Using Verb Subcategorization for Word Sense Disambiguation

Will Roberts*, **Valia Kordoni**†

*Institut für Anglistik und Amerikanistik, Humboldt University,
10099 Berlin, Germany
will.roberts@rz.hu-berlin.de

†German Research Centre for Artificial Intelligence (DFKI GmbH)
Germany
kordoni@dfki.de

Abstract

We develop a model for predicting verb sense from subcategorization information and integrate it into SSI-Dijkstra, a wide-coverage knowledge-based WSD algorithm. Adding syntactic knowledge in this way should correct the current poor performance of WSD systems on verbs. This paper also presents, for the first time, an evaluation of SSI-Dijkstra on a standard data set which enables a comparison of this algorithm with other knowledge-based WSD systems. Our results show that our system is competitive with current graph-based WSD algorithms, and that the subcategorization model can be used to achieve better verb sense disambiguation performance.

Keywords: Word Sense Disambiguation, Subcategorization, Lexical Semantics

1. Introduction

Automatic Word Sense Disambiguation (WSD) often struggles with verbs. Evaluations show that verbs are the hardest words to tag for sense; they exhibit higher entropy in their sense distributions than other parts of speech; and, they have the lowest inter-annotator agreement when setting up gold standards for traditional evaluations. In this study, we develop a model that allows predicting verb sense from subcategorization, and integrate it into a wide-coverage WSD algorithm in an effort to boost WSD performance on verbs.

Subcategorization (i.e., the number and types of arguments co-occurring with a verb to form a VP constituent) and verb sense are known to be related; indeed, Levin's widely-used verb classification is grounded in the hypothesis that a verb's syntactic behaviour and its meaning are strongly connected. Roland and Jurafsky (2002) identify two factors which influence verb subcategorization preferences: *discourse factors*, such as the design and type of a corpus, and verb sense. By controlling for discourse factors and verb sense, they show that verb subcategorization is stable across domains and language variants, and conclude that verb sense is the single best predictor of verb subcategorization.

This dependency is often strong enough to allow predicting syntax from semantics and vice-versa. For instance, word sense disambiguation has been shown to improve automatic subcategorization acquisition (Korhonen and Preiss, 2003). However, and despite this apparently strong relationship, subcategorization is only infrequently used as an information source for WSD, and we are not aware of any study explicitly investigating the effect of subcategorization frame (SCF) on sense disambiguation. Chen and Palmer (2005) showed that supervised verb sense disambiguation performance could be improved by incorporating syntactic features, including detailed analysis of some subcategorization phenomena. In related work, Dang and Palmer (2005) suggest that semantic role labels and SCFs can both be useful sources of information for WSD. Andrew et al. (2004) create a joint model of verb sense and subcategorization preference by combining a bag of words WSD system with an unlexicalized PCFG parser, and show that this model delivers modest performance improvements for both tasks.

In this study, we attempt to improve verb sense disambiguation by leveraging syntactic analysis. We combine a probability model of subcategorization preference with a knowledge-based WSD algorithm, SSI-Dijkstra (Cuadros and Rigau, 2008).

2. SSI-Dijkstra

The WSD literature displays a growing trend towards exploring *knowledge-based* algorithms, meaning unsupervised methods predominantly based on the use of lexical resources, such as machine readable dictionaries. These methods are attractive in that they do not require sense-labelled training data, which is often difficult and expensive to obtain. *Graph-based* methods, a type of knowledge-based method, operate on *semantic networks* (graphs), whose nodes represent word senses; often a graph connectivity measure such as PageRank is then used to identify the "important" nodes in the graph, and these are taken as sense assignments. Algorithms of this type have recently attained state of the art performance on standard WSD evaluation metrics. A property common to these systems is that they use only lexical semantic knowledge, and are ignorant of syntax and word order; this attribute is useful for our purposes, since such an algorithm should provide an ideal theatre for examining the marginal effect of adding subcategorization analysis of verbs.

SSI-Dijkstra is a knowledge-based algorithm which operates using a large directed graph. The graph is built directly from WordNet, such that nodes in the graph are WordNet synsets, and edges between these nodes represent semantic relations listed in WordNet; we generate inverse edges

as needed so that all relations are symmetric. Additionally, we add edges representing semantic relations taken from a number of other sources: eXtended WordNet[1], WordNet Domains[2], KnowNet[3] and WordNet++[4]. For these other semantic relations, we similarly create inverse edges as needed to ensure that nodes are linked symmetrically.

