What's in a Preposition? Dimensions of Sense Disambiguation for an Interesting Word Class

Dirk Hovy, Stephen Tratz, and Eduard Hovy
Information Sciences Institute
University of Southern California
{dirkh, stratz, hovy}@isi.edu

Abstract

Choosing the right parameters for a word sense disambiguation task is critical to the success of the experiments. We explore this idea for prepositions, an often overlooked word class. We examine the parameters that must be considered in preposition disambiguation, namely context, features, and granularity. Doing so delivers an increased performance that significantly improves over two state-of-the-art systems, and shows potential for improving other word sense disambiguation tasks. We report accuracies of 91.8% and 84.8% for coarse and fine-grained preposition sense disambiguation, respectively.

1 Introduction

Ambiguity is one of the central topics in NLP. A substantial amount of work has been devoted to disambiguating prepositional attachment, words, and names. Prepositions, as with most other word types, are ambiguous. For example, the word *in* can assume both temporal ("in May") and spatial ("in the US") meanings, as well as others, less easily classifiable ("in that vein"). Prepositions typically have more senses than nouns or verbs (Litkowski and Hargraves, 2005), making them difficult to disambiguate.

Preposition sense disambiguation (PSD) has many potential uses. For example, due to the relational nature of prepositions, disambiguating their senses can help with all-word sense disambiguation. In machine translation, different senses of the same English preposition often correspond to different translations in the foreign language. Thus, disambiguating prepositions correctly may help improve translation quality.[1] Coarse-grained PSD can also be valuable for information extraction, where the sense acts as a label. In a recent study, Hwang et al. (2010) identified preposition related features, among them the coarse-grained PP labels used here, as the most informative feature in identifying caused-motion constructions. Understanding the constraints that hold for prepositional constructions could help improve PP attachment in parsing, one of the most frequent sources of parse errors.

Several papers have successfully addressed PSD with a variety of different approaches (Rudzicz and Mokhov, 2003; O'Hara and Wiebe, 2003; Ye and Baldwin, 2007; O'Hara and Wiebe, 2009; Tratz and Hovy, 2009). However, while it is often possible to increase accuracy by using a different classifier and/or more features, adding more features creates two problems: a) it can lead to overfitting, and b) while possibly improving accuracy, it is not always clear where this improvement comes from and which features are actually informative. While parameter studies exist for general word sense disambiguation (WSD) tasks (Yarowsky and Florian, 2002), and PSD accuracy has been steadily increasing, there has been no exploration of the parameters of prepositions to guide engineering decisions.

We go beyond simply improving accuracy to analyze various parameters in order to determine which ones are actually informative. We explore the different options for context and feature se-

[1] See (Chan et al., 2007) for the relevance of word sense disambiguation and (Chiang et al., 2009) for the role of prepositions in MT.

Coling 2008: Poster Volume, pages 454–462,
Beijing, August 2010

lection, the influence of different preprocessing methods, and different levels of sense granularity. Using the resulting parameters in a Maximum Entropy classifier, we are able to improve significantly over existing results. The general outline we present can potentially be extended to other word classes and improve WSD in general.

2 Related Work

Rudzicz and Mokhov (2003) use syntactic and lexical features from the governor and the preposition itself in coarse-grained PP classification with decision heuristics. They reach an average F-measure of 89% for four classes. This shows that using a very small context can be effective. However, they did not include the object of the preposition and used only lexical features for classification. Their results vary widely for the different classes.

O'Hara and Wiebe (2003) made use of a window size of five words and features from the Penn Treebank (PTB) (Marcus et al., 1993) and FrameNet (Baker et al., 1998) to classify prepositions. They show that using high level features, such as semantic roles, significantly aid disambiguation. They caution that using collocations and neighboring words indiscriminately may yield high accuracy, but has the risk of overfitting. O'Hara and Wiebe (2009) show comparisons of various semantic repositories as labels for PSD approaches. They also provide some results for PTB-based coarse-grained senses, using a five-word window for lexical and hypernym features in a decision tree classifier.

SemEval 2007 (Litkowski and Hargraves, 2007) included a task for fine-grained PSD (more than 290 senses). The best participating system, that of Ye and Baldwin (2007), extracted part-of-speech and WordNet (Fellbaum, 1998) features using a word window of seven words in a Maximum Entropy classifier. Tratz and Hovy (2009) present a higher-performing system using a set of 20 positions that are syntactically related to the preposition instead of a fixed window size.

