An Unsupervised Method for General Named Entity Recognition and Automated Concept Discovery

Enrique Alfonseca and *Suresh Manandhar*

Abstract

Knowledge Acquisition is still the bottleneck in building many kinds of applications, such as inference engines. We describe here a procedure to automatically extend an ontology with domain-specific knowledge. The main advantage of our approach is that it is completely unsupervised, so it can be applied to different languages and domains. Our initial results have been highly successful and we believe that with some improvement in accuracy it can be applied to large ontologies.

1   Introduction

There are several general-purpose ontologies available for English and other languages, such as WordNet [Miller, 1995], Comlex [Macleod and Grishman, 1994, Mifflin et al., 1994], and EuroWordNet [Vossen, 1998]. However, extending them with domain-dependent information is still a labour-intensive task that requires a high degree of human supervision. Knowledge acquisition is today a bottleneck for construction of inference and expert systems, and there is a need for an automatic acquisition methodology. This paper presents an approach to enriching ontologies with domain-dependent information in a fully unsupervised way. For that aim, we have put together ideas from different fields in Natural Language Processing, such as *named entity recognition*, *knowledge acquisition*, and procedures used in *word sense disambiguation*, that we believe may be useful for solving the problem we have in hands.

We describe here a procedure to extend ontologies with domain-dependent information. The only input it requires is a collection of documents collected for one domain. In our preliminary experiments, with absolutely no human supervision, the new synsets from the texts are correctly placed in the ontology we have used.

1.1   Related work

Lexical repositories are a very useful resource for Natural Language Processing, and the availability of a few of them such as WordNet [Miller, 1995], Comlex [Macleod and Grishman, 1994] or Cyc [Lenat and Guha, 1990, Lenat, 1995] has made possible many successful applications. There are now automatic procedures to port WordNet to other languages such as Catalan [Daudé et al., 2000] and Korean [Lee et al., 2000], but it is still difficult to find good automatic methods to learn ontologies about specific domains.

Concerning Lexical Knowledge Acquisition from dictionaries and other semi-structured texts, it has been already attempted with good results, using certain patterns in the definitions to identify the relations among synsets. To cite a few works, Wilks et al. [1996], Grefenstette [1994] and Rigau [1998] extracted WordNet-like ontologies from dictionary definitions. However, to our knowledge, there is still no procedure to enrich WordNet with domain-specific information from free texts that does not rely on human intervention in some way or other. O'Sullivan et al. [1995] extended WordNet with a domain-specific ontology, but all the work was done by hand by domain experts. Other systems have a higher degree of automaticity but all of them depend on a human expert in some degree to classify the learnt synsets.

In the Asium system [Faure and Nédellec, 1998], a clustering algorithm is used to create a concept taxonomy. Maedche and Staab [2001] also describe a general architecture for acquiring ontologies and relationships directly from free texts, semi-structured texts (such as dictionaries) and data bases. While conceptual clustering is useful for grouping the concepts identified from a text, its application for extending already existing ontologies such as WordNet is not so straightforward. Furthermore, each time two concepts are clustered, a new superconcept is created as hypernym of them, which might have

no counterpart in the language. Other works, such as [Maedche and Staab, 2000], focus on learning non-taxonomical relations.

Kietz et al. [2000] describes a procedure to adapt WordNet to specific domains, by removing the non-relevant synsets and adding the domain-specific ones. When enriching WordNet with new synsets to the ontology, the system produces suggestions that have to be supervised by a human.

1.2 Structure of this document

Next section describes the task we want to achieve ultimately. Section 3 shows some word-sense disambiguation techniques that we applied to our work, and section 4 introduces our algorithm. Finally, we present our results and conclusions in sections 5 and 6.

2 General Named Entity Recognition

Let's suppose that we have an ontology with three components $O =< C, I, h >$ where

- C is a set of concepts (e.g. *human*).

- I is a set of instances of concepts (e.g. *Shakespeare*).

- h is the hypernymy function $h : C \cup I \to C$ that establishes a taxonomy of concepts and instances.

An example of such function h is the one defined as $h(c_1) = c_2$ iff the concept c_1 is a specialisation of the concept c_2, and it reads either c_1 *is a hyponym of* c_2 or c_2 *is a hypernym of* c_1.

Note that all members of I must be leaves in the taxonomy, but not all leaves are instances; some of them can represent concepts that have no instances in the hierarchy.

