Dutch Hypernym Detection: Does Decompounding Help?

Ayla Rigouts Terryn, Lieve Macken and Els Lefever
LT^3, Language and Translation Technology Team, Ghent University
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
ayla.rigoutsterryn,lieve.macken,els.lefever@ugent.be

Abstract

This research presents experiments carried out to improve the precision and recall of Dutch hypernym detection. To do so, we applied a data-driven semantic relation finder that starts from a list of automatically extracted domain-specific terms from technical corpora, and generates a list of hypernym relations between these terms. As Dutch technical terms often consist of compounds written in one orthographic unit, we investigated the impact of a decompounding module on the performance of the hypernym detection system. In addition, we also improved the precision of the system by designing filters taking into account statistical and linguistic information. The experimental results show that both the precision and recall of the hypernym detection system improved, and that the decompounding module is especially effective for hypernym detection in Dutch.

Keywords: terminology, hypernym detection, decompounding

1. Introduction

Structured lexical resources have been proven essential for different language technology applications such as efficient information retrieval, word sense disambiguation or coreference resolution. Also from a business perspective, ontologies and user-specific taxonomies appear to be very useful (Azevedo et al., 2015). Companies like to have their own mono- or multilingual enterprise semantic resources, but manual creation of structured lexical resources appears to be a very cumbersome and expensive task. Therefore, researchers have started to investigate how terminological and semantically structured resources such as ontologies or taxonomies can be automatically constructed from text (Biemann, 2005).

This paper focuses on the task of automatic hypernym detection from text, which can be considered the central task of automatic taxonomy construction. Detecting hyponym–hypernym tuples consists in finding subtype-supertype relations between terms of a given domain of interest. Different approaches have been proposed to automatically detect these hierarchical relations between terms.

Pattern-based approaches, which are inspired by the work of Hearst (1992), deploy a list of lexico-syntactic patterns able to identify hypernym pairs in text. An example of these manually defined Hearst patterns is "*NP* {*, NP*}* {*,*} *or other NP*", as in "Bruises, wounds, broken bones or other injuries", which results in three hypernym pairs: (*bruise, injury*), (*wound, injury*) and (*broken bone, injury*). The lexico-syntactic approach of Hearst has been applied and further extended for English (Pantel and Ravichandran, 2004) and various other languages such as Romanian (Mititelu, 2008), French (Malaisé et al., 2004) and Dutch (Lefever et al., 2014a). Researchers have defined these lexico-syntactic patterns manually (Kozareva et al., 2008), but also statistical and machine learning techniques have been deployed to automatically extract these patterns and to train hypernym classifiers (Ritter et al., 2009).

Other researchers have applied **distributional approaches** to automatically find hypernym pairs in text (Caraballo, 1999; Van der Plas and Bouma, 2005). Distributional approaches start from the assumption that semantically related words tend to occur in similar contexts. The hypernym detection task is then approached as a clustering task, where semantically similar words are clustered together and the hierarchical structure of the clustering is used to express the direction of the hypernym-hyponym relation. An extension of this approach is the distributional inclusion hypothesis, which has been the inspiration to use directional similarity measures to detect hypernym relations between terms (Lenci and Benotto, 2012). More recently, the potential of word embeddings, which are word representations computed using neural networks, has been investigated to predict hypernyms (Fu et al., 2014; Rei and Briscoe, 2014).

The morphological structure of terms has also been used as an information source to extract hypernym–hyponym relations from compound terms (Tjong Kim Sang et al., 2011). These **morpho-syntactic approaches** have shown to be successful for technical texts, where a large number of the domain specific terms appear to be compounds (Lefever et al., 2014b). The latter approaches start from the assumption that the full compound can be considered as the hyponym, whereas the head term is then to be considered as the hypernym term of the compound. Other approaches use heuristics to extract hypernym relations from **structured (collaborative) resources** such as Wikipedia. Ponzetto and Strube (2011) use methods based on connectivity in the Wikipedia network and lexico-syntactic patterns to automatically assign hypernym labels to the relations between Wikipedia categories. Navigli and Velardi (2010) use word class lattices, or directed acyclic graphs, to develop a pattern generalization algorithm trained on a manually annotated training set, which is able to extract definitions and hypernyms from web documents.