Though using a variety of different extraction methods, contexts, and feature words, none of these approaches explores the optimal configurations for PSD.

3 Theoretical Background

The following parameters are applicable to other word classes as well. We will demonstrate their effectiveness for prepositions.

Analyzing the syntactic elements of prepositional phrases, one discovers three recurring elements that exhibit syntactic dependencies and define a prepositional phrase. The first one is the governing word (usually a noun, verb, or adjective)[2], the preposition itself, and the object of the preposition.

Prepositional phrases can be fronted ("*In May*, prices dropped by 5%"), so that the governor (in this case the verb "drop") occurs later in the sentence. Similarly, the object can be fronted (consider "*a dessert* to die for").

In the simplest version, we can do classification based only on the preposition and the governor or object alone.[3] Furthermore, directly neighboring words can influence the preposition, mostly two-word prepositions such as "out of" or "because of".

To extract the words discussed above, one can either employ a fixed window size, (which has to be large enough to capture the words), or select them based on heuristics or parsing information. The governor and object can be hard to extract if they are fronted, since they do not occur in their unusual positions relative to the preposition. While syntactically related words improve over fixed-window-size approaches (Tratz and Hovy, 2009), it is not clear which words contribute most. There should be an optimal context, i.e., the smallest set of words that achieves the best accuracy. It has to be large enough to capture all relevant information, but small enough to avoid noise words.[4] We surmise that earlier approaches were not utilizing that optimal context, but rather include a lot of noise.

Depending on the task, different levels of sense granularity may be used. Fewer senses increase the likelihood of correct classification, but may in-

[2]We will refer to the governing word, irrespective of class, as governor.

[3]Basing classification on the preposition alone is not feasible, because of the very polysemy we try to resolve.

[4]It is not obvious how much information a sister-PP can provide, or the subject of the superordinate clause.

correctly conflate prepositions. A finer granularity can help distinguish nuances and better fit the different contexts. However, it might suffer from sparse data.

4 Experimental Setup

We explore the different context types (fixed window size vs. selective), the influence of the words in that context, and the preprocessing method (heuristics vs. parsing) on both coarse and fine-grained disambiguation. We use a most-frequent-sense baseline. In addition, we compare to the state-of-the-art systems for both types of granularity (O'Hara and Wiebe, 2009; Tratz and Hovy, 2009). Their results show what has been achieved so far in terms of accuracy, and serve as a second measure for comparison beyond the baseline.

4.1 Model

We use the MALLET implementation (McCallum, 2002) of a Maximum Entropy classifier (Berger et al., 1996) to construct our models. This classifier was also used by two state-of-the-art systems (Ye and Baldwin, 2007; Tratz and Hovy, 2009). For fine-grained PSD, we train a separate model for each preposition due to the high number of possible classes for each individual preposition. For coarse-grained PSD, we use a single model for all prepositions, because they all share the same classes.

4.2 Data

We use two different data sets from existing resources for coarse and fine-grained PSD to make our results as comparable to previous work as possible.

For the coarse-grained disambiguation, we use data from the POS tagged version of the Wall Street Journal (WSJ) section of the Penn TreeBank. A subset of the prepositional phrases in this corpus is labelled with a set of seven classes: beneficial (BNF), direction (DIR), extent (EXT), location (LOC), manner (MNR), purpose (PRP), and temporal (TMP). We extract only those prepositions that head a PP labelled with such a class ($N = 35,917$). The distribution of classes is highly skewed (cf. Figure 1). We compare the results of this task to the findings of O'Hara and Wiebe (2009).

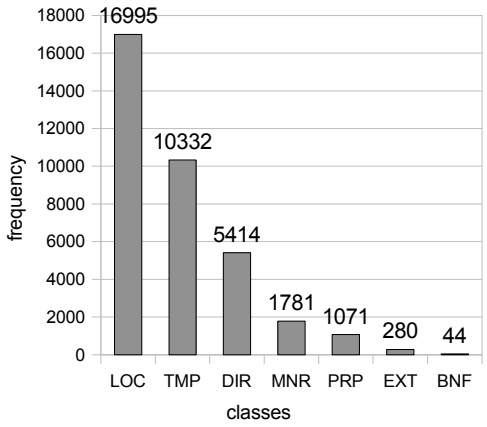

Figure 1: Distribution of Class Labels in the WSJ Section of the Penn TreeBank.