2.1 Task definition

General Named Entity Recognition is the task of identifying, for an unknown concept or instance u, the correct concept $c \epsilon C$ such that $h(u) = c$, i.e. it consists in finding the most accurate immediate generalisation of u in the known hierarchy of concepts.

For example, if we are processing a text about Tolkien's mythology, and we find the unknown concept *hobbit*, an accurate General Named Entity Recogniser would have it attached to the most accurate hypernym, which is, in WordNet 1.7, *fairy*, and it would be brother of the existing concepts *elf, dwarf*, etc.

2.2 Relation to Named Entity Recognition

Named Entity Recognition is the task of finding and classifying objects that are of interest to us. These objects can be people, organisations, locations, dates, or anything that is useful to solve a particular problem. For instance, in the following sentences, taken from the Wall Street Journal corpus in the Penn Treebank [Marcus et al., 1993], we can find two references to a person, one date and one organisation.

> [$_{person}$ Pierre Vinken], 61 years old, will join the board as a nonexecutive director [$_{date}$ Nov. 29]. [$_{person}$ Mr. Vinken] is chairman of [$_{organisation}$ Elsevier N.V.], the Dutch publishing group.

We consider Named Entity Recognition as a more restricted task, where the hierarchy is flat and it only contains a few concepts, e.g. person, organisation and location. On the other hand, we are considering a taxonomy of concepts organised in a more sophisticated ontology, which can have just subtle differences between them.

2.3 Relation to Word-Sense Disambiguation

In Word Sense Disambiguation, we have a word in a text, and the task is to decide which is the correct meaning of that word. For example, using WordNet this task involves deciding that out of the 10 senses of *bank*, in sentence (1a) it refers to the *slope beside a body of water*; and in (1b) it refers to a *financial institution*.

(1)   a. The boy played beside the river bank.

Word	Freq	w_1	w_2	Word	Freq	w_1	w_2	Word	Freq	w_1	w_2
1	1677	1.13	2.10	Dwarves	106	1.21	2.37	Doom	61	1.17	2.24
by	1124	0.48	0.61	flowers	97	1.01	1.75	yellow	61	1.14	2.15
2	658	0.91	1.49	Races	94	1.21	2.37	pink	61	1.17	2.24
killed	645	1.16	2.21	fairy	87	1.22	2.40	Barbarian	60	1.22	2.40
he	591	0.02	0.02	giant	84	1.11	2.04	Deep	58	1.20	2.35
146	307	1.17	2.24	Killed	84	1.17	2.25	Dungeons	58	1.22	2.40
145	230	1.21	2.37	Halfling	80	1.22	2.40	obtusa	57	1.22	2.40
Human	218	1.18	2.28	Cham.	76	1.22	2.40	Mixed	56	1.20	2.34
9	213	1.01	1.76	dwarves	75	1.04	1.84	Warrior	55	1.13	2.12
Elf	212	1.13	2.10	Pink	75	1.13	2.11	king	54	1.05	1.87
Gnome	150	1.19	2.31	Fairy	70	1.22	2.40	races	53	1.22	2.40
gnome	138	1.21	2.38	Cleric	61	1.22	2.40	kB.	53	1.22	2.40

Table 1: Some top words in the signature of the Dwarf. The second column is the frequency count, and the third column is the weight of the word, using Yarowsky's function (w_1) and Agirre's function (w_2).

> b. I have opened an account in a bank.

Again, General Named Entity Recognition can be considered as a more general task than Word Sense Disambiguation, where we have to find the synset whose meaning is the most similar among all the concepts in the whole taxonomy, instead than just the synsets containing that lexical word.

General Named Entity Recognition is a task that covers, and is harder than both Named Entity Recognition and Word Sense Disambiguation.

3 Word-sense disambiguation procedures

A topic signature of a word w is the list of the words that co-occur with it, together with their respective frequencies or weights. It is a tool that has been applied to word-sense disambiguation with promising results [Yarowsky, 1992] [Agirre et al., 2000]. Because WordNet does not include topic signatures, we have used the method invented by Agirre et al. [2000] for collecting them, in an unsupervised way, from Internet. We will include here a brief description of the procedure. Except for some minor changes, the work described in this section has been done before.

Agirre's method consists of the following steps. For every WordNet synset s_i,

1. Generate a query containing all the words in s_i and its hyponyms as positive keywords, and the words in other synsets that contain the same words as negative keywords.

