

# The Role of Conceptual Relations in Word Sense Disambiguation

David Fernández-Amorós, Julio Gonzalo, Felisa Verdejo  
Depto. de Lenguajes y Sistemas Informáticos, UNED  
{david,julio,felisa}@lsi.uned.es

## Abstract

We explore many ways of using conceptual distance measures in Word Sense Disambiguation, starting with the Agirre-Rigau conceptual density measure. We use a generalized form of this measure, introducing many (parameterized) refinements and performing an exhaustive evaluation of all meaningful combinations. We finally obtain a 42% improvement over the original algorithm, and show that measures of conceptual distance are not worse indicators for sense disambiguation than measures based on word-cooccurrence (exemplified by the Lesk algorithm). Our results, however, reinforce the idea that only a combination of different sources of knowledge might eventually lead to accurate word sense disambiguation.

## 1 Introduction

Competitive Word Sense Disambiguation (WSD) performance, as illustrated by participants in the first Senseval competition [KR00], can only be reached mixing all kinds of knowledge: co-occurrence information, syntactic information and collocations, additional information from dictionaries such as domain labels, selectional restrictions, and all kinds of heuristics (see for instance [NL96, GRA97, WS98, SW99]). A problem with such hybrid systems is that they make hard to discern what is the discriminative power of each of the different types of knowledge about the context of the word to be disambiguate. Our belief is that a separate, detailed study of each knowledge source is a necessary step to understand WSD challenges.

In this paper, we focus on conceptual relations as a source of information for WSD systems. The basic hypothesis is that the right senses for the words in a natural language expression will have closer semantically relations (in a semantic network) than incorrect combinations of senses. For instance, in “*Spring is my favorite season*”, the *springtime* sense of *spring* has a hyponymy (IS-A) relation with the *season of the year* meaning of *season*, while any other combination of senses (e.g. *spring* as fountain and *season* as sports season) have weaker semantic relationships.

Our aim is to perform an in-depth study (via exhaustive empirical evaluation) of the role that conceptual relations may play in accurate WSD. As a point of departure, we chose one of the most promising WSD methods based solely on conceptual relations, the Agirre-Rigau algorithm, based on a measure of conceptual density [AR96]. As in their work, we

have used the WordNet [Fel98] semantic network as the lexical database providing word senses and semantic relations between them. Wordnet includes around 168000 English word senses, and has also large-scale versions for many other languages [Vos98].

Then we generalized the original algorithm, parameterizing many aspects of the original system, including the conceptual density formula itself. The strategies incorporated to the algorithm include as much possibilities to exploit semantic relations as we could think of. Finally, we performed an exhaustive evaluation, running the system in more than 50 different configurations against all nouns in the Semcor test collection, the largest semantically annotated test collection known to us (even the original algorithm had not been previously tested against the whole Semcor collection).

In Section 2, the main algorithm and all the variants are explained. Section 3 describes the evaluation performed and the results obtained. Finally, Section 4 describes the main conclusions.

## 2 Description of the algorithm

The basic elements for the algorithm are a Lexical Knowledge Base (LKB) with conceptual information (such as Wordnet *synsets*, or sets of synonym terms), a binary relation  $R$  (usually the hypernymy relation, equivalent to the IS-A relation in an ontology) between the concepts in the LKB and a conceptual density formula (see below) giving the conceptual density of a concept with a certain amount of *activated* (with respect to  $R$ ) subconcepts.

To disambiguate a word we do the following: first, we take the surrounding text and form a window with a given fixed radius. Then we rank the senses of the central word following these steps:

- We look up the senses of all words in the window. For every sense of every word, we take a number of related concepts via the  $R$  relation, and we weight them according to some formula.
- For each sense of the central word, the concept (related to the sense via transitive application of  $R$ ) that has a highest conceptual density gives the conceptual density of the sense.
- Then we normalize the ranks for the senses of the word, and take the resulting values as output of the algorithm.

These steps define a conceptual density algorithm “template” with a wide range of possibilities. In the next section we discuss the parameters we have considered and the values we have tested.

### 2.1 Parameters

**Transitive relation  $R$ .** The most obvious is perhaps the hypernymy relation, but we have also considered the union of semantic relations such as hypernymy and meronymy (“PART-OF” relation).