For the fine-grained task, we use data from the SemEval 2007 workshop (Litkowski and Hargraves, 2007), separate XML files for the 34 most frequent English prepositions, comprising $16,557$ training and 8096 test sentences, each instance containing one example of the respective preposition. Each preposition has between two and 25 senses (9.76 on average) as defined by The Preposition Project (Litkowski and Hargraves, 2005). We compare our results directly to the findings from Tratz and Hovy (2009). As in the original workshop task, we train and test on separate sets.

5 Results

In this section we show experimental results for the influence of word extraction method (parsing vs. POS-based heuristics), context, and feature selection on accuracy. Each section compares the results for both coarse and fine-grained granularity. Accuracy for the coarse-grained task is in all experiments higher than for the fine-grained one.

5.1 Word Extraction

In order to analyze the impact of the extraction method, we compare parsing versus POS-based heuristics for word extraction.

Both O'Hara and Wiebe (2009) and Tratz and Hovy (2009) use constituency parsers to preprocess the data. However, parsing accuracy varies,

and the problem of PP attachment ambiguity increases the likelihood of wrong extractions. This is especially troublesome in the present case, where we focus on prepositions.[5] We use the MALT parser (Nivre et al., 2007), a state-of-the-art dependency parser, to extract the governor and object.

The alternative is a POS-based heuristics approach. The only preprocessing step needed is POS tagging of the data, for which we used the system of Shen et al. (2007). We then use simple heuristics to locate the prepositions and their related words. In order to determine the governor in the absence of constituent phrases, we consider the possible governing noun, verb, and adjective. The object of the preposition is extracted as first noun phrase head to the right. This approach is faster than parsing, but has problems with long-range dependencies and fronting of the PP (e.g., the PP appearing earlier in the sentence than its governor).

extraction method	fine	coarse
MALT	84.4	94.0
Heuristics	84.8	90.9
MALT + Heuristics	84.8	91.8

Table 1: Accuracies (%) for Word-Extraction Using MALT Parser or Heuristics.

Interestingly, the extraction method does not significantly affect the final score for fine-grained PSD (see Table 1). The high score achieved when using the MALT parse for coarse-grained PSD can be explained by the fact that the parser was originally trained on that data set. The good results we see when using heuristics-based extraction only, however, means we can achieve high-accuracy PSD even without parsing.

5.2 Context

We compare the effects of fixed window size versus syntactically related words as context. Table 2 shows the results for the different types and sizes of contexts.[6]

[5]Rudzicz and Mokhov (2003) actually motivate their work as a means to achieve better PP attachment resolution.

[6]See also (Yarowsky and Florian, 2002) for experiments on the effect of varying window size for WSD.

Context	coarse	fine
2-word window	91.6	80.4
3-word window	92.0	81.4
4-word window	91.6	79.8
5-word window	91.0	78.7
Governor, prep	80.7	78.9
Prep, object	***94.2***	56.9
Governor, prep, object	94.0	***84.8***

Table 2: Accuracies (%) for Different Context Types and Sizes

The results show that the approach using both governor and object is the most accurate one. Of the fixed-window-size approaches, three words to either side works best. This does not necessarily reflect a general property of that window size, but can be explained by the fact that most governors and objects occur within this window size.[7] This distance can vary from corpus to corpus, so window size would have to be determined individually for each task. The difference between using governor and preposition versus preposition and object between coarse and fine-grained classification might reflect the annotation process: while Litkowski and Hargraves (2007) selected examples based on a search for governors[8], most annotators in the PTB may have based their decision of the PP label on the object that occurs in it. We conclude that syntactically related words present a better context for classification than fixed window sizes.

5.3 Features

Having established the context we want to use, we now turn to the details of extracting the feature words from that context.[9] Using higher-level features instead of lexical ones helps accounting for sparse training data (given an infinite amount of data, we would not need to take any higher-level

[7]Based on such statistics, O'Hara and Wiebe (2003) actually set their window size to 5.

[8]Personal communication.

