



INRIASAC: Simple Hypernym Extraction Methods

Gregory Grefenstette

► To cite this version:

Gregory Grefenstette. INRIASAC: Simple Hypernym Extraction Methods. SemEval 2015, Jun 2015, Denver, United States. hal-01112844v1

HAL Id: hal-01112844

<https://inria.hal.science/hal-01112844v1>

Submitted on 4 Feb 2015 (v1), last revised 6 Jan 2016 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

INRIASAC: Simple Hypernym Extraction Methods

Gregory Grefenstette

Inria

1 rue Honoré d'Estienne d'Orves

91120 Palaiseau, France

Gregory.grefenstette@inria.fr

Abstract

Given a set of terms from a given domain, how can we structure them into a taxonomy without manual intervention? This is the task 17 of SemEval 2015. Here we present our simple taxonomy structuring techniques which, despite their simplicity, ranked first in this 2015 benchmark. We use large quantities of text (English Wikipedia) and simple heuristics such as term overlap and document and sentence co-occurrence to produce hypernym lists. We describe these techniques and present an initial evaluation of results.

1 Introduction

This paper describes the simple hypernym extraction methods implemented in this first participation of Inria in the Semeval campaigns. We participated in task 17 of the 2015 Semeval campaign (Bordea *et al.*, 2015). This task consists in structuring a list of pre-identified domain terms into a list of hypernym pairs. List of terms automatically identified for four domains (equipment; food, chemical, science) were provided by the task organisers. For each domain, two lists were provided, one extracted from WordNet and one from an other source, making eight lists in all. Using any resources, the participants were invited to return eight lists of pairs of terms, in which the first term was a hyponym of the second term. For example, if the words `airship` and `blimp` were included in the list of terms for a domain, the system was expected to return lines such as:

```
25      blimp      airship
```

(where the first number is a meaningless identifier). Given the domain terms lists by the task organizers, we used Wikipedia (downloaded from <http://dumps.wikimedia.org> on August 13, 2014) as our only resource for discovering these relations. From the download source, we only extracted the text of articles, leaving out any categories, infoboxes, or other typed information.

The campaign organizers provided training data from the domains of Artificial Intelligence, vehicles and plants, different from the test domains. The training data consisted in term lists (for plants), and term lists and lists of hypernyms (for AI and for vehicles). We examined these files to get an understanding of the task but did not process them in any way.

2 Domain Lists

We were provided with the following lists of domain terms, with no explanation of how they were created (though WN stands for WordNet):

`chemical.terms:` agarose, nickel sulfate heptahydrate, aminoglycan, pinoquercetin, lupanine, ...

`equipment.terms:` storage equipment, strapping, traveling microscope, minneapolis-moline, ...

`food.terms:` sauce gribiche, botifarra, phitti, food colouring, bean, limequat, kalach, ...

`science.terms:` electro-mechanical systems, biological and physical, history of religions of eastern origins, linguistic anthropology, metaphysics, religion, semantics, ...

`WN_chemical.terms:` abo antibodies, acaricide, acaroid resin, acceptor, acetal, acetaldehyde, acetaldol, acetamide, acetate, acetic acid, ...

`WN_equipment.terms:` acoustic modem, aerator, air search radar, amplifier, anti submarine rocket, apishamore, apparatus, astronomy satellite, atomic pile, audio amplifier, ...

WN_science.terms: abnormal psychology, acoustics, aerology, aeromechanics, aeronautics, ...

2.1 Preprocessing the resource

```
anarch _ _ polit philosophi
_ advoc stateless societi _ defin
_ self-govern voluntari institut _
_ sever author _ defin _
specif institut base _ non-hierarch
free associ
```

```
0 electro-mechan system
1 biolog _ physic
2 histori _ religion _ eastern origin
3 linguist anthropolog
4 metaphys
```

3 Extracting Hypernyms

3.1 Subterms

42	source code	code
2251	theory of inheritance	theory

We did not attempt to generalize the prefix matching to second words of length different from two, and so we missed hypernyms such as `fortimicin b` as a type of `fortimicin` or `ginsenoside c-y` as a type of `ginsenoside`.

Other examples of errors, false positives, caused by these heuristics are `licorice` as a type of `rice` or `surface` to `air missile system` as a type of `surface`, but they are often correct, so any terms in these relations were kept as hypernym pairs without any filtering.

