navigli and Velardi, 2010).

SemEval-2015 Task 17: Taxonomy Extraction Evaluation (Bordea et al., 2015) is concerned with automatically finding relations between pairs of terms and organizing them in a hierarchical structure. In this way, the task assumes that a list of domain specific terms is already available in order to focus on the relation detection between these terms.

To tackle this SemEval taxonomy learning task, we propose a multi-modular approach that combines lexico-syntactic, morphological and external structured lexical information. We will describe our hypernym detection system in Section 2. The results of the evaluation are presented in Section 3, while Section 4 concludes this paper.

2 System Description

Our hypernym detection system contains three main components: a lexico-syntactic pattern-based approach, a morpho-syntactic analyzer and a module retrieving hypernym relations from a structured lexical resource. Each module takes as input a domain specific term list.

2.1 Pattern-based Approach

The first module that automatically detects hypernym relations is a lexico-syntactic pattern-based approach, based on Hearst (1992). These patterns are implemented as a list of regular expressions containing lexicalized expressions (e.g. *like*), as well as isolated Part-of-Speech tags (e.g. noun) and chunk tags, which represent different Part-of-Speech sequences

(e.g. noun phrase (NP) = determiner + adjective + noun, adjective + noun, etc.). An example of these manually defined patterns is "NP {, NP}* {,} or/and other NP",[1] as in "*green beans, carrots, peas and other vegetables*", which results in four hypernym pairs, being (*vegetables, green beans*), (*vegetables, carrots*), (*vegetables, peas*) and (*vegetables, onions*).

Domain specific corpus. As this module aims to find hypernym relations by detecting terms occurring in specific lexico-syntactic constructions, we first needed to compile a domain specific corpus containing these terms. To compile the corpus, we used the the BootCaT toolkit (Baroni and Bernardini, 2004), which can be used to build a specialized web-based corpus starting from a list of seed terms. We considered the different term lists for task 17 as the "seed terms" to build the domain specific corpora, by allowing 10 queries per seed term. Due to technical reasons (the Bing search engine that is used by BootCat only allows 5000 queries per user account), we only compiled corpora for three domains, being *equipment, food and science*. As a post-processing step, we removed all sentences containing (1) only URL links or (2) no domain specific term, resulting in three corpora containing about 12 million tokens for the *food* domain, 6 million tokens for the *equipment* domain and 27 million tokens for the *science* domain.

Linguistic preprocessing. We performed a number of linguistic preprocessing steps in order to enrich the original web-based corpus: (1) tokenisation, (2) Part-of-Speech Tagging, (3) Lemmatisation and (4) Chunking. All linguistic preprocessing was per performed by means of the LeTs Preprocess toolkit (Van de Kauter et al., 2013).

Lexico-syntactic pattern matching. The resulting linguistically preprocessed corpus is the input for the pattern-based module. Example 1 shows a sentence matching the pattern:

$$\{other\}* NP\ such\ as\ NP\ \{, NP\} * \{(and\text{-}or)\ NP\}*$$

resulting in the two hypernym pairs (*cranberry products, tablets*) and (*cranberry products, capsules*). As can be seen in example 1, the lexi-

calised parts of the patterns (*other* and *such as* in this case) are not considered for the generation of the hyponym–hypernym tuples.

```
(1)  other other JJ B-NP
     cranberry cranberry NN I-NP
     products product NNS I-NP
     such such JJ B-AP
     as as IN I-AP
     tablets tablet NNS B-NP
     and and CC O
     capsules capsule NNS B-NP
```

We optimized the pattern-based model presented by (Lefever et al., 2014) in different ways. The efficiency of the module was improved by only considering noun phrases containing a maximum of 6 consecutive nouns and by ignoring named entities. This appeared to be necessary as the web-based corpus contains a lot of lists and enumerations, causing problems for the recursive way the regular expressions are built. Precision, on the other hand, was improved by ignoring tuples containing both terms as hypernym and hyponym (e.g. hand truck – truck and truck – hand truck) and by only considering patterns that revealed to obtain high precision in previous research (Lefever et al., 2014).

Finally, the output of the pattern-based module is filtered by only considering tuples where both terms (either lemma or full form) occur in the term list of the considered domain.

2.2 Morpho-syntactic Analyzer

Our second hypernym detection module applies a morpho-syntactic approach where the morphological structure of compound terms is used to extract a hypernym-hyponym relation from this term. This approach is inspired by the head-modifier principle (Sparck Jones, 1979) stating that in a compound noun, the linear arrangement of the compound parts expresses the kind of information being conveyed, the head referring to the more general semantic category, whereas the modifiers restrict the sense of the compound term. This way, the complete compound term can be considered as a hyponym of the head term. We implemented rules for three different syntactic hypernym-hyponym relations in compounds:

---

[1]Curly brackets indicate optional parts of the pattern.

1. **Single-word terms**: *If term T0 is a suffix string of term T1, T0 is considered to be a hypernym of T1.* Examples of hypernym pairs detected within single-word terms are (*sachertorte, torte*), (*candlepin, pin*) and (*psycholinguistics, linguistics*).