**Conceptual density measure.** We have tested four different conceptual density measures:

1. The original Agirre-Rigau conceptual density formula [AR96]:

$$CD(c, m) = \frac{\sum_{i=0}^{m-1} adesc^i}{\sum_{i=0}^{h-1} adesc^i} \quad (1)$$

where  $adesc$  is the average number of descendants of concept  $c$  according to  $R^{*-1}$ ,  $\alpha = 0.2$ ,  $m$  is the number of marks in the subhierarchy of  $c$ . And  $h$  is the depth of the subhierarchy under  $c$ . We have called this formula Strict Agirre-Rigau (SAR). The  $\alpha = 0.2$  value optimizes the results for WordNet1.4.

2. The same formula without  $\alpha$  (which was optimized by Agirre and Rigau for a different, much smaller test collection).
3. The *simple density formula* (SDF) =  $m / desc_c$ . A simple baseline to test the importance of the conceptual density formula.
4. The *logarithmic formula* (LF) =  $\frac{1}{desc_c} \log_2 d \sum_{i=0}^{m-1} adesc^i$ . Where  $d$  is the depth of the concept  $c$  in the hierarchy. This is the AR formula with a correction factor to favor more specific concepts (deeper in the hierarchy)

**Window size.** We have experimented with various window sizes.

**Selection of related concepts** When it comes to selecting the concepts related to a sense through  $R$  we have taken several possibilities into account.

- First, we have a parameter to rule out the upmost levels of the hierarchy induced by the transitive closure of  $R$ . The reason for this is that the higher levels in broad conceptual hierarchies tend to be helplessly subjective. If there is a concept representative of the topic discussed in the word window and this concept is supposed to be meaningful for the disambiguation task, it shouldn't be too abstract or generic (as the concepts at higher levels usually are) in relation to the word senses being disambiguated. The conceptual density formulas are designed to reflect this but with big window sizes it is inevitable that WordNet tops get high densities. A value of 0 in this parameter represents considering the whole hierarchy.
- We introduce another parameter  $l$ , to consider only the nearer  $l$  concepts through transitive application of  $R$ . In other words, when computing the conceptual density of a concept  $c$ , we won't consider the weight of a subconcept  $s$  if we have to iterate through  $R$  more than  $l$  times to reach  $c$ . The idea behind this is that a concept  $c$  and its immediate hypernym will be closely related semantically (as would be the case between highway\_1<sup>1</sup> and road\_1 in Wordnet); On

---

<sup>1</sup>Following the convention that  $w_i$  is the  $i$ th sense in WordNet of the word  $w$

the contrary, although a highway is surely an entity it is unclear that this information has any semantic impact in a disambiguation task. It may seem that this parameter and the one described above will yield similar results, but they show a very different behavior in our experiments. In this parameter, a limit value of 0 represents taking into account all the concepts related via  $R$ .

**Sense weighting.** To compute the conceptual density of a concept  $c$ , in the hierarchy induced by  $R$  we have considered three possibilities to count and weight how many marks,  $m$ , lie below :

**synsets** Counting each sense of the words in the window related to  $c$  as one mark. This is the original formulation of Agirre & Rigau. The problem here is that the words interfere severely with themselves. If we take for instance the word *end*, which has 14 senses in WordNet, and draw the hypernymy hierarchy for the senses under *entity* (with some intermediate nodes omitted for clarity) we get the results in Figure 1.

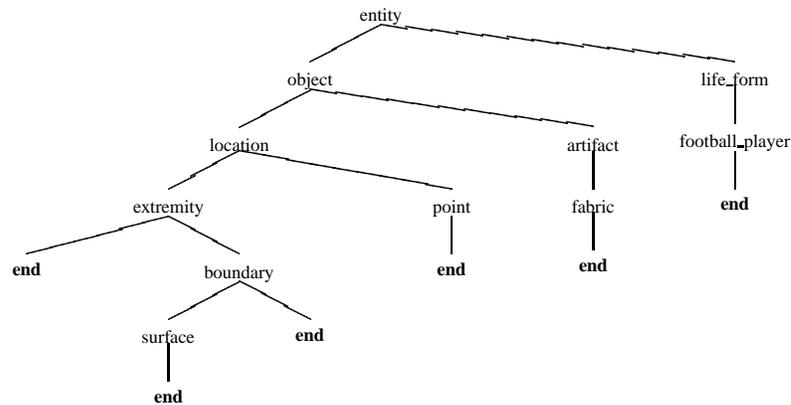


Figure 1: the hierarchy of end

It is easy to see here that the remaining 8 senses of *end* (which are not hyponyms of *entity*) will probably be discriminated against these because, in the absence of any context, the concept *object* in the figure gets a very high density. If we add more words as context in the window, the chances are that the majority of the senses will fall under the subhierarchy of *entity* and the algorithm would discard the other senses. Another adverse effect of highly polysemous words is that they tend to dominate the conceptual density measures. For instance, *end* has 14 senses and therefore 14 marks in the density measures, and that seems very unfair given that around one-third of the words in running text are monosemous. In order to minimize these effects, we have tested two additional forms of weighting senses:

**fractional** Counting for every sense a word in the window  $1/m$  (where  $m$  is the total number of senses of that word) to prevent a highly polysemous word from bi-

asing the conceptual density, although probably this won't prevent some words from disambiguating themselves.

**words** Counting as marks under the subhierarchy of a concept only the number of different words in the window contributing with senses under  $c$ . This way, all words in the window will contribute the same and also a high local in-word density (usually derived of the fine-grainedness of WordNet) shouldn't discriminate the senses of that word outside that area.

### 3 Evaluation

The evaluation has been conducted on the Semcor collection [MLTB93], a set of 171 documents where all content words are annotated with the most appropriate Wordnet sense. In our evaluation, each of the versions of the WSD algorithm has been tested on every noun in every Semcor document.

The behavior of the system is reported as a *recall* measure as defined for the first SENSEVAL campaign :

The score regime allows scores between 0 and 1 where the system returns more than one sense for an instance, with the probability mass shared. Recall is computed by dividing the system's scores over all correct senses by the total number of items to be disambiguated.

This measure compares correct disambiguations against all nouns in the collection; therefore, a system that is very precise but has a low coverage will also have a low recall overall.

#### 3.1 Overall performance

Table 1 compares the original Agirre-Rigau algorithm, our best conceptual density system, and three reference measures: a most frequent sense heuristic (always picking up the first wordnet sense), a random WSD baseline and a classical WSD strategy based on cocurrence of words in dictionary definitions (Lesk).

WSD algorithm	Recall
UNED conceptual density	31.3%
Lesk	27.4%
Agirre-Rigau	22.0%
Random baseline	28.5%
Most frequent heuristic	70.0%

Table 1: Overall performance

Surprisingly, the recall of the original Agirre-Rigau system is below the random baseline. This figure is slightly misleading, because the precision of Agirre-Rigau is above the

random baseline; But a random election has a 100% coverage, while the original conceptual density measure is not able to disambiguate all words. In any case, the performance of the original density measure is much poorer than expected. Results reported in [AR96] were more promising, but they were obtained on a test collection 50 times smaller than the whole Semcor collection (they used only four Semcor documents).

Our best system achieves 31.3% recall, a 42% improvement over the original Agirre-Rigau system. This is a dramatic improvement with respect to the original algorithm, but still the results are far below the most frequent sense heuristic. The comparison with the 70% recall of this simple heuristic could lead to discard conceptual relations as a source of information for the Word-Sense Disambiguation task. This would be, however, an erroneous conclusion, for a number of reasons:

- We have also compared the performance of conceptual density with a classical WSD algorithm based on contextual information in dictionary definitions [Les86], which was used as a strong baseline in SENSEVAL 1 [KR00]. The recall of Lesk algorithm is 27.4%, also below the random baseline! There are some reasons that explain these results:
  - The human annotations, taken as a gold standard, are biased in favor of the first wordnet sense, (which corresponds to the most frequent). Human annotators, in an all-words disambiguation task, have to select the appropriate sense for a different word in each iteration, each word having more than 5 senses in average. Inevitably, the annotator tends to pick up the first sense that seems to fit the context, and this produces a bias in favor of higher ranked senses. Studies on WSD evaluation [RY99, KR00] have argued in favor of a lexical sample task, where the annotator repeatedly annotates occurrences of the same word, reaching a minimal familiarity with the senses of the chosen word. This was the approach taken in SENSEVAL-1, where the Lesk algorithm behaves much better than in this Semcor-based evaluation. Unfortunately, the SENSEVAL test collection is based on a different dictionary (*Hector*), and thus cannot be used to test our conceptual density strategies.
  - Beyond human annotation problems, the all-words task implies that the system must be repeatedly attempting to disambiguate instances of very common terms, which may have 20 different senses in the database. This terms are almost impossible to disambiguate, and probably also useless to disambiguate for a majority of applications.