In this paper we propose a domain-independent approach to automatically detect hyponym–hypernym relations between Dutch domain specific terms. To find hierarchical relations between terms, we propose a data-driven approach combining a lexico-syntactic pattern-based module, a morpho-syntactic analyzer and a decompounding module. Given the very productive compounding system in Dutch, we expect to improve the recall of the hypernym detection system by decompounding all Dutch terms. In addition, we

also investigate the impact of filtering the relations, based on the results of automatic terminology extraction.
The remainder of this paper is structured as follows. In Section 2., we describe the annotated data sets we constructed. Section 3. presents our system, which combines fully automatic term extraction with a data-driven hypernym detection approach. In Section 4., the experimental results are discussed, while Section 5. formulates some conclusions and ideas for future research.

2. Data Set Construction

In order to evaluate the impact of filtering and decompounding on the hypernym detection performance, we constructed a development and a test corpus. The development corpus consists of manually annotated, highly specialized texts for the dredging and financial domains in Dutch. Dredging texts are annual reports from a Belgian dredging company, whereas the financial texts are news articles from the business newspaper De Tijd. For the test corpus, we used a technical manual for mobile screens. The development corpus was used to tune the linguistic and statistical filtering of the terminology extraction output (See Section 4.2.). Both the development and test corpus were manually annotated with BRAT (Stenetorp et al., 2012). This web-based tool was used to annotate all terms and named entities and the hypernym relations between them. Figure 1 shows an example sentence in which 5 terms: *persoonlijke beschermingsmiddelen* (English: personal protective equipment), *beschermingsmiddelen* (English: protective equipment), *veiligheidsbril* (English: safety goggles), *bril* (English: glasses) and *handschoenen* (English: gloves) and the hypernym relations between those terms were annotated. The annotation results were then exported and processed into a gold standard. The manual annotation for the development and test corpus allowed us to measure both precision and recall. Another advantage was that the evaluation did not have to rely on general-purpose inventories, such as WordNet, and could therefore also accurately evaluate specialized terms which do not occur in lexical inventories. Table 1 gives an overview of the number of hypernym relations in the development en test corpora, which contain each around 10,000 tokens.

Gold Standard	# Relations in gold standard
Dredging (Development)	822
Financial (Development)	364
Technical manual (Test)	480

Table 1: Gold Standard relations per data set.

3. System Description

The hypernym detection system starts from a raw domain-specific corpus that is first linguistically preprocessed by means of the LeTs Preprocess toolkit (Van de Kauter et al., 2013), which performs tokenization, Part-of-Speech tagging, lemmatization and chunking. The preprocessed corpus is then the input for both the terminology extractor and the different modules of the hypernym detection system.

3.1. Terminology Extraction

In order to automatically extract domain specific terms from our corpus, we applied the TExSIS terminology extraction system (Macken et al., 2013). TExSIS is a hybrid system, which first generates syntactically valid candidate terms and then applies statistical filtering (termhood as implemented by Vintar (2010), log-likelihood, etc.), resulting in a list of domain specific single and multiword terms. Examples of terms extracted by TExSIS are 'rupsbanden' (caterpillar tracks) and 'brandstofinjectiepomp' (fuel injection pump). The resulting term lists are then used to filter the results of the hypernym detection. For example, the hypernym detection system might discover that 'language' is a hypernym of 'English'. Although this is correct, it is likely that for specialized technical texts, the user is not interested in this particular relation and wants to focus on terms that are relevant to the field, such as 'iron ore' as a hyponym of 'primary raw materials'.
Apart from this original filtering, analysing the results of the development corpus revealed some additional patterns that could be used to further tune the terminology extraction to the hypernym detection and improve the precision without hurting the recall. The first correlation we discovered between the extracted terms and the terms in the development gold standard, was the termhood score. The higher the termhood score of the extracted terms, the more likely the term would appear in our gold standard. Figure 2 shows the percentage of terms that were in the gold standard of the development corpus out of all the terms with a termhood score within a certain range.
Another correlation was discovered in the Part-of-Speech tags of the terms. Terms in the gold standard had less diverse PoS tags than terms that were not in the gold standard and were mostly restricted to less 'complicated' PoS categories, such as a single noun or an adjective-noun combination (see figure 3 and 4). More complex PoS categories such as 'Noun Preposition Determiner Adjective Noun' were never found in the gold standard. Based on this information, we experimented with different filters on the development corpus to find the ones that discarded the most irrelevant terms, without rejecting terms present in the gold standard.