2. **Multiword terms**: *If term T0 is the head term of term T1, T0 is considered to be a hypernym of T1.* It is important to mention that we also allow multiple possible hypernyms in case different terms occur as suffixes of the compound term, e.g. *phu quoc fish sauce* is the hyponym of both *sauce* and *fish sauce*. As the head of a nominal phrase appears at the right edge of a multiword NP in English, the last constituent of the NP is regarded as the head of the compound, and thus as the hypernym of the complete term, as is the case in the generated hypernym pair (*béarnaise sauce, sauce*).

3. **Complex prepositional phrases**: *If term T0 is the first part of a term T1 containing a noun phrase + preposition + noun phrase, T0 is considered to be a hypernym of T1.* In the case of a prepositional compound phrase, the head is situated at the left edge of the compound term. Examples of such hypernym pairs are (*sociology of culture, sociology*) and (*soup all'imperatrice, soup*)

In addition, we added some restrictions to these general rules in order to improve the precision of the module. First, we set a threshold of minimum three characters for the detection of valid hypernyms. An example of invalid hypernyms filtered out this way is *tu* that could be detected as a hypernym of *pesarattu*, both terms occurring in the food term list. Second, we noticed that food terms (etc. dishes) are often loan words from other languages. Therefor we added a list of foreign adjectival affixes (e.g. french affix *al/ale*) that should not be considered as a hypernym of the compound term. This way we prevent for instance *ale* to be detected as the hypernym of *chicken provencale* or *café royale*.

2.3 Structured Lexical Resources: WordNet

The third hypernym detection module retrieves information from an external lexical resource, being WordNet in this case. This module looks up the synsets in WordNet for all domain-specific terms and retrieves all hypernyms appearing in the full hierarchical path of these synsets. Hypernym tuples containing identical terms were removed. Examples of hyponym-hypernym pairs retrieved from WordNet are (*semantics, science*) and (*semantics, linguistics*).

2.4 Combined System

To generate the final list of hypernym relations, we combined the output of all three modules and removed all doubles from the hyponym–hypernym pair list.

3 Results

The resulting taxonomies are evaluated through comparison with gold standard relations collected from existing domain specific ontologies and WordNet. In addition, expert evaluation has been performed on the hypernym relations submitted by the participants. The system organizers calculated precision, recall and F-score as well as a cumulative Fowlkes & Mallows measure, which is inspired by clustering evaluation and takes into account the hierarchical structure of the gold standard taxonomy and the taxonomy that is produced by the system.

In addition, a number of structural measures were calculated such as the number of distinct vertices and edges, the number of connected components and intermediate nodes to evaluate whether the taxonomy connects all nodes with the root. From this evaluation it was clear that our taxonomy contains cycles, which is conflicting with correct hierarchical relations. For more detailed information about the gold standards and evaluation metrics, we refer to (Bordea et al., 2015). Table 1 lists the averaged Precision, Recall, F-measure and Fowlkes & Mallows scores for all participating systems, while Table 2 lists the individual scores for the three domains in which we participated. The very high recall scores for the *WN* data sets can be explained by the fact that our system also contains a module that retrieves hypernym relations from WordNet.[2]

---

[2]We included a WordNet module, since originally only BabelNet was specified as the gold standard for the task. At evaluation time, however, WordNet was also used to evaluate the taxonomies.

	INRIASAC	LT3	ntnu	QASSIT	TALN-UPF	USAAR-WLV
Precision	0.1721	0.3612	0.1754	0.1563	0.0720	0.2014
Recall	0.4279	0.6307	0.2756	0.1588	0.1165	0.3139
F-score	0.2427	0.3886	0.2075	0.1575	0.0798	0.2377
Fowlkes & Mallows	0.3278	0.4130	0.0582	0.3882	0.2635	0.0770

Table 1: Comparative evaluation of all participating systems considering the average Precision, Recall, F-measure and Cumulative Fowlkes & Mallows measure scores.

Test set	Precision	Recall	F-Score	F&M
Equipment	0.7021	0.3219	0.4414	0.1137
Food	0.2892	0.2974	0.2932	0.2163
Science	0.4013	0.3806	0.3907	0.3303
WN_Equipment	0.3168	0.9484	0.4749	0.6892
WN_Food	0.2155	0.9719	0.3528	0.5899
WN_Science	0.2422	0.8639	0.3783	0.5391

Table 2: Precision, Recall, F-measure and Cumulative Fowlkes & Mallows measure scores for all test sets.

As we wanted to gain more insights in the contribution of the different modules, we also calculated precision and recall per module for the different term lists. The results per module and for the combined system are shown in Table 3.

		Morpho-syntactic Module	Pattern-based Module	Word-Net Module	full system
WN_Equip	P	0.696	0.143	0.320	0.317
	R	0.245	0.008	0.932	0.948
Equipment	P	0.791	0.214	0.310	0.702
	R	0.307	0.005	0.021	0.322
WN_Food	P	0.613	0.204	0.230	0.216
	R	0.242	0.091	0.958	0.972
Food	P	0.602	0.191	0.219	0.289
	R	0.176	0.059	0.134	0.297
WN_Science	P	0.696	0.215	0.778	0.242
	R	0.270	0.063	0.857	0.864
Science	P	0.641	0.292	0.277	0.401
	R	0.273	0.045	0.144	0.381

Table 3: Precision (P) and recall (R) from relation overlap scores per hypernym detection module per domain.