2. Submit the query to an Internet search engine, and collect the results.

3. Download the documents, look for the synset words in them, and calculate the frequencies of the words that occur around them, in a context of width w.

4. Store the list of words and frequencies, l_i, excluding the most common closed-class words (determiners, pronouns, conjunctions, etc).

The following is an example of the WordNet synset for *country* (sense 06621523 in WordNet 1.7).

> state, nation, country, land, commonwealth, res publica, body politic –(a politically organized body of people under a single government; "the state has elected a new president"; "African nations"; "students who had come to the nation's capitol"; "the country's largest manufacturer"; "an industrialized land")

The query that was produced is the following:

> "country" AND ("body politic" OR "commonwealth" OR "land" OR "nation" OR "res publica" OR "state" OR "Reich" OR "suzerain" OR "sea power" OR "great power" OR "major power" OR "power" OR "superpower" OR "world power" OR "city state" OR "ally") AND NOT ("a people" OR "area" OR "rural area")

Next, the raw frequencies are changed into weights. The reason is that some words are equally frequent in all document collections, so they do not provide contextual support and can be ruled out. Furthermore, some document collections may be bigger than others, so a normalisation is required to give the same overall weight to all signatures.

For every list of word frequencies l_i,

3

attach(u, C)
u is the unknown synset,
C is a collection of domain-specific documents.
1. Calculate l_u, the list of frequencies of words co-occurring with u, using the documents in C.
2. Let r be the root synset in the ontology.
3. return analyseLevel(l_u, r)

analyseLevel(l_u, c)
l_u is the unknown synset's list of word frequencies,
c is the candidate synset most similar to u.

1. Get c's synset children, $\{c_1, c_2, ..., c_n\}$.
2. $t_c \leftarrow c$'s topic signature
3. $\{t_{c_1}, t_{c_2}, ..., t_{c_n}\} \leftarrow c$'s children's topic signatures.
4. Find the signature which is more similar to l_u
      4.1 If that signature is t_c, return c
      4.2 Let t_{c_i} be the signature that scored better.
      4.3 return analyseLevel(l_u, c_i)

Figure 1: Pseudocode of the algorithm for finding the correct place where the unknown synset u will be attached in the ontology

- Transform the word frequencies into weights, and produce the topic signature t_i.

In our current work, we have used two weight functions, both of which have been already used for word-sense disambiguation.

3.1 First weight function

[Yarowsky, 1992]'s weight function is computed as follows: let's suppose that we have several lists of word frequencies $\{l_1, ..., l_n\}$, counted from document collections that contain, respectively, the words in synsets $\{s_1, ..., s_n\}$. Then, the weight for each word is given by equation 1.

$$\log \frac{P(w|s_i) \cdot P(s_i)}{P(w)} \tag{1}$$

where $P(w)$ is the overall probability of a word; $P(w|s_i)$ is the probability of w given that it is in the context of a synset s_i; and $P(s_i)$ is the probability that a word is in the context of s_i. The first two probabilities are estimated from the document collections, and the third one is assumed to be uniform.

3.2 Second weight function

The second weight function we have tested is the same that Agirre et al. [2000] used in their word-sense disambiguation experiments. If w_j is a word, and $freq_{i,j}$ is its frequency in the frequency list l_i, then its expected mean $m_{i,j}$ is defined as

$$m_{i,j} = \frac{\sum_i freq_{i,j} \cdot \sum_j freq_{i,j}}{\sum_{i,j} freq_{i,j}} \tag{2}$$

The weight for synset s_j in the topic signature t_i is then

$$w_{i,j} = \begin{cases} \frac{(freq_{i,j} - m_{i,j})}{m_{i,j}}, & \text{if } freq_{i,j} > m_{i,j} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

3.3 Discussion

Both functions have the desirable property that the weight associated to each word is a measure of the support that a word provides that we are in the context of a certain WordNet synset, regardless of the actual frequency values. So, if two words have appeared in the context of a synset with different

frequencies, but neither of them appears in the context of any other synset, they both will score the maximum value of the weight, because they both are maximally supporting that synset.

In our experiments, although the word weights and the similarity metrics are slightly different using both functions, they always produced the same synset classifications.

Table 1 shows the signatures corresponding to the synset *dwarf,* and the weight values with both functions.