The graph can be used to give a measure of *semantic distance*, which we define to be the shortest path through the graph between two WordNet synsets; this distance can be efficiently computed using Dijkstra's algorithm.

The SSI-Dijkstra algorithm proceeds in an iterative, greedy fashion: it starts with a *semantic context* C consisting of the set of WordNet senses representing the monosemous words in the current sentence; polysemous words in the sentence are placed into a *pending set* P to be disambiguated. On each iteration, the algorithm computes, for each sense s of each word to be disambiguated, the total semantic distance from all the senses in the semantic context to the sense s. The sense having the least distance to the semantic context is then chosen and added to the context; the word (or words) for which it is a possible sense are then marked as being disambiguated and are removed from the set P. For disambiguating running text, we also include in the context C those words from the previous sentence which have already been disambiguated.

We introduce a novel extension of the SSI-Dijkstra algorithm, in that we assign weights to the graph edges. In our variant, every edge ending at a node n (representing a WordNet synset s) has a weight given by $\frac{1}{P(s)}$, the inverse of the prior probability of seeing an instance of the synset s in a balanced corpus. We estimate this probability distribution over synsets by counting word senses in SemCor (Miller et al., 1993) and smoothing the counts with Good-Turing estimation.

We use this same approach to integrate syntactic information in the form of our probabilistic model of subcategorization preference into the algorithm. Under this scheme, the edges which end at a node n (representing a verb sense v in the current sentence, with lemma l and subcategorization frame f) have a weight of $\frac{1}{P(v|l,f)}$, the inverse of the posterior probability of the sense v given the lemma and SCF.

3. Subcategorization Frames

To build our model, we use a subset of the subcategorization frames given in (Andrew et al., 2004). Our 12 frames, shown in Table 1, are implemented using `tgrep` search strings; our SCFs undo passivization but do not analyse verb particles as arguments, since phrasal verbs are already analysed as multi-word expressions in SemCor. A verb instance in a parse tree can be categorised for SCF by finding the first `tgrep` string which matches.

We build our model from SemCor, which is tagged for word sense but does not contain parse trees; therefore, we parsed SemCor with the Stanford Parser (Klein and Manning, 2003)[5], an unlexicalized PCFG parser[6]. This gives us 81,461 verb instances tagged for verb sense and subcategorization, giving counts which allow us to estimate a joint probability model over verb sense and subcategorization. For illustration, Table 2 shows the counts obtained for two senses of the verb *appear*: sense 1 (to "give a certain impression or have a certain outward aspect") selects strongly for **VPto**, whereas sense 2 (to "come into sight or view") instead selects for **PP**.

To mitigate problems caused by sparse data, we construct two related "back-off" distributions: one which counts co-occurrences of VerbNet class and subcategorization frame,

SCF	Example
$\emptyset$	Her little brown face *wrinkled up*.
NP	They *polished* [the windshield].
PP	If he can *bounce back* [with one of those years], the club will be better off.
NP PP	A light-colored roof will *reduce* [sun heat] [by 50 per cent].
NP NP	There would be time enough to *pay* [the devil] [his due].
VPto	Dwellers thereabouts *preferred* [to get their pies at the bakery].
VPing	The driver *started* [zigzagging the truck].
S for to	One of the agreements *calls* [for the New Eastwick Corp.] [to purchase a 1311 acre tract for $12192865].
NP SBAR	An officer had *told* [him] [that in case of attack he was not to open fire].
NP VPing	Rachel had *seen* [me] [watching the young man].
NP VPto	You can *use* [heat-absorbing glass] [to stop the sun].
Other	Gene Marshall has *announced* [that the garden will open to members].

Table 1: Subcategorization frames used in this study.

SCF	appear-1	appear-2
VPto	62	0
$\emptyset$	12	19
Other	12	5
PP	9	54
S for to	5	1
NP	3	3
NP PP	1	1
VPing	1	1

Table 2: Counts of verb sense-SCF pairs in SemCor.

[1] http://xwn.hlt.utdallas.edu

[2] http://wndomains.itc.it

[3] http://adimen.si.ehu.es/web/KnowNet, we use the KnowNet-10 version here.

