Combining Weak Knowledge Sources for Sense Disambiguation

Mark Stevenson and Yorick Wilks
Department of Computer Science
University of Sheffield,
Regent Court, 211 Portobello Street,
Sheffield, S1 4DP
United Kingdom
{marks,yorick}@dcs.shef.ac.uk

Abstract

There has been a tradition of combining different knowledge sources in Artificial Intelligence research. We apply this methodology to word sense disambiguation (WSD), a long-standing problem in Computational Linguistics. We report on an implemented sense tagger which uses a machine readable dictionary to provide both a set of senses and associated forms of information on which to base disambiguation decisions. The system is based on an architecture which makes use of different sources of lexical knowledge in two ways and optimises their combination using a learning algorithm. Tested accuracy of our approach on a general corpus exceeds 94%, demonstrating the viability of all-word disambiguation as opposed to restricting oneself to a small sample.

1 Introduction

The methodology and evaluation of word sense disambiguation (WSD) as a distinct task are somewhat different from those of others in NLP, and one can distinguish three aspects of this difference, all of which come down to evaluation problems, as does so much in NLP these days. First, researchers are divided over using a general method (one that attempts to apply WSD to all the content words of texts, the large vocabulary approach taken in this paper) versus one that is applied to only a small trial selection of words (for example [Schütze, 1992] and [Yarowsky, 1995]). The latter researchers have obtained very high levels of success: Yarowsky quotes 97% correct disambiguation for the small vocabulary over which his system operates, results close to the figures for other "solved" NLP tasks, such as part of speech taggers. The issue is whether these small word sample methods and techniques will transfer to general WSD over a more complete vocabulary.

Others, besides ourselves (for example [Mahesh *et al.*, 1997] and [Harley and Glennon, 1997]) have pursued the general option on the grounds that it is the real task and should be tackled directly, even with rather lower success rates. The division between the approaches probably comes down to no more than the availability of gold standard text in sufficient quantities, which is more costly to obtain for WSD than other tasks. In this paper we describe a method we have used for obtaining more test material by transforming one resource into another, an advance we believe is unique and helpful in this impasse.

Secondly, there have also been deeper problems about evaluation, which led sceptics like [Kilgarriff, 1993] to question the whole WSD enterprise, because it is harder for subjects to assign one and only one sense to a word in context (and hence produce the test material itself) than to perform other NLP related tasks. One of the present authors has discussed Kilgarriff's arguments elsewhere [Wilks, 1997] and argued that they are not, in fact, as gloomy as he suggests. Again, this is probably an area where there is an "expertise effect": some subjects can almost certainly make finer, more inter-subjective, sense distinctions than others in a reliable way, just as lexicographers do [Jorgensen, 1990][Felbaum *et al.*, 1997].

But there is a third, quite different, source of unease about the evaluation base: everyone agrees that new senses appear in corpora that cannot be assigned to any existing dictionary sense, and this is an issue of novelty, not just one about the difficulty of discrimination. If that is the case, it tends to undermine the standard mark-up-model-and-test methodology of most recent empirical NLP, since it will not then be possible to mark up sense assignment in advance against a dictionary if new senses are present. We shall not tackle this difficult issue further here, but press on towards experiment.

One further issue must be mentioned, because it is unique to WSD as a task and at the core of our approach. Unlike other well-known NLP modules, WSD seems to draw upon a number of apparently different information sources. All the following have been implemented as the basis of experimental WSD at various times: part of speech, semantic preferences, collocating items or classes, thesaural or subject areas, dictionary definitions, synonym lists, among others (including bilingual equivalents in parallel texts). These linguistic phenomena seem different, so how can they all be, separately or in combination, informational clues to a single phenomenon, WSD? This is a situation quite unlike syntactic parsing or part of speech tagging: in the latter case, for example, one can write a Cherry-style rule tagger or an HMM learning model, but there is no reason the believe these represent different types of information, rather than different ways of conceptualising and encoding it. That seems not to be the case, at first sight, with the many forms of information for WSD.

2 The Methodology of Combining Knowledge Sources

In our work we adopted the methodology first explicitly proposed for WSD by [McRoy, 1992], and more recently [Ng and Lee, 1996] and [Wilks and Stevenson, 1998b], namely that of bringing together a number of partial sources of information about a phenomenon and combining them in a principled manner. This is in the AI tradition of combining "weak" methods for strong results (usually ascribed to [Newell, 1973]) and used in the CRL-NMSU lexical work on the Eighties [Wilks *et al.*, 1990]. We shall present a system that combines three of the types of information listed above (together with part of speech filtering) and, more importantly, applies a learning algorithm to determine the optimal combination of such modules for a given word distribution; it being obvious, for example, that thesaural methods work better for nouns than for verbs, and so on.

