Improving Parsing and PP attachment Performance with Sense Information

Eneko Agirre
IXA NLP Group
University of the Basque Country
Donostia, Basque Country
e.agirre@ehu.es

Timothy Baldwin
LT Group, CSSE
University of Melbourne
Victoria 3010 Australia
tim@csse.unimelb.edu.au

David Martinez
LT Group, CSSE
University of Melbourne
Victoria 3010 Australia
davidm@csse.unimelb.edu.au

Abstract

To date, parsers have made limited use of semantic information, but there is evidence to suggest that semantic features can enhance parse disambiguation. This paper shows that semantic classes help to obtain significant improvement in both parsing and PP attachment tasks. We devise a gold-standard sense- and parse tree-annotated dataset based on the intersection of the Penn Treebank and SemCor, and experiment with different approaches to both semantic representation and disambiguation. For the Bikel parser, we achieved a maximal error reduction rate over the baseline parser of 6.9% and 20.5%, for parsing and PP-attachment respectively, using an unsupervised WSD strategy. This demonstrates that word sense information can indeed enhance the performance of syntactic disambiguation.

1 Introduction

Traditionally, parse disambiguation has relied on structural features extracted from syntactic parse trees, and made only limited use of semantic information. There is both empirical evidence and linguistic intuition to indicate that semantic features can enhance parse disambiguation performance, however. For example, a number of different parsers have been shown to benefit from lexicalisation, that is, the conditioning of structural features on the lexical head of the given constituent (Magerman, 1995; Collins, 1996; Charniak, 1997; Charniak, 2000; Collins, 2003). As an example of lexicalisation, we may observe in our training data that *knife* often occurs as the manner adjunct of *open* in prepositional phrases headed by *with* (c.f. *open with a knife*), which would provide strong evidence for *with (a) knife* attaching to *open* and not *box* in *open the box with a knife*. It would not, however, provide any insight into the correct attachment of *with scissors* in *open the box with scissors*, as the disambiguation model would not be able to predict that *knife* and *scissors* are semantically similar and thus likely to have the same attachment preferences.

In order to deal with this limitation, we propose to integrate directly the semantic classes of words into the process of training the parser. This is done by substituting the original words with semantic codes that reflect semantic classes. For example, in the above example we could substitute both *knife* and *scissors* with the semantic class TOOL, thus relating the training and test instances directly. We explore several models for semantic representation, based around WordNet (Fellbaum, 1998).

Our approach to exploring the impact of lexical semantics on parsing performance is to take two state-of-the-art statistical treebank parsers and preprocess the inputs variously. This simple method allows us to incorporate semantic information into the parser without having to reimplement a full statistical parser, and also allows for maximum comparability with existing results in the treebank parsing community. We test the parsers over both a PP attachment and full parsing task.

In experimenting with different semantic representations, we require some strategy to disambiguate the semantic class of polysemous words in context (e.g. determining for each instance of *crane* whether it refers to an animal or a lifting device). We explore a number of disambiguation strategies, including the use of hand-annotated (gold-standard) senses, the

Proceedings of ACL-08: HLT, pages 317–325,
Columbus, Ohio, USA, June 2008.

use of the most frequent sense, and an unsupervised word sense disambiguation (WSD) system.

This paper shows that semantic classes help to obtain significant improvements for both PP attachment and parsing. We attain a 20.5% error reduction for PP attachment, and 6.9% for parsing. These results are achieved using most frequent sense information, which surprisingly outperforms both gold-standard senses and automatic WSD.

The results are notable in demonstrating that very simple preprocessing of the parser input facilitates significant improvements in parser performance. We provide the first definitive results that word sense information can enhance Penn Treebank parser performance, building on earlier results of Bikel (2000) and Xiong et al. (2005). Given our simple procedure for incorporating lexical semantics into the parsing process, our hope is that this research will open the door to further gains using more sophisticated parsing models and richer semantic options.

2 Background

This research is focused on applying lexical semantics in parsing and PP attachment tasks. Below, we outline these tasks.

Parsing

As our baseline parsers, we use two state-of-the-art lexicalised parsing models, namely the Bikel parser (Bikel, 2004) and Charniak parser (Charniak, 2000). While a detailed description of the respective parsing models is beyond the scope of this paper, it is worth noting that both parsers induce a context free grammar as well as a generative parsing model from a training set of parse trees, and use a development set to tune internal parameters. Traditionally, the two parsers have been trained and evaluated over the WSJ portion of the Penn Treebank (PTB: Marcus et al. (1993)). We diverge from this norm in focusing exclusively on a sense-annotated subset of the Brown Corpus portion of the Penn Treebank, in order to investigate the upper bound performance of the models given gold-standard sense information.

PP attachment in a parsing context

Prepositional phrase attachment (PP attachment) is the problem of determining the correct attachment site for a PP, conventionally in the form of the noun or verb in a V NP PP structure (Ratnaparkhi et al., 1994; Mitchell, 2004). For instance, in *I ate a pizza with anchovies*, the PP *with anchovies* could attach either to the verb (c.f. *ate with anchovies*) or to the noun (c.f. *pizza with anchovies*), of which the noun is the correct attachment site. With *I ate a pizza with friends*, on the other hand, the verb is the correct attachment site. PP attachment is a structural ambiguity problem, and as such, a subproblem of parsing.

Traditionally the so-called RRR data (Ratnaparkhi et al., 1994) has been used to evaluate PP attachment algorithms. RRR consists of 20,081 training and 3,097 test quadruples of the form `(v,n1,p,n2)`, where the attachment decision is either `v` or `n1`. The best published results over RRR are those of Stetina and Nagao (1997), who employ WordNet sense predictions from an unsupervised WSD method within a decision tree classifier. Their work is particularly inspiring in that it significantly outperformed the plethora of lexicalised probabilistic models that had been proposed to that point, and has not been beaten in later attempts.