A more appropriate conclusion would be, then, that neither conceptual nor contextual measures are sufficient, in isolation, to perform accurate Word Sense Disambiguation.

- Our algorithm assigns probabilities to senses (unlike the most frequent sense heuristic), and the overall distribution of probabilities produces better results in a Text Retrieval system based on concept retrieval than the most frequent heuristic, as we have previously reported in [VPG99]. This is an indication that the recall measure in a pure WSD task may not reflect the utility of a WSD system in final Natural

Language applications. This indirect measure is a proof of the potential utility of conceptual density measures for Word-Sense Disambiguation.

We will focus now on the separate evaluation of all the variants introduced in the original algorithm, which led us to the best combination reported here.

### 3.2 Type of conceptual relation

Table 2 shows the results of the algorithm using different types of semantic relations. Apparently, meronymy/holonymy relations do not add any useful information to hypernymy.

Relation	Recall
Hypernymy	31.28%
Hypernymy + Meronymy	31.28%
Hypernymy + Holonymy	30.88%
Meronymy	26.62%
Holonymy	27.00%

Table 2: Recall with different conceptual relations

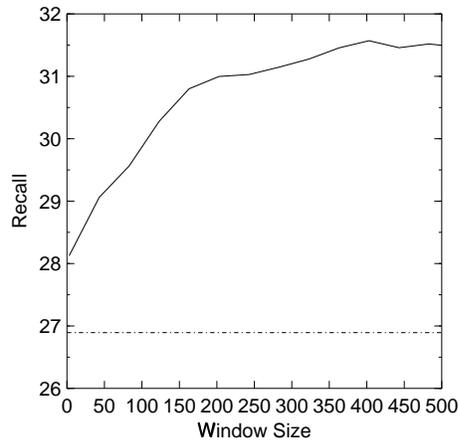


Figure 2: Effects of window size

### 3.3 Window size

Figure 2 shows the behavior of the algorithm with window sizes between 1 and 500 words. Remarkably, disambiguation gets consistently (although steadily) better with larger win-

dow sizes up to 150 words. This probably means the contextual information in a whole document is useful to disambiguate a word, providing topic information for the document.

### 3.4 Conceptual Density Formula

The effects of the density formula can be seen in Table 3. The alternative formulations LF and SDF behave worse than the original formula by Agirre and Rigau. However, their  $\alpha$  parameter, which was adjusted to 0.2 in order to optimize disambiguation over four particular Semcor documents in WordNet 1.4, is clearly inadequate when evaluating against all Semcor documents in WordNet 1.5 :  $\alpha = 1$  (AR) produces a 10% improvement against  $\alpha = 0.4$  (SAR).

Table 3: Effects of conceptual density measures

Density Formula	Recall
AR	31.28%
SAR	27.45%
LF	26.60%
SDF	23.29%

The different formulas give recall figures between 23.3% and 31.3% (50% greater), showing that choosing an adequate formula has a direct impact on the results. Perhaps a better formula could further improve results.

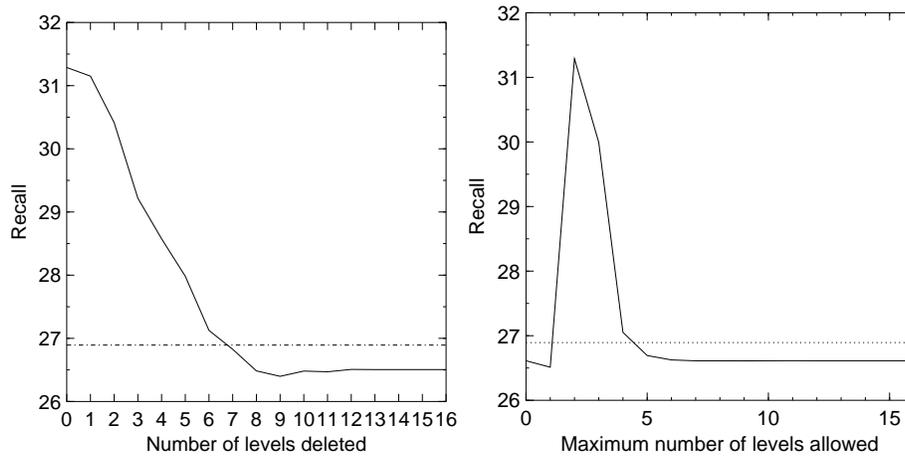


Figure 3: Selection of synsets

### 3.5 Selection of synsets

#### Removal of upper levels

Figure 3 (left plot) shows the effects of removing upper levels of the hierarchy. Contrary to our hypothesis, even only removing the two upper levels harms the recall of the system. Removing more than 6 levels produces a random behavior, as most information in WordNet lies in the first 6 levels. This results seems to indicate that the WSD algorithm is not performing as expected: the upper levels are used in the disambiguation, and therefore the conceptual density measure is using conceptual relationships that are far too general to be meaningful for disambiguation. This can partially explained the poor absolute performance of the density measures.