We shall use the machine readable version of the *Longman Dictionary of Contemporary English* (LDOCE)[Procter, 1978] for our experiments. This is a learners' dictionary, designed for students of English, which contains around 36,000 word types. LDOCE was innovative in its use of a defining vocabulary of 2,000 word from which the textual definitions were written, if a learner of English could master this small core then, in theory, they could understand every entry in the dictionary. In LDOCE, the senses for each word type are grouped into *homographs*, sets of senses with related meanings. For example, one of the homographs of "bank" means roughly 'things piled up', the different senses distinguishing exactly what is piled up. It should be noted that the granularity of sense distinctions at the LDOCE homograph level (eg. "bank" as 'edge of river' or 'financial institution') is comparable to the distinctions made by small-scale WSD algorithms (eg. [Schütze, 1992] and [Yarowsky, 1995]).

It seems that there is a difference in the way in which different lexical knowledge sources can be useful for WSD in different ways. In experiments with LDOCE part of speech codes and the Brill tagger [Wilks and Stevenson, 1998a] suggest that this source can be used to discriminate between senses, or homographs, which are possibly correct in context, and those which are very likely not to be. This source could then be used to remove, or *filter*, senses from the set of possibilities for ambiguous words. However, this strategy can only be used for knowledge sources which we have confidence in since, if the correct sense is removed from consideration, then the tagger can never correctly disambiguate that word. We propose a framework in which separate knowledge sources can be used for WSD either to remove senses which are very unlikely or to suggest senses which may be correct. The first type of module shall be dubbed as a *filter* and the second type will be *partial taggers*.

3 A Sense Tagger

We now go on to describe and evaluate a sense tagger implemented within this methodology. Our sense tagger makes use of several modules which perform disambiguation, each being a filter or partial tagger. LDOCE is used to provide a set of senses and as a knowledge base to provide information upon which disambiguation decisions can be made. The architecture of the system is represented in Figure 1, we now go on to describe each component in detail.

3.1 Preprocessing

Before the filters or partial taggers are applied the text is tokenised, lemmatised, split into sentences and part of speech tagged using the Brill syntactic tagger [Brill, 1992]. A named entity identifier is then run over the text to mark and categorise proper names. These preprocessing stages are carried out by modules from Sheffield University's Information Extraction system, *LaSIE* [Gaizauskas *et al.*, 1996].

Our system disambiguates only the content words in the text, the part of speech tags assigned by Brill's tagger are used to decide which are content words, and does not attempt to disambiguate any of the words identified as part of a named entity.

3.2 Part of Speech

Our first module makes use of part of speech tags. We take the part of speech tags assigned by the Brill tagger and use a manually created mapping to translate these to the corresponding LDOCE grammatical category. Any senses which do not correspond to the category returned are removed from consideration. In practice the part of speech filtering is carried out at the same time as the lexical lookup phase and the senses whose grammatical category does not correspond to the tag assigned are never attached to the ambiguous word. This avoids attaching senses which will be immediately removed by the filter. There is also an option to turn off filtering so that all senses are attached regardless of the part of speech tag.

It could be reasonably argued that removing senses is a dangerous strategy since, if the part of speech tagger made an error, the correct sense could be removed from consideration. As a precaution against this we have designed our system so that if none of the dictionary senses for a given word agree with the part of speech tag then all are kept. There is also good evidence from [Wilks and Stevenson, 1997] that this approach works well despite part of speech tagging errors.