3.2. Hypernym Detection

For the automatic extraction of Dutch hypernym relations, we combined the lexico-syntactic pattern-based approach and morpho-syntactic analyzer of Lefever et al. (2014a) with a newly developed hypernym detection module integrating decompounding information. The current system takes as input a list of automatically extracted terms and a linguistically preprocessed corpus, and generates a list of hyponym–hypernym tuples.

3.2.1. Pattern-based Module

The first hypernym detection module is a pattern-based module. The patterns are implemented as a list of regular expressions containing lexicalized strings (e.g. *like*), isolated Part-of-Speech tags (e.g. *Noun*) and chunk tags, which consist of sequences of Part-of-Speech tags (e.g. *Noun Phrase*). For a complete list of Dutch lexico-syntactic

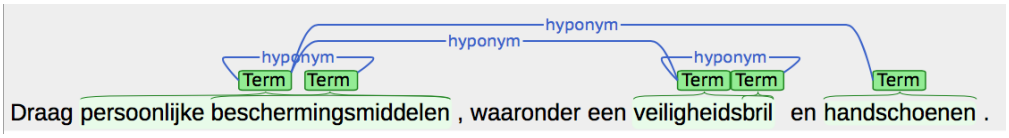

Figure 1: Annotation of terms and hypernym relations in BRAT.

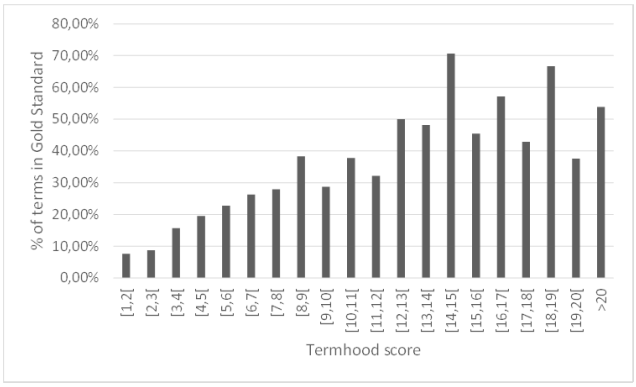

Figure 2: Likelihood that a term is in the gold standard of the development corpus.

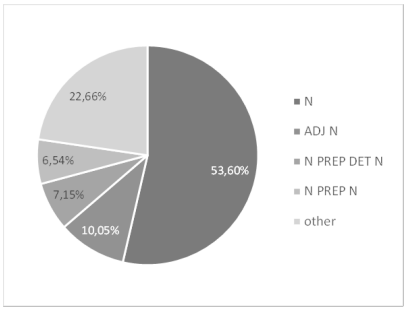

Figure 3: PoS sequences of terms not appearing in the hypernym gold standard of the development corpus (N: noun, ADJ: adjective, PREP: preposition, DET: determiner).