#### Upper limit on hierarchical chains

The effects of limiting the inspection of hypernym chains are shown in Figure 3 (right plot). The plot shows that the algorithm is useless without such limitation, and the optimal limit is two.

This criterion confirms that going up in the hierarchy without limitation introduces noise - due to underspecific concepts - that spoils the performance of the algorithm.

### 3.6 Sense weighting

Table 4 shows recall for the three approaches to sense weighting. Surprisingly, assigning lower weights to senses for highly ambiguous words ("Fractional") does not improve performance over the standard approach ("Synsets"). Taking the number of different source words in the density measure ("Words") produces an improvement, but nearly negligible.

Criterion	Recall
Words	31.28%
Synsets	30.81%
Fractional	27.94%

Table 4: Effects of sense weighting

### 3.7 Behavior on different text categories

The Semcor documents, a fraction of the Brown Corpus [FK82], are classified according to a set of predefined domains (Press, General Fiction, Romance, Humor, etc.). It is interesting to see how WSD performance varies along different document categories. In Table 5, overall performance is split according to such categories. Categories where conceptual density works better are ranked higher in the table.

The results are remarkable. While a random disambiguation produces similar recall figures (indicating that the mean polysemy is similar in any kind of documents), the WSD system performs better on non-fiction categories (*Press: reportage, reviews, skills and*

Text category	Random recall	Algorithm recall	Improvement
A. Press: reportage	26.95%	36.67%	36.08%
C. Press: reviews	27.06%	34.91%	29.01%
E. Skills & hobbies	26.89%	33.68%	25.26%
F. Popular Lore	26.73%	32.79%	22.66%
D. Religion	25.75%	31.22%	21.23%
H. Miscellaneous	26.14%	31.65%	21.09%
J. Learned	27.17%	32.78%	20.64%
L. Mystery & detective fiction	25.29%	29.89%	18.16%
P. Romance & love story	25.03%	29.19%	16.63%
B. Press: editorial	27.64%	31.69%	14.65%
G. Belles lettres, biography, essays	28.14%	31.93%	13.48%
M. Science fiction	26.49%	29.87%	12.74%
K. General fiction	25.63%	28.32%	10.49%
R. Humor	26.66%	29.06%	9.00%
N. Adventure & western fiction	22.08%	23.06%	4.45%

Table 5: WSD performance in different text categories

*hobbies*, etc.), and worse on fiction categories (*adventure*, *humor*, *general fiction*). Conceptual density improves random WSD in a 36% for *Press: reportage*, while for *Adventure & Western Fiction* the improvement is negligible (4.5%). This confirms the hypothesis that WSD is more plausible in technical documents, where word senses have clearer distinctions, metaphors are less common, and the context provides more accurate domain information.

## 4 Conclusions

We have provided an exhaustive evaluation of different WSD algorithms that rely solely on the conceptual relations between candidate word senses. Our point of departure has been the Agirre-Rigau algorithm, based on a conceptual density measure over the WordNet hierarchy. This algorithm, which had a competitive performance over smaller test collection, behaves poorly in a complete evaluation against all Semcor documents. We have experimented with many kinds of improvements to the algorithm, and tuned all parameters associated to them, obtaining evaluations for more than fifty variants of the WSD system. For comparison purposes, we have also implemented and evaluated a simple version of the classical Lesk algorithm, based solely on contextual information from dictionary definitions, which is also used in SENSEVAL WSD competitions.

Some of the main conclusions from our experiments are:

- Our best system performs 42% better than the original Agirre-Rigau algorithm, and 14% better than the Lesk algorithm based on cooccurrence in dictionary definitions. This improvement is obtained with an implementation that runs in linear time and

has been used to disambiguate large text collections in three different languages (English, Spanish and Catalan) within the ITEM project [VGP<sup>+</sup>00]. Its performance, however, is still low in terms of absolute recall, indicating that conceptual relations should be combined with other types of information (contextual, syntactic, domain information, etc.). We have also argued that the test collection itself -Semcor- is not appropriate for testing systems; it is desirable that the new Senseval initiative creates a better evaluation ground to provide a reliable way of measuring the effectiveness of WSD systems in final NLP tasks.