4   Augmenting an ontology

We use the notion that words semantically related must co-occur with the same kinds of words [Maedche and Staab, 2000]. In the same way that word co-occurrence information is useful for word-sense disambiguation [Yarowsky, 1992] [Agirre et al., 2000], they can also be useful to calculate a degree of similarity between concepts and, therefore, to decide which concept in an ontology is the most similar to a new unknown concept u. All the work described from this section on is original.

The procedure is detailed in Figure 1. It is a top down search starting at the most general concept in the taxonomy, which tries to find the concept whose topic signature is closest to the target concept.

Let's suppose that we have a domain-specific collection of documents and we find references to a concept u that is not present in our ontology. First, we compile the list l_u of words that co-occur with that concept in the sample documents, and compute their frequencies. Next, at each level of the ontology, we find the concept whose topic signature is the most similar to l_u. We may stop at the middle of a hierarchy if a concept's signature scores higher than all of his children concepts' signatures.

4.1   Similarity metric

Let t_i be the topic signature of a concept, and l_u be the list of frequencies of co-occurring words for the unknown concept.

$$t_i = \{< w_1, w_{i1} >, ..., < w_n, w_{in}) >\}$$
$$l_u = \{< w_1, f_1 >, ..., < w_n, f_n >\}$$

where w_j is the jth word in the list, w_{ij} is its weight in the topic signature t_i, and f_j is the frequency count in the contexts of u in the collection of domain-specific documents.

Then, the similarity function we have used is the dot product of both vectors [Yarowsky, 1992]:

$$Similarity(t_i, l_u) = \sum_{j=0}^{n} w_{ij} \cdot f_j \tag{4}$$

Therefore, to find the concept that is most similar to the unknown concept u (step 4 in the algorithm) we have to find

$$\mathrm{argmax}_i Similarity(t_i, l_u) \tag{5}$$

4.2   Adapting WordNet to the problem

Before this procedure can be applied to WordNet, two changes are desirable. The first and more important one is its enrichment with topic signatures. The procedure to do it is completely automatic and needs no human supervision, but we have not been able to finish it because downloading all the documents is rather slow, and there was not enough time for changing WordNet.

Secondly, it would be desirable that each synset had a flag indicating whether it represents an instance or a concept. Because instances (e.g. *Shakespeare*) cannot have hyponyms, if they were marked the search space for our algorithm would be smaller. We describe a way to do this annotation in [Alfonseca and Manandhar, 2002].

5   Preliminary experiments

In the beginning, we tested our algorithm on small ontology, displayed here in Figure 2.

The domain-specific documents we used consisted in an electronic version of *The Lord of the Rings* [Tolkien, 1968], which contains roughly 478,000 words. We chose the unknown concept *hobbit* and the unknown instance *Mordor,* both appearing in the text.
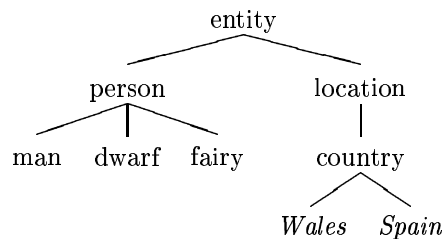
```
                              entity
                          ____/    \____
                      person          location
                   __/  |  \__            |
                 man  dwarf  fairy      country
                                        /    \
                                    Wales    Spain
```

Figure 2: Initial ontology in the preliminary experiments

.

```
              entity                              entity
          ___/      \___                      ___/      \___
      person          location            person          location
      sim=285         sim=250             sim=329          sim=373
    __/  |  \__          |                   |                |
  man  dwarf  fairy     ...                 ...            country
  sim=642 sim=555 sim=340                                  sim=?
                                                           /    \
                                                       Wales    Spain
                                                       sim=592  sim=253
```
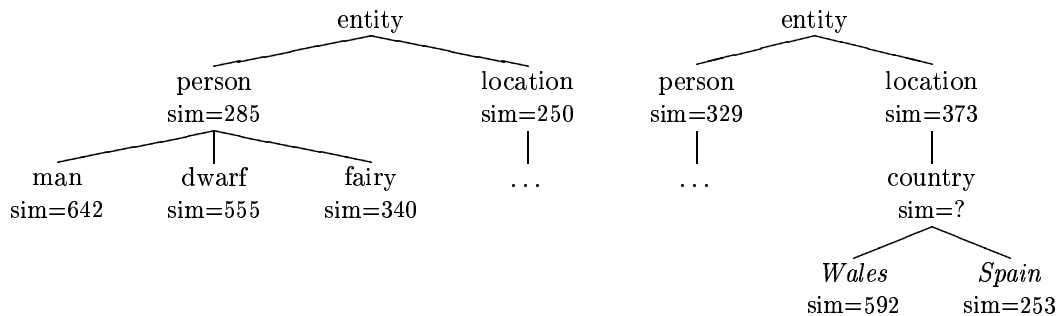
Figure 3: Decisions taken from classifying *Hobbit* (left) and *Mordor* (right).