3.2 Sentence and Document Co-occurrence Statistics

Any domain terms produced as possible hyponyms by the prefix or the suffix heuristic were no longer considered. For the remaining terms (which could, of course, include the hypernyms found by the suffix and prefix heuristics), we decided, after trying a number of alternatives described in the next section below, to use the statistics of document presence, and of co-occurrence of terms in sentences to predict hypernym relations.

Let $D_{\text{porter}}(\text{term})$ be the document frequency of a Porter-stemmed term in the stemmed version of Wikipedia. Since Wikipedia article boundaries were stored, we considered each Wikipedia article as a new document.

Let $\text{SentCooc}_{\text{porter}}(\text{term}_i, \text{term}_j)$ be the number of times that the Porter-stemmed versions of term_i and term_j appear in the same sentence in the stemmed English Wikipedia.

Given two terms, term_i and term_j , we decided that if term_i appears in more documents than term_j , then term_i is a candidate hypernym for term_j .

$$\text{CandHypenym}(\text{term}_i) = \{ \text{term}_j : \begin{array}{l} \text{SentCooc}_{\text{porter}}(\text{term}_i, \text{term}_j) > 0 \ \&\& \\ D_{\text{porter}}(\text{term}_j) > D_{\text{porter}}(\text{term}_i) \end{array} \}$$

This heuristically derived set is meant to capture the intuition that general terms are more widely distributed than more specific terms (e.g., `dog` appears in more Wikipedia than `poodle`).

Next define the best candidate for term_i as being the term term_k that appears in the most documents (the most articles in Wikipedia, here):

$$\begin{array}{l} \text{BestHypernym}(\text{term}_i) = \text{term}_k \\ \text{such that} \\ \forall \text{term}_j \in \text{CandHypenym}(\text{term}_i) : \\ D_{\text{porter}}(\text{term}_k) \geq D_{\text{porter}}(\text{term}_j) \end{array}$$

Next, we removed this term term_k from $\text{CandHypenym}(\text{term}_i)$ and repeated the choice twice, retaining, then, the three candidate hypernyms

appearing in the most documents for each term not found by using the prefix or suffix heuristics.

<i>Domain</i>	<i>suffix</i>	<i>prefix</i>	<i>cooc</i>	<i>Total hypernyms produced</i>
WN_chemical	750	10	3766	4001
WN_equipment	171	3	1338	1369
WN_food	616	25	4121	4238
WN_science	174	0	1070	1102
chemical	10780	91	19322	28443
equipment	241	17	1126	1168
food	471	33	4277	4363
science	193	17	1130	1164

Table 1. Number of prefix and suffix hypernyms produced, compared to the total number of hypernyms returned for each domain. Suffix and prefix subterms account for 10% to 36% of the hypernyms we produced. The cooccurrence technique produced the most hypernym candidates.

3.2.1 Co-occurrence Example

Consider the following example. In the domain file `science.terms` there is the term `biblical studies`. The Porter-stemmed version of this term `biblic studi` appears in 887 sentences. Considering all the other terms in `science.terms`, we find that `biblic studi` appears 215 times in the same sentence as the stemmed version of `theology` (`theologi`), 111 times in the same sentences as stemmed `history` (`histori`), 50 times with `religion`, 43 times with `music`, and 42 times with `science` (`scienc`).

```
215 887 21977  biblic studi  theologi
111 887 383927 biblic studi  histori
50  887 64044  biblic studi  religion
43  887 412791 biblic studi  music
42  887 224983 biblic studi  scienc
```

We decided to keep the top three for simplicity, so this term contributed three lines to our submitted `science.taxo` file:

```
121 biblical studies    history
122 biblical studies    religion
123 biblical studies    theology
```

3.3 Other Attempts at Finding Relations

We tried a number of other methods to find hypernyms, none of which gave satisfaction by looking at the results. We implemented a method to recog-

nize sentences containing Hearst patterns (list from (Cimiano *et al.*, 2005)) involving the domain terms. For example, *tape* is in *equipment.terms*, and were able to find stemmed sentences of the form *A, B and other C ... such as today , sticki note , 3m #tape# @, and other@ #tape# ar exampl of psa (pressure-sensit adhes)* from which we should have been able to extract relations such as *3m tape is a type of tape*, and *sticky note is a type of tape*. But we would have had to parse the sentence, and been willing to add new terms (which was permitted by the organizers, to the derived hypernym lists) but in our first participation in Semeval, we did not want to make that processing investment yet.