[4] http://lcl.uniroma1.it/wordnetplusplus

[5] http://nlp.stanford.edu/software/lex-parser.shtml

[6] We note that part of the Brown corpus is available in parsed form in the Penn Treebank; however, this material overlaps with less than half of SemCor, which we considered unacceptable, given the problems with data sparseness that SemCor's small size already creates.

and one for verb lemma and SCF. We apply Good-Turing smoothing to all three distributions and convert them into probability models of subcategorization frame conditional on verb sense. For the VerbNet model, we define a function $K(v_s)$ which gives the set of top-level VerbNet classes k that include the WordNet verb sense v_s as a member. We can then define the conditional probability of a SCF f given a verb sense v_s using the VerbNet model by averaging the distributions of all VerbNet classes that the sense belongs to:

$$P_V(f|v_s) = \frac{1}{|K(v_s)|} \sum_{\{k \in K(v_s)\}} P(f|k)$$

We then combine the three models using linear interpolation. Here P is our final model, P_{MLE} is the original maximum likelihood model of verb sense and SCF, P_V is the model of VerbNet class and SCF, and P_L is the model of verb lemma and SCF. Interpolation is performed between the verb sense/SCF model and the VerbNet class/SCF model by preference; if a given verb sense is not found in VerbNet, then we use the verb lemma/SCF model. For a given SCF f and and a verb sense v_s with corresponding lemma l:

$$P(f|v_s) = \begin{cases} \alpha P_{MLE}(f|v_s) + (1-\alpha) P_V(f|v_s) \\ \quad \text{if } v_s \text{ is in VerbNet} \\ \beta P_{MLE}(f|v_s) + (1-\beta) P_L(f|l) \\ \quad \text{otherwise} \end{cases}$$

The interpolation parameters used were $\alpha = 0.5$, $\beta = 0.55$; these values were estimated by optimization on SemCor using 10-fold cross-validation. Finally, this model is used to give the posterior probability of a verb sense v_s given a lemma l and SCF f[7]:

$$P(v_s|l,f) = \frac{P(v_s,l,f)}{P(l,f)} = \frac{P(f|v_s)}{P(l,f)} P(v_s)$$

Note that this model can be used by itself to perform verb sense disambiguation on parsed text.

4. Evaluation

We evaluate our version of SSI-Dijkstra on the Senseval-2 English all words task[8]; note that these test data are parsed to allow the use of the SCF model. Results are shown in Table 3. This table also shows the random and *most frequent sense* (MFS) baselines, which are typically taken as lower and upper bounds for unsupervised systems[9]. On the Senseval-2 task, systems were required to POS-tag and lemmatize words by themselves; the random baseline figures given here assume an oracle that always knows the correct POS tag and lemma, and thus have a small advantage over participating systems. We compute our MFS baseline by always taking the first sense of a word listed in WordNet; note that word senses in WordNet are ordered by their frequency in SemCor (as SemCor was created with WordNet 1.6, this frequency information is only available for word senses that were present in that version). For comparison, the table gives the results of the best supervised system at the time of the competition, SMUaw, and the best unsupervised system, UNED-AW. The table also lists results from recent graph-based WSD methods: Mih05 (Mihalcea, 2005), Sinha07 (Sinha and Mihalcea, 2007), Tsatsa07 (Tsatsaronis et al., 2007), as well as results from the current best graph-based WSD algorithm, Agirre and Soroa's (2009) word-to-word Personalized PageRank (Agi09).

The original SSI-Dijkstra algorithm (without edge weighting) performs better than the random baseline and has good coverage. Adding the edge weighting scheme results in better disambiguation for nouns, adjectives, and overall score (statistically significant at the $p < 0.001$ level)[10], and brings the system's performance close to the MFS baseline. By itself, the subcategorization model performs similarly to our edge-weighted method on verbs, and slightly better than the MFS baseline. On verbs, the SCF model is competitive with state of the art unsupervised algorithms; considering this, we believe that analysing subcategorization represents a promising avenue for future WSD research.

Integrating the SCF model with SSI-Dijkstra improves results on verb disambiguation beyond the levels observed for either SSI-Dijkstra or the SCF model in isolation (the improvement is, however, not statistically significant on this study).

The significance of this last effect is twofold. Firstly, it provides empirical support for the hypothesis that subcategorization and verb sense are related. Secondly, we show that verb sense disambiguation can be improved by leveraging syntactic analysis; here, we use only the output of a statistical constituency parser, and record performance that is competitive with state of the art unsupervised algorithms. Our study would furthermore suggest that syntactic and semantic information are complementary for the sense disambiguation task.