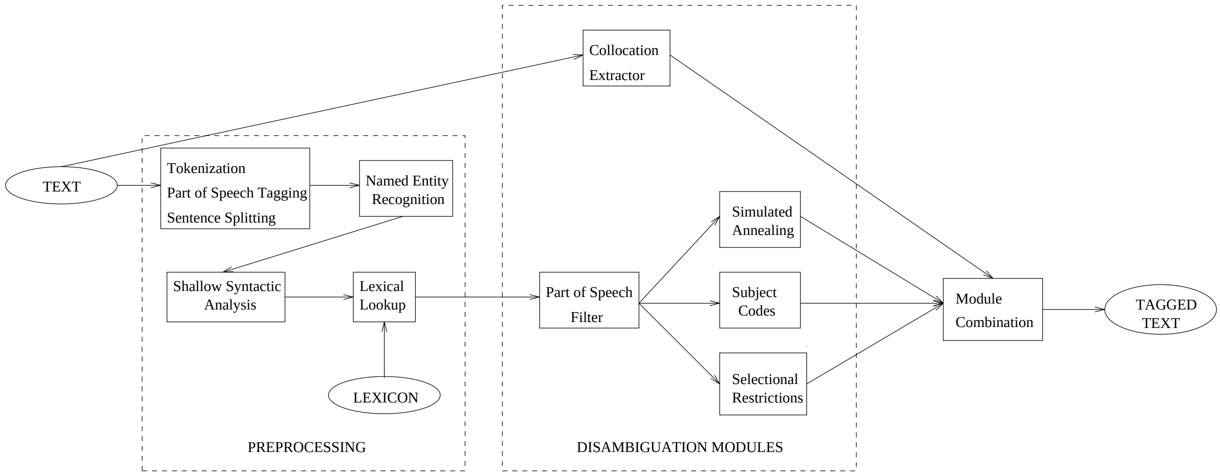

Figure 1: Sense Tagger Architecture

3.3 Dictionary Definitions

A method was proposed by [Lesk, 1986] for carrying out sense disambiguation which used an overlap count of content words in dictionary definitions as a measure of semantic closeness. In this way it is possible, at least in theory, to tag each word in a sentence with its sense from any dictionary which contains textual definitions for its senses. However, it was found that the computations which would be necessary to test every combination of senses, even for a sentence of modest length, was prohibitive.

The approach was made practical by [Cowie *et al.*, 1992] who computed the overlap using the simulated annealing optimisation algorithm which eliminated the need to calculate all possible combinations of senses. An initial guess at the solution to a given problem is made and the algorithm gradually moves towards an optimal solution by generating permutations of the current solution, and evaluating which of these are improvements. As with all hill-climbing algorithms, there is the danger that the algorithm will converge on a locally optimal solution rather than the desired optimal global solution. Simulated annealing avoids this by introducing a stochastic random element dependent on the temperature of the system, when the temperature is high there is a high probability that the algorithm will choose a solution that is worse than the current solution, with the probability of this happening reducing when the temperature is low. The temperature is high when the process begins and is gradually reduced as the algorithm proceeds. This random element allows the search to jump away from local minima and find the true global solution. This approach correctly disambiguated 47% of words to the sense level, and 72% to the homograph level.

In the Cowie et. al. implementation the optimisation was carried out over a simple count of definition words in common, however this meant that longer definitions were preferred (since they have more words which can contribute to the overlap) and short definitions or definitions by synonym were correspondingly penalised. We attempted to solve this problem by computing the overlap in a different way. Instead of each word contributing one we normalised its contribution by the number of words in the definition it came from. The Cowie et. al. implementation returned one sense for each ambiguous word in the sentence, without any indication of the system's confidence in its choice, but we adapted the system to return a set of suggested senses for each ambiguous word in the sentence. We found that our changes led to an improvement in the algorithm's effectiveness and 65% of senses are correctly disambiguated by this module.

3.4 Selectional Restrictions

LDOCE senses contain simple selectional restrictions for each content word in the dictionary. A set of 35 semantic classes are used, such as `H` = Human, `M` = Human male, `P` = Plant, `S` = Solid and so on. Each word sense for a noun is given one of these semantic types; senses for adjectives list the type which they expect for the noun they modify; senses for adverbs the type they expect of their modifier and verbs list between one and three types, depending on their transitivity, which are the expected semantic types of the verb's subject, direct object and indirect object. Grammatical links between verbs, adjectives and adverbs and the head noun of their arguments are identified using a specially constructed shallow syntactic analyser [Stevenson, 1998].

The semantic classes in LDOCE are not formed into a hierarchy, but [Bruce and Guthrie, 1992] manually identified hierarchical relations between the semantic classes, placing them in a hierarchy which we use to resolve the restrictions. We resolve the restrictions by returning, for each word, the set of senses which do not break the constraints (that is, those whose semantic category is at the same level, or lower, in the hierarchy).