Figure 3 shows the value of the similarity function, at each level, for classifying *hobbit* and *Mordor*. The first one was finally attached to *man*, although there was a strong evidence as well pointing to *dwarf* as a possible hypernym. Concerning *Mordor*, because *Wales* and *Spain* are instances, it is finally attached to *country*. Note as well that the algorithm did not calculate the similarity with *country*, because *Mordor* had been identified as being an instance, using evidence from the texts, and therefore the algorithm just proceeded downwards so as to leave it as a leaf in the hierarchy.

The algorithm performed well with the small hierarchy that we produced for our experiments. However, the first discrimination, that of deciding whether it was a *person* or a *location* was always a very narrow one. Because WordNet is a much more complex network, and there are nodes with hundreds of children (e.g. *person*), this first approach would probably need some fine-tuning and adjusting before it works properly.

6   Final experiments

The main problem we identified in the preliminary experiments was that *person* and *location* are far too general concepts, and their topic descriptions also contained many general terms, that were usually not very representative of the sub-concepts located below them. Therefore, we made another study where, for calculating the topic description of a concept, all the frequency lists of its sub-concepts were added up as well. For instance, to calculate the topic description of *person*, first the program added to its list of frequencies those of its sub-concepts *dwarf, fairy* and *man*. This produced much better results.

Figure 5 shows the ontology used in our last experiments. It is slightly more complex than the previous one. The concepts we extracted from the domain-specific texts are *hobbit, wizard, Mordor, eagle* and *horse*.

We performed several experiments, by varying the size of the context from which the word frequencies were calculated. As expected, the bigger the context, more words are used for the topic signature, which is more complete, but that also introduces noise words and makes more difficult the classification. Table 2 shows the classification of these new concepts in three different settings. In the first column, frequencies for the topic signatures were collected from the paragraphs that contained the concept (e.g. city, man, etc.) and frequencies for the domain-dependent concepts were collected from the sentences containing them. In the second column, both frequencies lists were taken using as context the sentences. In the third column, the frequencies for the domain-dependent concepts were collected using a context of five
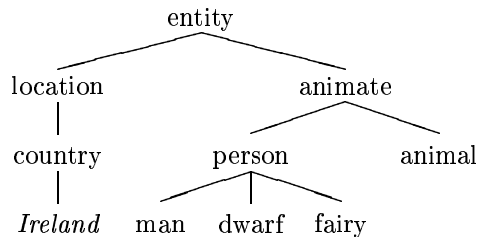
6

```
                              entity
                    ┌───────────┴───────────┐
                location                  animate
                    │              ┌──────────┴──────────┐
                country         person                 animal
                    │        ┌─────┼─────┐
                 Ireland    man  dwarf  fairy
```

Figure 4: Ontology used in the final experiments.

Setting	Par/Sent	Sent/Sent	Sent/5wds
Mordor	location	*man	location
Hobbiton	location	*animal	location
Isengard	location	*man	location
hobbit	*animal	man	man
wizard	*animal	man	man
eagle	animal	animal	animal
horse	animal	*man	*man
Accuracy	71%	43%	86%

Table 2: Classification of concepts *hobbit, Mordor, wizard, horse* and *eagle* with different context sizes. Wrong classifications are marked with an asterisk. Not only the last approach gave more correct classifications, but the similarity function gave more support to its decisions.

words at each side. In this last setting, not only more concepts were correctly classified, but also *the similarity measure*, at each decision step, *supported with more strength the correct decisions*.

The resulting taxonomy was that of Figure 5. The only concept that was misclassified was *horse*. It is the case that the list of words and frequencies colected for horse contains words that also appear in the context of people, such as colors, verbs of movement, and some adjectives.

Finally, Figure 6 shows the similarity values when locating the proper place for the concept *wizard*.