[9]As one reviewer pointed out, these two dimensions are highly interrelated and influence each other. To examine the effects, we keep one dimension constant while varying the other.

features into account, since every case would be covered). Compare O'Hara and Wiebe (2009).

Following the prepocessing, we use a set of rules to select the feature words, and then generate feature values from them using a variety of feature-generating functions.[10] The word-selection rules are listed below.

Word-Selection Rules

- Governor from the MALT parse
- Object from the MALT parse
- Heuristically determined object of the preposition
- First verb to the left of the preposition
- First verb/noun/adjective to the left of the preposition
- Union of (First verb to the left, First verb/noun/adjective to the left)
- First word to the left

The feature-generating functions, many of which utilize WordNet (Fellbaum, 1998), are listed below. To conserve space, curly braces are used to represent multiple functions in a single line. The name of each feature is the combination of the word-selection rule and the output from the feature-generating function.

WordNet-based Features

- {Hypernyms, Synonyms} for {1st, all} sense(s) of the word
- All terms in the definitions ('glosses') of the word
- Lexicographer file names for the word
- Lists of all link types (e.g., meronym links) associated with the word
- Part-of-speech indicators for the existence of NN/VB/JJ/RB entries for the word
- All sentence frames for the word
- All {part, member, substance}-of holonyms for the word
- All sentence frames for the word

Other Features

- Indicator that the word-finding rule found a word
- Capitalization indicator
- {Lemma, surface form} of the word
- Part-of-speech tag for the word
- General POS tag for the word (e.g. NNS → NN, VBZ → VB)
- The {first, last} {two, three} letters of each word
- Indicators for suffix types (e.g., de-adjectival, de-nominal [non]agentive, de-verbal [non]agentive)
- Indicators for a wide variety of other affixes including those related to degree, number, order, etc. (e.g., *ultra-, poly-, post-*)
- Roget's Thesaurus divisions for the word

To establish the impact of each feature word on the outcome, we use leave-one-out and only-one evaluation.[11] The results can be found in Table 3. A word that does not perform well as the only attribute may still be important in conjunction with others. Conversely, leaving out a word may not hurt performance, despite being a good single attribute.

	coarse		fine	
Word	**LOO**	**Only**	**LOO**	**Only**
MALT governor	92.1	80.1	84.3	78.9
MALT object	93.4	94.2	84.9	56.3
Heuristics VB to left	92.0	77.9	85.0	62.1
Heur. NN/VB/ADJ to left	92.1	78.7	84.3	78.5
Heur. Governor Union	92.1	78.4	84.5	81.0
Heuristics word to left	92.0	78.8	84.4	77.2
Heuristics object	91.9	93.0	84.9	56.8
none	91.8	–	84.8	–

Table 3: Accuracies (%) for Leave-One-Out (LOO) and Only-One Word-Extraction-Rule Evaluation. *none* includes all words and serves for comparison. Important words reduce accuracy for LOO, but rank high when used as only rule.

Independent of the extraction method (MALT parser or POS-based heuristics), the governor is the most informative word. Combining several heuristics to locate the governor is the best single feature for fine-grained classification. The rule looking only for a governing verb fails to account

[10] Some words may be selected by multiple word-selection rules. For example, the governor of the preposition may be identified by the *Governor from MALT parse* rule, *first noun/verb/adjective to left*, and the *first word to the left* rule.

[11] Since the feature words are not independent of one another, neither of the two measures is decisive on its own.