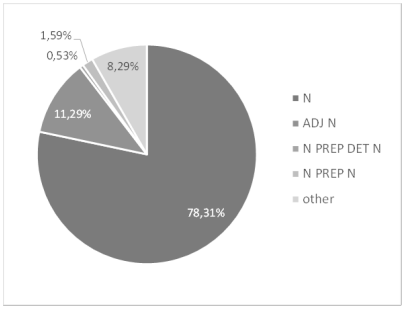

Figure 4: Part-of-Speech sequences of terms appearing in the hypernym gold standard of the development corpus.

patterns, we refer to Lefever et al. (2014a). An example of a Dutch pattern is *NP, zoals NP {,NP}* {(en/of) NP}*, matching the test sentence *veiligheidsfuncties, zoals noodstopknoppen en beschermknoppen* (English: *safety features such as emergency stop buttons and safety guards*) and resulting in the hypernym-hyponym tuples *(veiligheidsfuncties, noodstopknoppen)* and *(veiligheidsfuncties, beschermkappen)*.

3.2.2. Morpho-syntactic Module

The second module starts from the automatically generated term list to generate hypernym–hyponym tuples based on the morpho-syntactic structure of the terms. This approach is inspired by the head-modifier principle of Sparck Jones (1979), which states that the head of the compound refers to a more general semantic category, while the modifying part narrows down the meaning of the compound term. Three different morpho-syntactic rules were implemented:

single-word noun phrases. If Term1 is a suffix string of Term2, Term1 is considered as a hypernym of Term2. Example: *(mat, deurmat)* (English: *(mat, doormat)*).

multi-word noun phrases. If Term1 is the head term of Term2, Term1 is considered to be a hypernym of Term2. As the head of a noun phrase (NP) appears at the right edge of a multiword in Dutch and English, the last constituent of the NP is regarded as the head term. Example: *(afstandsbediening, draadloze afstandsbediening)* (English: *(remote control, wireless remote control)*).

noun phrase + prepositional phrase. If Term1 is the first part of Term2 containing a noun phrase + preposition + noun phrase, Term1 is considered as the hypernym of Term2. In Dutch, the head of a prepositional compound phrase is situated at the left edge of the compound term. Example: *(Raad, Raad van State)* (English: *(Council, Council of State)*).

A qualitative analysis revealed that the morpho-syntactic approach overgenerates because it has no information on the validity of the remaining part of the compound. As an example, we can refer to *soil*, which contains the term *oil*, but the remaining part *-s* is not a valid lexeme. In addition, the morpho-syntactic approach is also constrained by the occurrence of both the hypernym and hyponym term in the automatically extracted term list. To overcome both issues, we implemented a third module that takes into account decompounding information for all domain specific terms.

3.2.3. Decompounding Module

As already mentioned, the right-hand part of a compound in Dutch is the head and determines the meaning of the compound, e.g. *bediening+s+knoppen* (English: control buttons), are a special type of *knoppen* (English: buttons). This information can be used to find hypernyms, because the head of the compound is mostly also the hypernym of the compound, e.g. *knoppen* is a hypernym of *bedieningsknoppen*. Compounds can be nested and especially in technical texts, compounds of more than two components frequently occur, such as *[nood+stop]+knoppen* (English: [emergency stop] buttons). To add decompounding

# relations in gold standard = 480	Patterns	Morphosynt.	Patterns and Morphosynt.	Decompounder	All modules combined	Filter 1	Filter 2
Relations found	29	895	923	317	1091	970	961
Correct relations found	8	259	266	230	410	409	409
Precision	0.275862	0.289385	0.288191	0.725552	0.375802	0.421649	**0.425598**
Recall	0.016667	0.539583	0.554167	0.479167	**0.854167**	0.852083	0.852083

Table 2: Results for the test corpus.