The selectional restriction resolution algorithm makes use of the information provided by the named entity identifier (Section 3.1). Although we are not disambiguating named entities they are still useful to help disambiguate other words: for example, if a verb has two senses one of which places the restriction `H` (=Human) on its object, the other `I` (=Inanimate) and the object was a named entity marked `PERSON` then we would prefer the first sense for that verb.

We implemented another voting system for this partial tagger and found that 44% of words were correctly disambiguated by this module.

3.5 Subject Codes

Our final partial tagger is a reverse engineering of the broad context algorithm developed by [Yarowsky, 1992]. This algorithm is dependent upon a categorisation of words in the lexicon into subject areas, Yarowsky used Roget large categories. In LDOCE pragmatic codes indicate the subject area of senses and, since primary codes have a wider coverage, we chose them as our subject categories. Since Roget is a thesaurus each entry in the lexicon belongs to a large category, however not every LDOCE sense has a primary pragmatic code. In order to counter this we created a dummy category, denoted by `--`, used to indicate a sense which is not associated with any specific subject area and this category is assigned to all senses without a pragmatic code. The differences between the structures of LDOCE and Roget meant that we had to adapt the original algorithm reported in [Yarowsky, 1992]. Space restrictions prevent us from reporting this in detail, however a detailed account is provided in [Stevenson, 1999]. After this partial tagger has computed the most likely pragmatic code, the set of senses marked with that code are returned for each ambiguous word. We also implemented a voting system for this partial tagger and found that it was the most effective, disambiguating 79% of senses.

3.6 Combining Knowledge Sources

Each partial tagger can only suggests possible senses for each word and so it is necessary to have some method to combine the results. We decided that the most effective way to carry this out would be to make use of the algorithms produced by the machine learning research community. Consequently we experimented with several of the publicly available algorithms and examined three main approaches: inductive logic programming (ILP), rule induction and memory-based learning. ILP and rule induction approaches operate by representing data as a set of rules abstracted from training data. Memory based learning stores training examples and classifies new instances by identifying the closest one. We examined the `PROGOL` algorithm [Muggleton, 1995] as a representative of ILP approaches, the `CN2` algorithm [Clark and Niblett, 1989] for rule induction approaches and the `TiMBL` algorithm [Daelemans *et al.*, 1998] for memory based learning. We found that the `TiMBL` algorithm was most suitable for our purposes since it carried out the required processing in a reasonable time as well as producing good results.

We presented the learning algorithm with a number of training words for which the correct sense is known. The senses for each training word are represented in a feature vector format, with a vector for each sense, apart from those removed by the part of speech filter (Section 3.2). The vector consists of the results from each of the partial taggers, frequency information and 10 basic collocations (first noun/verb/preposition to the left/right and first/second word to the left/right). (A simple module, the Collocation Extractor, is used to identify these from the source text.) Each sense is marked as either `appropriate` (if it is the correct sense given the context) or `inappropriate`. The learning algorithm stores each of the example senses according to its classification (`appropriate/inappropriate`).

To disambiguate un-tagged text, the partial taggers and filters are run and the learning algorithm used to identify the training instance which is most similar to the new, unclassified, example. If the memory based learner suggests that more than one sense is appropriate for any given word then the first of those is chosen as a tie-breaker.

Although the system is trained on a fixed vocabulary it is restricted to these. If a word is encountered which was not in the training data then the results of the partial taggers and frequency information can be used to make the disambiguation decision.

4 Producing an Evaluation Corpus

Since our system is designed to disambiguate all content words in text the most appropriate evaluation procedure will be to compare the output of the system against some "gold standard" texts, but these are very labour-intensive to obtain. Lexical semantic markup is generally accepted as a more difficult and time-consuming task than part of speech markup. Rather than expend a vast amount of effort on manual tagging we decided to adapt two existing resources to our purposes. We took SEMCOR [Landes *et al.*, 1998], a 200,000 word corpus with the content words manually tagged as part of the WordNet project. The semantic tagging was carried out under disciplined conditions using trained lexicographers with tagging inconsistencies between manual annotators controlled. SENSUS [Knight and Luk, 1994] is a large-scale ontology designed for machine-translation and was produced by merging the ontological hierarchies of WordNet and LDOCE [Bruce and Guthrie, 1992]. To facilitate this merging it was necessary to derive a mapping between the senses in the two lexical resources. We used this mapping to translate the WordNet-tagged content words in SEMCOR to LDOCE tags.