	fine		coarse	
Prep	**Total**	**Acc**	**Total**	**Acc**
aboard	–	–	6	100.0
about	364	94.0	5	80.0
above	23	69.6	78	65.4
across	151	96.7	87	79.3
after	53	79.2	841	92.5
against	92	92.4	16	43.8
along	173	96.0	45	71.1
alongside	–	–	5	80.0
amid	–	–	58	70.7
among	50	80.0	358	93.9
amongst	–	–	1	0.0
around	155	69.0	107	86.0
as	84	100.0	232	84.5
astride	–	–	2	50.0
at	367	86.4	3078	92.0
atop	–	–	5	100.0
because	–	–	420	91.7
before	20	90.0	384	83.3
behind	68	77.9	65	87.7
below	–	–	94	71.3
beneath	28	78.6	11	72.7
beside	29	100.0	4	100.0
besides	–	–	1	0.0
between	102	94.1	98	84.7
beyond	–	–	45	64.4
by	248	88.3	1341	87.5
down	153	81.7	16	56.2
during	39	87.2	547	92.1
except	–	–	1	0.0
for	478	82.4	1455	84.5
from	578	85.5	1712	90.5
in	688	77.0	15706	95.0
inside	38	73.7	24	91.7
into	297	86.2	415	80.0
like	125	90.4	53	47.2
near	–	–	74	93.2
nearest	–	–	1	0.0
next	–	–	7	71.4
of	1478	87.9	71	64.8
off	76	84.2	28	75.0
on	441	81.4	2287	90.8
onto	58	91.4	15	53.3
out	–	–	90	68.9
outside	–	–	62	90.3
over	98	79.6	417	89.4
past	–	–	6	83.3
per	–	–	3	100.0
round	82	65.9	–	–
since	–	–	449	94.4
than	–	–	2	0.0
through	208	48.1	364	69.0
throughout	–	–	62	93.5
till	–	–	3	100.0
to	572	89.7	3166	97.5
toward	–	–	55	65.5
towards	102	97.1	2	100.0
under	–	–	604	91.4
underneath	–	–	2	50.0
until	–	–	208	94.2
up	–	–	20	75.0
upon	–	–	23	73.9
via	–	–	22	40.9
whether	–	–	1	100.0
while	–	–	3	33.3
with	578	84.4	272	69.5
within	–	–	213	96.2
without	–	–	69	63.8
Overall	8096	84.8	**35917**	**91.8**

Table 4: Accuracies (%) for Coarse and Fine-Grained PSD, Using MALT and Heuristics. Sorted by preposition.

for noun governors, which consequently leads to a slight improvement when left out.

Curiously, the word directly to the left is a better single feature than the object (for fine-grained classification). Leaving either of them out increases accuracy, which implies that their information can be covered by other words.

	Most Frequent Sense			O'Hara/Wiebe 2009			10-fold CV		
Class	**prec**	**rec**	**f1**	**prec**	**rec**	**f1**	**prec**	**rec**	**f1**
LOC	71.8	97.4	82.6	90.8	93.2	92.0	94.7	96.4	***95.6***
TMP	77.5	39.4	52.3	84.5	85.2	84.8	94.6	94.6	***94.6***
DIR	91.6	94.2	92.8	95.6	96.5	***96.1***	94.6	94.5	94.5
MNR	69.9	43.2	53.4	82.6	55.8	66.1	83.3	75.0	***78.9***
PRP	78.2	48.8	60.1	79.3	70.1	74.4	90.6	83.8	***87.1***
EXT	0.0	0.0	0.0	81.7	84.6	82.9	87.5	82.1	***84.7***
BNF	0.0	0.0	0.0	–	–	–	75.0	34.1	***46.9***

Table 5: Precision, Recall and F1 Results (%) for Coarse-Grained Classification. Comparison to O'Hara and Wiebe (2009). Classes ordered by frequency

5.4 Comparison with Related Work

To situate our experimental results within the body of work on PSD, we compare them to both a most-frequent-sense baseline and existing work for both granularities (see Table 6). The results use a syntactically selective context of preposition, governor, object, and word to the left as determined by combined extraction information (POS tagging and parsing).

	coarse	**fine**
Baseline	75.8	39.6
Related Work	89.3*	78.3**
Our system	93.9	84.8

Table 6: Accuracies (%) for Different Classifications. Comparison with O'Hara and Wiebe (2009)*, and Tratz and Hovy (2009)**.

Our system easily exceeds the baseline for both coarse and fine-grained PSD (see Table 6). Comparison with related work shows that we achieve an improvement of 6.5% over Tratz and Hovy (2009), which is significant at $p < .0001$, and of 4.5% over O'Hara and Wiebe (2009), which is significant at $p < .0001$.

A detailed overview over all prepositions for frequencies and accuracies of both coarse and fine-grained PSD can be found in Table 4.

In addition to overall accuracy, O'Hara and Wiebe (2009) also measure precision, recall and F-measure for the different classes. They omitted BNF because it is so infrequent. Due to different training data and models, the two systems are not strictly comparable, yet they provide a sense of the general task difficulty. See Table 5. We note that both systems perform better than the most-frequent-sense baseline. DIR is reliably classified using the baseline, while EXT and BNF are never selected for any preposition. Our method adds considerably to the scores for most classes. The low score for BNF is mainly due to the low number of instances in the data, which is why it was excluded by O'Hara and Wiebe (2009).