7 Discussion and Conclusions

We have presented here an algorithm that is, to our knowledge, the only fully unsupervised method to extend an ontology with unknown concepts taken from domain-specific documents. It can be applied to different languages and domains as it is. We believe that it will be able to tackle big ontologies once we have collected enough data in the topic signatures, and we have experimented more similarity functions and statistical models. It is highly versatile, and it allows the attachment of new concepts to any intermediate level in an ontology, not only at the leaves.

We have experimented several contexts for extracting word frequencies, and the conclusion is that it is better, in this task, to consider a small context of highly related words than a big context that includes
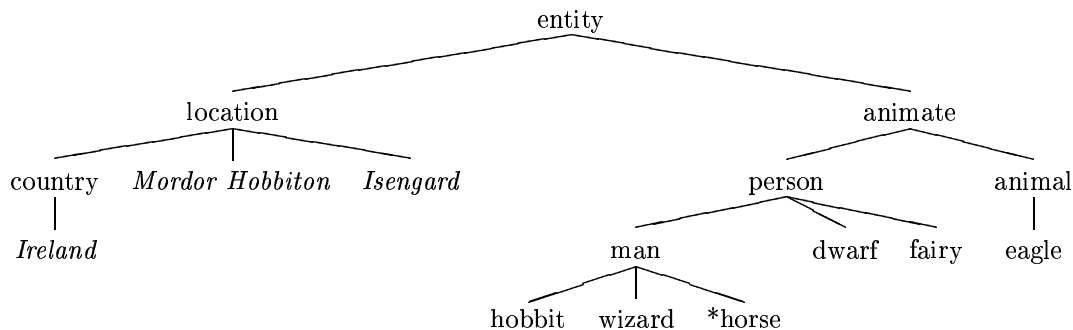
```
                                    entity
                    ┌─────────────────┴─────────────────┐
                 location                             animate
          ┌────────┬───┴────┬─────────┐          ┌──────┴──────┐
       country  Mordor  Hobbiton  Isengard     person        animal
          │                                  ┌────┴──────┐       │
       Ireland                             man     dwarf fairy  eagle
                                        ┌───┼───┐
                                     hobbit wizard *horse
```

Figure 5: Resulting ontology after the best experiments.

7

```
                                         entity
                         _____/        _____
                        /                                      \
              location                                        animate
              sim=8,44                                        sim=37,32
                  |                            _____/   _____
                  |                           /                      \            \
                 ...                       person                   animal        stop
                                          sim=27,34                 sim=21,46    sim=0
                              _____/   |   _____
                             /               |               \
                          man             dwarf             fairy          stop
                        sim=83,84        sim=31,48        sim=81,95       sim=63,27
```
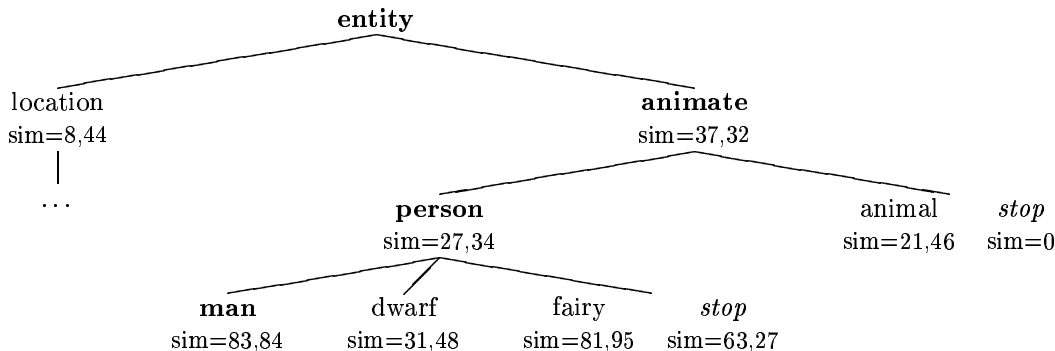
Figure 6: Values for classifying the new concept *wizard*. The similarity labelled as *stop* corresponds to the decision of stopping at that level in the tree, and attaching the new concept to the parent node (e.g. animate or person)

more words. This is in contrast to Agirre et al. [2000], who used a context window of 100 words.

In theory, this algorithm can also be used to create a new ontology from scratch. In this case, however, we must be careful that the concepts are learnt from the most general to the most specific one, because once a concept is attached to the hierarchy it is not possible to move it from its position.