5. Conclusion

We have presented a simple method for estimating a joint probability distribution on verb sense and subcategorization and shown that this model is capable of disambiguating verbs at a level comparable to the first sense baseline. We have evaluated a simple wide-coverage WSD algorithm, SSI-Dijkstra, on a commonly used disambiguation competition, allowing direct comparison of this algorithm to recently published conceptually similar graph-based methods. On the Senseval-2 all words task, our overall F-score of 60.0% beats the best unsupervised system that participated in the competition, and would have put SSI-Dijkstra

[7] Note that verb sense completely determines lemma, giving $P(v_s, l) = P(v_s)$, and so $P(v_s, l, f) = P(v_s, f)$.

[8] We point out that the Senseval-2 task was mapped to WordNet 2.1 for this evaluation; since a small fraction of tagged words cannot be mapped due to changes in the WordNet inventory, the results obtained are slightly distorted by this process. Using the first sense baseline as an indicator, we believe that the values given here for the SSI-Dijkstra system are overestimated by about 2%.

[9] The addition of word sense frequency information to our algorithm means that it is not strictly unsupervised; the edge weights tend to bias the output towards the most frequent sense, and so our method might best be described as a *hybrid* system.

[10] Significance tests for comparing algorithm performance reported in this study use the paired McNemar test.

System	All	Noun	Verb	Adj	Adv
Random baseline	42.0	45.6	21.9	45.8	60.1
MFS baseline	60.1	71.2	39.0	61.1	75.4
SMUaw	68.6	78.0	52.9	69.9	81.7
UNED-AW	55.2	60.0	38.5	60.2	74.7
Mih05	54.2	57.5	36.5	56.7	70.9
Sinha07	57.6	66.2	34.1	61.8	60.4
Tsatsa07	49.3	—	—	—	—
Agi09	58.6	70.4	38.9	58.3	70.1
SCF Model only	14.0	0.0	40.3	0.0	0.0
SSI-Dijkstra	54.4	60.4	38.6	60.0	68.1
SSI-Dijkstra + edge weighting	59.3	67.5	41.3	67.1	67.7
SSI-Dijkstra + edge weighting + SCF	60.0	67.5	43.7	67.1	68.5

Table 3: F-score results on the Senseval-2 English all-words task.

fourth overall (out of 22 systems). We demonstrate that adding the subcategorization model to the SSI-Dijkstra algorithm can give an improvement to WSD performance; future work will investigate possible avenues for improving both the model and methods of integrating it with WSD systems.

6. References

Eneko Agirre and Aitor Soroa. 2009. Personalizing PageRank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 33–41.

Galen Andrew, Trond Grenager, and Christopher D. Manning. 2004. Verb sense and subcategorization: Using joint inference to improve performance on complementary tasks. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 150–157.

Jinying Chen and Martha Palmer. 2005. Towards robust high performance word sense disambiguation of English verbs using rich linguistic features. *Natural Language Processing–IJCNLP 2005*, pages 933–944.

Montse Cuadros and German Rigau. 2008. KnowNet: Building a large net of knowledge from the Web. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 161–168.

Hoa Trang Dang and Martha Palmer. 2005. The role of semantic roles in disambiguating verb senses. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 42–49.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 423–430.

Anna Korhonen and Judita Preiss. 2003. Improving subcategorization acquisition using word sense disambiguation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 48–55.

Rada Mihalcea. 2005. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 411–418.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *Proceedings of the Workshop on Human Language Technology*, pages 303–308. Princeton, NJ.

Douglas Roland and Daniel Jurafsky. 2002. Verb sense and verb subcategorization probabilities. In Paola Merlo and Suzanne Stevenson, editors, *The Lexical Basis of Sentence Processing: Formal, Computational, and Experimental Issues*, chapter 16. John Benjamins.

Ravi Sinha and Rada Mihalcea. 2007. Unsupervised graph-based word sense disambiguation using measures of word semantic similarity. In *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007)*, pages 363–369, Irvine, CA.

George Tsatsaronis, Michalis Vazirgiannis, and Ion Androutsopoulos. 2007. Word sense disambiguation with spreading activation networks generated from thesauri. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence (IJCAI)*, pages 1725–1730.