information to our term list, we used the compound splitter of Macken and Teczan (in press), which is a hybrid compound splitter for Dutch that makes use of corpus frequency information and linguistic knowledge. The compound splitting tool determines a list of eligible compound constituents on the basis of word frequency information derived from a PoS-tagged Wikipedia dump of 150 million words extended with a smaller dynamically compiled frequency list derived from the extraction corpus. Part-of-speech information is used to restrict the list of possible constituents. The compound splitter selects the split point with the highest geometric mean of word frequencies of its parts (Koehn and Knight, 2003): $(\prod_{i=1}^{n} freq_p)^{1/n}$ in which *n* is the number of split points in the compound and $freq_p$ is the frequency of the component parts. The compound splitter allows a linking-s between two component parts and can be called recursively to deal with with nested compounds.
A qualitative analysis of the results revealed that this module may overgenerate as well, for example with words such as *hand+schoenen* (English: gloves, literal translation: *hand+shoes*). In this case, the decompounding module might wrongly say that *shoes* is a hypernym of *gloves*. Despite these exceptions, experiments with the development corpus showed that the precision of the decompounding module was still higher than that of the pattern-based and morphosyntactic modules for Dutch. In addition, it also deals with some of the shortcomings of the morphosyntactic module as it limits the number of split points, can correctly process compounds with a linking-s and can be called recursively for nested compounds.

4. Experimental Results

4.1. Improving Recall

To find more correct hypernym relations, we expanded the pattern-based and morphosyntactic modules with the decompounding module. Since compound terms are very common in Dutch, especially in technical texts such as the ones used for this experiment, this led to a significant increase in recall. The pattern-based and morphosyntactic modules combined were able to find 266 out of the 480 relations in the gold standard. The decompounding module on its own already found 230 correct relations and all three modules combined achieved a score of 410 correctly identified relations out of the 480 relations in the gold standard. This means a recall of 85%, an increase of 30% over the original system.

4.2. Improving Precision

The decompounding module did not overgenerate much and got a high recall, which already increased the precision of the combined system with 9%. However, by using additional filters based on the terminology extraction, the precision could be further improved. We implemented 2 different filters. The first filter was based purely on the PoS tags of the extracted terms: all terms which were assigned a PoS pattern that was not in the list created on the basis of the experiments carried out for the development corpus, were automatically filtered out. The list we used was: N, ADJ N, N N, N N N, N N N N, N PREP N, N CONJ-coord N, N PREP DET N, N PREP ADJ N. This filter works well, but may in some cases be too strict and discard some terms which are still relevant. That is why we developed an alternative filter, which filtered out all the same terms as the first filter, except if they had a termhood score of more than 10. Even though, in the case of our test corpus, the second filter made little difference, it may be a good safety precaution when recall is more important than precision. These filters can easily be adapted to focus more on precision or recall by adding or deleting certain PoS categories and by choosing a higher or lower minimum termhood score for the second filter.

5. Conclusions and Future Work

We presented proof-of-concept experiments for a data-driven hypernym detection system for Dutch, which starts from an automatically generated term list and combines a pattern-based module, a morpho-syntactic analyzer and a decompounding module. Both the precision and recall of the extracted hypernym relations clearly improved by adding the decompounding module. Precision was further improved by filtering the results of the terminology extraction based on Part-of-Speech and termhood score.
In future work, we will investigate whether our methodology, especially the decompounding module, works equally well on other languages with many compounds, such as German. In addition, we will work towards hypernym detection in multiple levels. For example, if terrier is a hyponym of dog, and dog is a hyponym of animal, then terrier is also a hyponym of animal. Ultimately, this can result in hypernym trees, which can be used to structure terminology databases. Finally, during the annotation, we came upon the problem of split compound terms, such as "gezondheid- en veiligheidsbescherming" (health and safety protection) or split multiword terms such as "elektrische en statische vonken" (electric and static sparks). It was difficult or even impossible to correctly annotate these terms with BRAT and our hypernym detection system cannot process these terms yet either. Nevertheless, this syntax is not uncommon and systems for annotation, automatic terminology extraction and hypernym detection could benefit from being able to process these complex terms and relations.