We tried to discover the *basic vocabulary* (Kit, 2002) of each domain by counting the number of times that each term appeared in Wikipedia in the set phrase *A, such as*. For example, using all the terms from *equipment.terms*, we found

225 instances of *equipment*, such as
 24 instances of *internet*, such as
 4 instances of *telescop*, such as
 2 instances of *manual*, such as
 2 instances of *manipul*, such as

But this did not seem very useful or productive.

4 Evaluation

Each participant in Task 17 of SemEval 2015 was allowed to submit one run for each of the 8 domains (see Table 1 for the names of the domains, and the number of hypernym pairs we submitted). The task organizers evaluated the submissions of the six participating teams, using automated and manual methods, and published their evaluation three weeks after the submission deadline. Our team placed first in the official ranking of the six teams.

The evaluation criteria, which were not published before the submission, combined the presences of cycles in the hypernyms submitted, the Fowlkes & Mallows measure of the overlap between the submitted, the F-score ranking, the number of domains submitted (not all teams returned results for all domains), and a manual precision ranking (for hypernyms not present in the gold standard). The gold standards used by the task organizers came

from published taxonomies, or from subtrees of WordNet (prefixed as *WN_* above). Here is a quick evaluation of how well our simple hypernym extraction techniques fared on each gold standard in Table 2.

Domain	suffix	prefix	cooc	union	gold to find
WN_chemical	377	5	574	644	1387
WN_equipment	119	0	168	184	485
WN_food	371	2	681	726	1533
WN_science	119	0	230	240	441
chemical	2019	9	715	2407	24817
equipment	184	1	286	305	615
food	279	1	807	822	1587
science	121	7	193	209	465

Table 2. Number of gold standard relations to find in the last column. Columns 2, 3 and 4 are the number of gold standard relations found by each technique. “union” is the union of columns 2, 3 and 4. Since the co-occurrence technique can find relations that have been found by the suffix and prefix techniques.

Domain	suffix	prefix	cooc	union	gold to find
WN_chemical	26%	0.3%	40%	46%	1387
WN_equipment	24%	0%	34%	38%	485
WN_food	23%	0.1%	43%	47%	1533
WN_science	26%	0%	51%	54%	441
chemical	8%	0.03%	3%	10%	24817
equipment	30%	0.02%	47%	50%	615
food	18%	0.06%	51%	52%	1587
science	26%	1.8%	42%	45%	465

Table 3. Percentage of correct answers found by each method.

As Table 3 shows, most of the correct answers found come from the sentence and document co-occurrence method described in section 4.2.

5 Conclusion

Even though training data was provided for this taxonomy creation task, we did not exploit it in this our first participation in Semeval. We implemented some simple frequency-based co-occurrence statistics, and substring inclusion heuristics to propose a set of hypernyms. We did not implement any graph algorithms (cycle detection, branch deletion) that would be useful to build a true hierarchy. We hope to learn from interaction from the other participants what paths to explore in the future to improve recall.

Acknowledgments

This research is partially funded by a research grant from INRIA, and from the Paris-Saclay Institut de la Société Numérique funded by the IDEX Paris-Saclay, ANR-11-IDEX-0003-02.

References

- Georgeta Bordea, Paul Buitelaar, Stefano Faralli, and Roberto Navigli. 2015. Semeval-2015 task 17: Taxonomy Extraction Evaluation. *Proceedings of the 9th International Workshop on Semantic Evaluation*. Association for Computational Linguistics
- Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. 1995. Automatic query expansion using SMART: TREC3. *Proceedings of the 3rd Text REtrieval Conference (TREC-3)*. NIST Special Publication 500-226. National Institute of Standards and Technology (NIST), Gaithersburg, MD, pp: 69–80.
- Philipp Cimiano, Aleksander Pivk, Lars Schmidt-Thieme, and Steffen Staab. 2005. Learning taxonomic relations from heterogeneous sources of evidence. *Ontology Learning from Text: Methods, evaluation and applications*. IOS Press.
- Gregory Grefenstette. 1999. Tokenization. *Syntactic Wordclass Tagging*. Springer Netherlands, pp. 117-133.
- Chunyu Kit. 2002. Corpus tools for retrieving and deriving termhood evidence. *Proceedings of the 5th East Asia Forum of Terminology*, pp. 69-80.
- Peter Willett. 2006. The Porter stemming algorithm: then and now. *Program* 40(3): 219-223.