We notice that the WordNet module indeed achieves very high recall for the *WN* test sets, but that the recall for the more technical term lists is much lower (with only 0.021 for the *Equipment* data set). The recall achieved by the pattern-based module is also very modest, with scores ranging from 0.005 (*Equipment*) to 0.063 (*WN_Science*). The Morpho-syntactic Module, on the other hand, contributes in a consistent way to the recall for all term lists. Finally, table 3 also shows that the obtained recall by the system that combines all different mod-

ules consistently beats the recall of the individual modules for all test sets. With regard to precision, we observe that the morpho-syntactic approach obtains very good results for all the different test domains, resulting in system precision scores that outperform all participating systems.

A qualitative analysis of the output revealed shortcomings of the different hypernym detection modules. As discussed above, the *morpho-syntactic* module achieves good recall. The downside is that the module clearly over generates. Examples of invalid hypernym pairs are for instance (*pineapple juice, apple juice*), (*hot and sour soup, sour soup*) and (*ice cream, cream*). Although WordNet is a manually verified taxonomy, we also discovered invalid hypernym pairs in the output of the WordNet module. For the *food* domain, for instance, we discovered that all beverages have "food" as an inherited hypernym, resulting in hypernym pairs such as (*pineapple juice, food*) and (*absinth, food*).

4   Conclusion

To tackle SemEval Task 17: Taxonomy Extraction Evaluation, we proposed a hypernym detection system combining a lexico-syntactic pattern matcher, a morpho-syntactic analyzer and a module retrieving hypernym relations from WordNet and showed promising results for the different test domains. Analyzing the recall per hypernym detection module revealed that all modules made distinct contributions to the final hyponym–hypernym list.

In future work, we would like to improve the recall of the system by exploring additional hypernym detection modules (e.g. a distributional model built on the basis of the domain corpora). We also plan to build a dedicated module that constructs an actual taxonomy based on the extracted hierarchical relations by removing relations that result in cycles.

References

Marco Baroni and Silvia Bernardini. 2004. BootCaT: Bootstrapping Corpora and Terms from the Web. In *In Proceedings of LREC 2004*, pages 1313–1316.

Chris Biemann. 2005. Ontology Learning from Text: A Survey of Methods. *LDV Forum*, 20(2):75–93.

Georgeta Bordea, Paul Buitelaar, Stefano Faralli, and Roberto Navigli. 2015. Semeval-2015 task 17: Taxonomy extraction evaluation (texeval). In *Proceedings of the 9th International Workshop on Semantic Evaluation*.

Sharon A. Caraballo. 1999. Automatic acquisition of a hypernym-labeled noun hierarchy from text. In *Proceedings of ACL-99*, pages 120–126, Baltimore, MD.

Marti Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the International Conference on Computational Linguistics*, pages 539–545.

Els Lefever, Marjan Van de Kauter, and Véronique Hoste. 2014. HypoTerm: detection of hypernym relations between domain-specific terms in Dutch and English. *Terminology*, 20(2):250–278.

Alessandro Lenci and Giulia Benotto. 2012. Identifying hypernyms in distributional semantic spaces. In *Proceedings of the first Joint conference on Lexical and Computational Semantics (*SEM)*, pages 75–79, Montréal, Canada.

Roberto Navigli and Paola Velardi. 2010. Learning Word-Class Lattices for Definition and Hypernym Extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327, Uppsala, Sweden.

Patrick Pantel and Deepak Ravichandran. 2004. Automatically labeling semantic classes. In *Proceedings of HLT/NAACL-04*, pages 321–328, Boston, MA.

Alan Ritter, Stephen Soderland, and Oren Etzioni. 2009. What is this, anyway: Automatic hypernym discovery. In *Proceedings of Association for Advancement of Artificial Intelligence Spring Symposium on Learning by Reading and Learning to Read*, pages 88–93.

Karen Sparck Jones. 1979. Experiments in relevance weighting of search terms. *Information Processing and Management*, 15:133–144.

Erik Tjong Kim Sang, Katja Hofmann, and Maarten De Rijke. 2011. Extraction of hypernymy information from text. In A. Van den Bosch and G. Bouma, editors, *Interactive multi-modal question-answering*, Series: Theory and Applications of Natural Language Processing, pages 223–245.

Marjan Van de Kauter, Geert Coorman, Els Lefever, Bart Desmet, Lieve Macken, and Véronique Hoste. 2013. LeTs Preprocess: the Multilingual LT3 Linguistic Preprocessing Toolkit. *Computational Linguistics in the Netherlands Journal*.

Lonneke van der Plas and Gosse Bouma. 2005. Automatic acquisition of lexico-semantic knowledge for question answering. In *Proceedings of the IJCNLP Workshop on Ontologies and Lexical Resources*, Jeju Island, Korea.