6. Bibliographical References

Azevedo, C., Iacob, M., Almeida, J., van Sinderen, M., Pires, F., L., and Guizzardi, G. (2015). Modeling Resources and Capabilities in Enterprise Architecture: A Well-founded Ontology-based Proposal for ArchiMate. *Information Systems*, 54:235–262.

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

Caraballo, S. (1999). Automatic Acquisition of a Hypernym-labeled Noun Hierarchy from Text. In *Proceedings of ACL-99*, pages 120–126, Baltimore, MD.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning Semantic Hierarchies via Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 1199–1209, Baltimore, Maryland, USA.

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

Koehn, P. and Knight, K. (2003). Empirical Methods for Compound Splitting. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003)*, pages 187–193, Budapest, Hungary.

Kozareva, Z., Riloff, E., and Hovy, E. (2008). Semantic Class Learning from the Web with Hyponym Pattern Linkage graphs. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1048–1056, Columbus, Ohio, USA.

Lefever, E., Van de Kauter, M., and Hoste, V. (2014a). HypoTerm: Detection of Hypernym Relations between Domain-specific Terms in Dutch and English. *Terminology*, 20(2):250–278.

Lefever, E., Van de Kauter, M., and Hoste, V. (2014b). Evaluation of Automatic Hypernym Extraction from Technical Corpora in English and Dutch. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 490–497, Reykjavik, Iceland.

Lenci, A. and Benotto, G. (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.

Macken, L. and Tezcan, A. (in press). Dutch Compound Splitting for Bilingual Terminology Extraction. In Ruslan Mitkov et al., editor, *Multi-word Units in Machine Translation and Translation Technology*. John Benjamins.

Macken, L., Lefever, E., and Hoste, V. (2013). TExSIS: Bilingual Terminology Extraction from Parallel Corpora using Chunk-based Alignment. *Terminology*, 19(1):1–30.

Malaisé, V., Zweigenbaum, P., and Bachimont, B. (2004). Detecting Semantic Relations between Terms in Definitions. In *the CompuTerm workshop 2004: 3rd International Workshop on Computational Terminology*, pages 55–62.

Mititelu, V. (2008). Hyponymy Patterns. Semi-automatic Extraction, Evaluation and Inter-lingual Comparison. *Text, Speech and Dialogue: Lecture Notes in Computer Science*, 5246:37–44.

Navigli, R. and Ponzetto, S. (2010). BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 216–225, Uppsala, Sweden.

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

Ponzetto, S. and Strube, M. (2011). Taxonomy Induction based on a Collaborative Built Knowledge Repository. *Artificial Intelligence*, 175:1737–1756.

Rei, M. and Briscoe, T. (2014). Looking for Hyponyms in Vector Space. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 68–77, Baltimore, Maryland, USA.

Ritter, A., Soderland, S., and Etzioni, O. (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.

Sparck Jones, K. (1979). Experiments in Relevance Weighting of Search Terms. *Information Processing and Management*, 15:133–144.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a Web-based Tool for NLP-Assisted Text Annotation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL'12)*, pages 102–107, Avignon, France.

Tjong Kim Sang, E., Hofmann, K., and de Rijke, M. (2011). Extraction of Hypernymy Information from Text. In *Interactive Multi-modal Question-Answering. Series: Theory and Applications of Natural Language Processing*, pages 223–245. Springer-Verlag Berlin Heidelberg.

Van de Kauter, M., Coorman, G., Lefever, E., Desmet, B., Macken, L., and Hoste, V. (2013). LeTs Preprocess: The Multilingual LT3 Linguistic Preprocessing Toolkit. *Computational Linguistics in the Netherlands Journal*, 3:103–120.

Van der Plas, L. and Bouma, G. (2005). Automatic Acquisition of Lexico-semantic Knowledge for Question Answering. In *Proceedings of the IJCNLP Workshop on Ontologies and Lexical Resources*, Jeju Island, Korea.

Vintar, S. (2010). Bilingual Term Recognition Revisited. The Bag-of-equivalents Term Alignment Approach. *Terminology*, 16(2):141–158.