- We have shown that, in practice, the original Agirre-Rigau algorithm uses long hierarchical chains to disambiguate, which are associated with vague conceptual associations that give noisy results. Our optimal setting uses maximal hypernymy chains of size 2, combined with other optimizations to keep the coverage of the system. We have also shown that bigger window sizes provide better results, because they exploit all domain information in a text to disambiguate.
- We have provided quantitative evidence proving that WSD is more feasible on non-fiction, domain-specific documents rather than on general fiction texts with this technique.
- Finally, we have shown that a direct comparison of recall with a Most frequent heuristic over Semcor does not reflect the properties of WSD systems; our system has a much lower recall than this heuristic, but gives better results in a text retrieval experiment using word senses, as we have previously shown in [VPG99]. This is a strong evidence that the evaluation of WSD systems should also be measured indirectly in NLP applications.

## 5 Acknowledgments

This work has been supported by a Ph.D. research grant from the Comunidad Autónoma de Madrid, and also by the Spanish Comisión Interministerial de Ciencia y Tecnología, project *Hermes* (TIC2000-0335-C03-01).

## References

- [AR96] Eneko Agirre and German Rigau. Word sense disambiguation using conceptual density. *COLING (International Conference in computational linguistics)*, 1996.
- [Fel98] C. Fellbaum. A semantic network of english: the mother of all wordnets. *Computers and the Humanities, Special Issue on EuroWordNet*, 1998.
- [FK82] W. N. Francis and H. Kucera. *Frequency Analysis of English Usage: Lexicon and Grammar*. Houghton Mifflin Company, Boston, 1982.
- [GRA97] Jordi Atserias German Rigau and Eneko Agirre. Combinig unsupervised lexical knowledge methods for word sense disambiguation. In *Proceedings of the ACL (Association for Computational Linguistics)*, 1997.

- [KR00] A. Kilgarriff and J. Rosenzweig. Framework and results for english SENSE-VAL. *Computers and the Humanities*, 34(1-2), 2000.
- [Les86] Michael E. Lesk. Automatic sense disambiguation using machine readable dictionaries : How to tell a pine cone from an ice cream cone. In *Proceedings of SIGDOC (Special Interest Group for Documentation) Conference, Toronto, Canada*, 1986.
- [MLTB93] G. A. Miller, C. Leacock, R. Teng, and R. T. Bunker. A semantic concordance. In *Proceedings of the ARPA (Advanced Research Projects Agency) Workshop on Human Language Technology*. Morgan Kaufman, 1993.
- [NL96] H. Ng and H. Lee. Integrating multiple knowledge sources to disambiguate word sense: an exemplar-based approach. In *Proceedings of ACL (Association for Computational Linguistics)*, 1996.
- [RY99] P. Resnik and D. Yarowsky. Distinguishing systems and distinguishing senses: New evaluation methods for word sense disambiguation 5(2):113–134. *Natural Language Engineering*, 1999.
- [SW99] Mark Stevenson and Yorick Wilks. Combining weak knowledge sources for sense disambiguation. *Proceedings of IJCAI (International Joint Conference on Artificial Intelligence) in Stockholm, Sweden*, 1999.
- [VGP<sup>+</sup>00] F. Verdejo, J. Gonzalo, A. Peñas, F. López, and D. Fernández. Evaluating wordnets in Cross-Language Text Retrieval: the ITEM multilingual search engine. In *Proceedings LREC (International Conference on Language Resources and Evaluation)*, 2000.
- [Vos98] P. Vossen. *EuroWordNet: a multilingual database with lexical semantic networks*. Kluwer Academic Publishers, 1998.
- [VPG99] P. Vossen, W. Peters, and J. Gonzalo. Towards a universal index of meaning. In *Proceedings of SIGLEX(Special Interest Group on the Lexicon)*, 1999.
- [WS98] Yorick Wilks and Mark Stevenson. Word sense disambiguation using optimised combinations of knowledge sources. *Proceedings of COLING (International Conference in computational linguistics) - ACL (Association for Computational Linguistics) in Montreal, Quebec, Canada*, 1998.