This work can also be used to test the degree of adequacy between existing ontologies, such as WordNet, and the usage of concepts in language. For instance, *fairy* and *dwarf* are considered, in WordNet, hyponyms of the concept *psychological feature*, and thence they are located far away from *animate being*; but they are always used in language in the same way as animated beings, in the sense that they are usually selected by the same verbs and have similar complements.

7.1 Improvements and future lines

The following are only a few of the possible improvements that we may try with this procedure:
- Try different similarity and weight functions. Combine them with other semantic distance metrics.
- Try other features to measure similarity, or look for natural-language expressions that denote hyponymy.
- Produce better topic signatures, using a bigger set of documents.
- Use a beam search, looking for several candidate hypernyms at the same time, and finally decide between them.
- Identify, from the evidence in the domain-specific texts, whether we are looking for an instance or a concept. If it is an individual, the search can be simplified because we know it can only be a leaf.

8 Acknowledgements

9 Authors affiliation

Enrique Alfonseca is an assistant lecturer at the Computer Science Department, Universidad Autónoma de Madrid, and a part-time research student at the University of York. Suresh Manandhar is a lecturer at the Computer Science Department, University of York.

Contact: {enrique, suresh}@cs.york.ac.uk

References

E. Agirre, O. Ansa, E. Hovy, and D. Martinez. Enriching very large ontologies using the www. In *In Proceedings of the Ontology Learning Workshop, ECAI*, Berlin, Germany, 2000.

E. Alfonseca and S. Manandhar. Distinguishing instances and concepts in wordnet. In *Poceedings of the First International Conference on General WordNet*, Mysore, India, 2002.

J. Daudé, L. Padró, and G. Rigau. Mapping wordnets using structural information. In *38th Anual Meeting of the Association for Computational Linguistics (ACL'2000)*, Hong Kong, 2000.

D. Faure and C. Nédellec. A corpus-based conceptual clustering method for verb frames and ontology acquisition. In *LREC workshop on Adapting lexical and corpus resources to sublanguages and applications*, Granada, Spain, 1998.

G. Grefenstette. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, 1994.

J. Kietz, A. Maedche, and R. Volz. A method for semi-automatic ontology acquisition from a corporate intranet. In *Workshop "Ontologies and text", co-located with EKAW'2000*, Juan-les-Pins, French Riviera, 2000.

Changki Lee, Geunbae Lee, and Seo Jung Yun. Automatic wordnet mapping using word sense disambiguation. In *38th Anual Meeting of the Association for Computational Linguistics (ACL'2000)*, Hong Kong, 2000.

D. Lenat. *Steps to Sharing Knowledge*. Mars N., editor, Towards Very Large Knowledge Bases. IOS Press, 1995.

D. B. Lenat and R. V. Guha. *Building Large Knowledge-Based Systems*. Addison-Wesley, Reading (MA), USA, 1990.

C. Macleod and R. Grishman. Comlex syntax reference manual, 1994.

A. Maedche and S. Staab. Discovering conceptual relations from text, 2000.

A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent systems*, 16(2), 2001.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of english: the penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

H. Mifflin, M.A. Boston, R. Grishman, Catherine Macleod, and Adam Meyers. Comlex syntax: Building a computational lexicon. In *Proceedings of COLING-94*, Kyoto, Japan, 1994.

George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

D. O'Sullivan, A. McElligott, and R. F. E. Sutcliffe. Augmenting the princeton wordnet with a domain specific ontology. In *Proceedings of the Workshop on Basic Issues in Knowledge Sharing at the 14th International Joint Conference on Artificial Intelligence*, Montreal, 1995.

G. Rigau. *Automatic Acquisition of Lexical Knowledge from MRDs*. PhD Thesis, Departament de Llenguatges i Sistemes Informàtics.– Universitat Politècnica de Catalunya. – Barcelona, 1998.

J. R. R. Tolkien. *The Lord of the Rings*. Allen and Unwin, 1968.

P. Vossen. *EuroWordNet - A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, 1998.

Y. A. Wilks, B. M. Slator, and L. M. Guthrie. *Electric words: Dictionaries, computers and meanings*. Cambridge, MA: MIT Press, 1996.

David Yarowsky. Word-sense disambiguation using statistical models of roget's categories trained on large corpora. In *Proceedings of COLING-92*, pages 454–460, Nantes, France, 1992.