6 Conclusion

To get maximal accuracy in disambiguating prepositions—and also other word classes—one needs to consider context, features, and granularity. We presented an evaluation of these parameters for preposition sense disambiguation (PSD).

We find that selective context is better than fixed window size. Within the context for prepositions, the governor (head of the NP or VP governing the preposition), the object of the preposition (i.e., head of the NP to the right), and the word directly to the left of the preposition have the highest influence.[12] This corroborates the linguistic intuition that close mutual constraints hold between the elements of the PP. Each word syntactically and semantically restricts the choice of the other elements. Combining different extraction methods (POS-based heuristics and dependency parsing) works better than either one in isolation, though high accuracy can be achieved just using heuristics. The impact of context and features varies somewhat for different granularities.

[12] These will likely differ for other word classes.

Not surprisingly, we see higher scores for coarser granularity than for the more fine-grained one.

We measured success in accuracy, precision, recall, and F-measure, and compared our results to a most-frequent-sense baseline and existing work. We were able to improve over state-of-the-art systems in both coarse and fine-grained PSD, achieving accuracies of 91.8% and 84.8% respectively.

Acknowledgements

The authors would like to thank Steve DeNeefe, Victoria Fossum, and Zornitsa Kozareva for comments and suggestions. StephenTratz is supported by a National Defense Science and Engineering fellowship.

References

Baker, C.F., C.J. Fillmore, and J.B. Lowe. 1998. The Berkeley FrameNet Project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics Morristown, NJ, USA.

Berger, A.L., V.J. Della Pietra, and S.A. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Chan, Y.S., H.T. Ng, and D. Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Annual Meeting – Association For Computational Linguistics*, volume 45, pages 33–40.

Chiang, D., K. Knight, and W. Wang. 2009. 11,001 new features for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 218–226, Boulder, Colorado, June. Association for Computational Linguistics.

Fellbaum, C. 1998. *WordNet: an electronic lexical database*. MIT Press USA.

Hwang, J. D., R. D. Nielsen, and M. Palmer. 2010. Towards a domain independent semantics: Enhancing semantic representation with construction grammar. In *Proceedings of the NAACL HLT Workshop on Extracting and Using Constructions in Computational Linguistics*, pages 1–8, Los Angeles, California, June. Association for Computational Linguistics.

Litkowski, K. and O. Hargraves. 2005. The preposition project. *ACL-SIGSEM Workshop on "The Linguistic Dimensions of Prepositions and Their Use in Computational Linguistic Formalisms and Applications"*, pages 171–179.

Litkowski, K. and O. Hargraves. 2007. SemEval-2007 Task 06: Word-Sense Disambiguation of Prepositions. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, Prague, Czech Republic.

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

McCallum, A.K. 2002. MALLET: A Machine Learning for Language Toolkit. 2002. http://mallet. cs. umass. edu.

Nivre, J., J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.

O'Hara, T. and J. Wiebe. 2003. Preposition semantic classification via Penn Treebank and FrameNet. In *Proceedings of CoNLL*, pages 79–86.

O'Hara, T. and J. Wiebe. 2009. Exploiting semantic role resources for preposition disambiguation. *Computational Linguistics*, 35(2):151–184.

Rudzicz, F. and S. A. Mokhov. 2003. Towards a heuristic categorization of prepositional phrases in english with wordnet. Technical report, Cornell University, arxiv1.library.cornell.edu/abs/1002.1095-?context=cs.

Shen, L., G. Satta, and A. Joshi. 2007. Guided learning for bidirectional sequence classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, volume 45, pages 760–767.

Tratz, S. and D. Hovy. 2009. Disambiguation of preposition sense using linguistically motivated features. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Student Research Workshop and Doctoral Consortium*, pages 96–100, Boulder, Colorado, June. Association for Computational Linguistics.

Yarowsky, D. and R. Florian. 2002. Evaluating sense disambiguation across diverse parameter spaces. *Natural Language Engineering*, 8(4):293–310.

Ye, P. and T. Baldwin. 2007. MELB-YB: Preposition Sense Disambiguation Using Rich Semantic Features. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, Prague, Czech Republic.