The mapping is not one-to-one, and some WordNet senses are mapped onto two or three LDOCE senses when the WordNet sense does not distinguish between them. The mapping also contained significant gaps (words and senses not in the translation). SEMCOR contains 91,808 words tagged with WordNet synsets, 6,071 of which are proper names which we ignore, leaving 85,737 words which could potentially be translated. The translation contains only 36,869 words tagged with LDOCE senses, although this is a reasonable size for an evaluation corpus for this type of task; it is several orders of magnitude larger than those used by other researchers working in large vocabulary WSD, for example [Cowie *et al.*, 1992], [Harley and Glennon, 1997] and [Mahesh *et al.*, 1997]. This corpus was also constructed without the excessive cost of additional hand-tagging and does not introduce any inconsistencies may occur with a poorly controlled tagging strategy.

5 Results

Our system was tested using a technique known as 10-fold cross validation. This process is carried out by splitting the available data into ten roughly equal subsets. (This is done by randomly selecting the first tenth, then choosing another from the remaining data and so on until only one tenth remains.) One of the subsets is chosen as the test data with the `TiMBL` algorithm being trained on the remainder. This is repeated ten times, so that each subset is used as test data exactly once, and results are averaged across each of the test runs. This technique provides two advantages; firstly, the best use can be made of the available data and, secondly, the computed results are more statistically reliable than those which would be obtained by simply setting aside a single portion of the data for testing.

We found that the system correctly disambiguated 90% of the ambiguous instances to the fine grained sense level and in excess of 94% to the homograph level. We also analysed the performance of our system over each of the four different grammatical categories it analysed and these results are shown in Table 1. [Yarowsky, 1995] comments that nouns tend to be disambiguated by broad contextual considerations while adjectives, adverbs and verbs are more affected by the words acting as their arguments. This would suggest that our partial taggers may have different effects over the four grammatical categories on which they operate. Future research is planned to investigate this in detail.

6 Conclusion

These experimental results show that it is possible to disambiguate a large vocabulary of content words to high levels of accuracy at both the rough-grained homograph and fine-grained sense levels. Our system uses an optimised combination of diverse lexical knowledge sources and this appears to be a successful strategy for this problem. Although the results reported here are slightly lower than those reported for systems which disambiguate a very restricted vocabulary, such as [Yarowsky, 1995] who quotes 97% for a test set of 12 words, our figure is far greater than has been achieved so far by other large vocabulary disambiguation systems such as [Harley and Glennon, 1997].

The fact that the optimised figure from the module learning (90%) is so much larger than that from the individual modules (which range between 44% and 79%) shows that the information content of the different modules must be different (i.e. are not notational variants of each other) or else the higher, optimised, figure would not be possible.

Acknowledgments

The work described here was supported by the European Union Language Engineering project ECRAN – Extraction of Content: Research at Near-market (LE-2110). The authors are also grateful for the comments provided by the three anonymous reviewers of this paper.

References

[Brill, 1992] E. Brill. A simple rule-based part of speech tagger. In *Proceeding of the Third Conference on Applied Natural Language Processing (ANLP-92)*, pages 152–155, Trento, Italy, 1992.

[Bruce and Guthrie, 1992] R. Bruce and L. Guthrie. Genus disambiguation: A study in weighted preference. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 1187–1191, Nantes, France, 1992.

[Clark and Niblett, 1989] P. Clark and T. Niblett. The CN2 Induction Algorithm. *Machine Learning Journal*, 3(4):261–283, 1989.

[Cowie *et al.*, 1992] J. Cowie, L. Guthrie, and J. Guthrie. Lexical disambiguation using simulated annealing. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 359–365, Nantes, France, 1992.

[Daelemans *et al.*, 1998] W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. TiMBL: Tilburg memory based learner version 1.0. Technical report, ILK Technical Report 98-03, 1998.

[Felbaum *et al.*, 1997] C. Felbaum, J. Grabowski, and S. Landes. Analysis of a hand-tagging task. In *Proceedings of the SIGLEX Workshop 'Tagging Text with Lexical Semantics: What, why and how?'*, pages 34–40, Washington, DC., 1997.

[Gaizauskas *et al.*, 1996] R. Gaizauskas, T. Wakao, K. Humphreys, H. Cunningham, and Y. Wilks. Description of the LaSIE system as used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 207 – 220, San Francisco, CA., 1996.

[Harley and Glennon, 1997] A. Harley and D. Glennon. Sense tagging in action: Combining different tests with additive weights. In *Proceedings of the SIGLEX Workshop "Tagging Text with Lexical Semantics"*, pages 74–78, Washington, DC., 1997.

[Jorgensen, 1990] J. Jorgensen. The psychological reality of word senses. *Journal of Psycholinguistic Research*, 19(3):167–190, 1990.

	All words	Nouns	Verbs	Adjectives	Adverbs
Homograph	94.34%	94.72%	93.30%	94.40%	90.67%
Sense	90.09%	91.16%	88.54%	90.60%	68.63%

Table 1: Accuracy of the Reported System

[Kilgarriff, 1993] A. Kilgarriff. Dictionary word sense distinctions: An enquiry into their nature. *Computers and the Humanities*, 26:356–387, 1993.

[Knight and Luk, 1994] K. Knight and S. Luk. Building a large knowledge base for machine translation. In *Proceedings of the American Association for Artificial Intelligence Conference (AAAI-94)*, pages 185–109, Seattle, WA., 1994.

[Landes *et al.*, 1998] S. Landes, C. Leacock, and R. Tengi. Building a semantic concordance of English. In C. Fellbaum, editor, *WordNet: An electronic lexical database and some applications*. MIT Press, Cambridge, MA., 1998.

[Lesk, 1986] M. Lesk. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of ACM SIGDOC Conference*, pages 24–26, Toronto, Canada, 1986.

[Mahesh *et al.*, 1997] K. Mahesh, S. Nirenburg, S. Beale, E. Viegas, V. Raskin, and B. Onyshkevych. Word sense disambiguation: Why have statistics when we have these numbers? In *Proceedings of the 7th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 151–159, Santa Fe, NM, 1997.

[McRoy, 1992] S. McRoy. Using multiple knowledge sources for word sense disambiguation. *Computational Linguistics*, 18(1):1–30, 1992.

[Muggleton, 1995] S. Muggleton. Inverse Entailment and Progol. *New Generation Computing Journal*, 13:245–286, 1995.

[Newell, 1973] A. Newell. Computer models of thought and language. In Schank and Colby, editors, *Artificial Intelligence and the Concept of Mind*. Freeman, San Francisco, CA, 1973.

[Ng and Lee, 1996] H. Ng and H. Lee. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th Meeting of the Association for Computational Linguistics (ACL-96)*, pages 40–47, Santa Cruze, CA, 1996.

[Procter, 1978] P. Procter, editor. *Longman Dictionary of Contemporary English*. Longman Group, Essex, UK, 1978.

[Schütze, 1992] H. Schütze. Dimensions of meaning. In *Proceedings of Supercomputing '92*, pages 787–796, Minneapolis, MN, 1992.

[Stevenson, 1998] M. Stevenson. Extracting syntactic relations using heuristics. In *Proceedings of the European Summer School on Logic, Language and Information '98*, pages 248–256, Saarbrücken, Germany, 1998.

[Stevenson, 1999] M. Stevenson. *Multiple Knowledge Sources for Word Sense Disambiguation*. PhD thesis, University of Sheffield, 1999.

[Wilks and Stevenson, 1997] Y. Wilks and M. Stevenson. Combining independent knowledge sources for word sense disambiguation. In *Proceedings of the Third Conference on Recent Advances in Natural Language Processing Conference (RANLP-97)*, pages 1–7, Tzigov Chark, Bulgaria, 1997.

[Wilks and Stevenson, 1998a] Y. Wilks and M. Stevenson. The grammar of sense: Using part-of-speech tags as a first step in semantic disambiguation. *Journal of Natural Language Engineering*, 4(2):135–144, 1998.

[Wilks and Stevenson, 1998b] Y. Wilks and M. Stevenson. Word sense disambiguation using optimised combinations of knowledge sources. In *The 17th International Conference on Computational Linguistics and the 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL-98)*, pages 1398–1402, Montreal, Canada, 1998.

[Wilks *et al.*, 1990] Y. Wilks, D. Fass, C.-M. Guo, J. McDonald, T. Plate, and B. Slator. A tractable machine dictionary as a basis for computational semantics. *Journal of Machine Translation*, 5:99–154, 1990.

[Wilks, 1997] Y. Wilks. Senses and Texts. *Computers and the Humanities*, 31:77–90, 1997.

[Yarowsky, 1992] D. Yarowsky. Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 454–460, Nantes, France, 1992.

[Yarowsky, 1995] D. Yarowsky. Unsupervised word-sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-95)*, pages 189–196, Cambridge, MA, 1995.