In a recent paper, Atterer and Schütze (2007) criticised the RRR dataset because it assumes that an oracle parser provides the two hypothesised structures to choose between. This is needed to derive the fact that there are two possible attachment sites, as well as information about the lexical phrases, which are typically extracted heuristically from gold standard parses. Atterer and Schütze argue that the only meaningful setting for PP attachment is within a parser, and go on to demonstrate that in a parser setting, the Bikel parser is competitive with the best-performing dedicated PP attachment methods. Any improvement in PP attachment performance over the baseline Bikel parser thus represents an advancement in state-of-the-art performance.

That we specifically present results for PP attachment in a parsing context is a combination of us supporting the new research direction for PP attachment established by Atterer and Schütze, and us wishing to reinforce the findings of Stetina and Nagao that word sense information significantly enhances PP attachment performance in this new setting.

Lexical semantics in parsing

There have been a number of attempts to incorporate word sense information into parsing tasks. The

most closely related research is that of Bikel (2000), who merged the Brown portion of the Penn Treebank with SemCor (similarly to our approach in Section 4.1), and used this as the basis for evaluation of a generative bilexical model for joint WSD and parsing. He evaluated his proposed model in a parsing context both with and without WordNet-based sense information, and found that the introduction of sense information either had no impact or degraded parse performance.

The only successful applications of word sense information to parsing that we are aware of are Xiong et al. (2005) and Fujita et al. (2007). Xiong et al. (2005) experimented with first-sense and hypernym features from HowNet and CiLin (both WordNets for Chinese) in a generative parse model applied to the Chinese Penn Treebank. The combination of word sense and first-level hypernyms produced a significant improvement over their basic model. Fujita et al. (2007) extended this work in implementing a discriminative parse selection model incorporating word sense information mapped onto upper-level ontologies of differing depths. Based on gold-standard sense information, they achieved large-scale improvements over a basic parse selection model in the context of the Hinoki treebank.

Other notable examples of the successful incorporation of lexical semantics into parsing, not through word sense information but indirectly via selectional preferences, are Dowding et al. (1994) and Hektoen (1997). For a broader review of WSD in NLP applications, see Resnik (2006).

3 Integrating Semantics into Parsing

Our approach to providing the parsers with sense information is to make available the semantic denotation of each word in the form of a semantic class. This is done simply by substituting the original words with semantic codes. For example, in the earlier example of *open with a knife* we could substitute both *knife* and *scissors* with the class TOOL, and thus directly facilitate semantic generalisation within the parser. There are three main aspects that we have to consider in this process: (i) the semantic representation, (ii) semantic disambiguation, and (iii) morphology.

There are many ways to represent semantic relationships between words. In this research we opt for a class-based representation that will map semantically-related words into a common semantic category. Our choice for this work was the WordNet 2.1 lexical database, in which synonyms are grouped into synsets, which are then linked via an IS-A hierarchy. WordNet contains other types of relations such as meronymy, but we did not use them in this research. With any lexical semantic resource, we have to be careful to choose the appropriate level of granularity for a given task: if we limit ourselves to synsets we will not be able to capture broader generalisations, such as the one between *knife* and *scissors*;[1] on the other hand by grouping words related at a higher level in the hierarchy we could find that we make overly coarse groupings (e.g. *mallet*, *square* and *steel-wool pad* are also descendants of TOOL in WordNet, none of which would conventionally be used as the manner adjunct of *cut*). We will test different levels of granularity in this work.

The second problem we face is semantic disambiguation. The more fine-grained our semantic representation, the higher the average polysemy and the greater the need to distinguish between these senses. For instance, if we find the word *crane* in a context such as *demolish a house with the crane*, the ability to discern that this corresponds to the DEVICE and not ANIMAL sense of word will allow us to avoid erroneous generalisations. This problem of identifying the correct sense of a word in context is known as word sense disambiguation (WSD: Agirre and Edmonds (2006)). Disambiguating each word relative to its context of use becomes increasingly difficult for fine-grained representations (Palmer et al., 2006). We experiment with different ways of tackling WSD, using both gold-standard data and automatic methods.

Finally, when substituting words with semantic tags we have to decide how to treat different word forms of a given lemma. In the case of English, this pertains most notably to verb inflection and noun number, a distinction which we lose if we opt to map all word forms onto semantic classes. For our current purposes we choose to substitute all word

[1] In WordNet 2.1, *knife* and *scissors* are sister synsets, both of which have TOOL as their 4th hypernym. Only by mapping them onto their 1st hypernym or higher would we be able to capture the semantic generalisation alluded to above.

forms, but we plan to look at alternative representations in the future.

4 Experimental setting

We evaluate the performance of our approach in two settings: (1) full parsing, and (2) PP attachment within a full parsing context. Below, we outline the dataset used in this research and the parser evaluation methodology, explain the methodology used to perform PP attachment, present the different options for semantic representation, and finally detail the disambiguation methods.

4.1 Dataset and parser evaluation

One of the main requirements for our dataset is the availability of gold-standard sense and parse tree annotations. The gold-standard sense annotations allow us to perform upper bound evaluation of the relative impact of a given semantic representation on parsing and PP attachment performance, to contrast with the performance in more realistic semantic disambiguation settings. The gold-standard parse tree annotations are required in order to carry out evaluation of parser and PP attachment performance.

The only publicly-available resource with these two characteristics at the time of this work was the subset of the Brown Corpus that is included in both SemCor (Landes et al., 1998) and the Penn Treebank (PTB).[2] This provided the basis of our dataset. After sentence- and word-aligning the SemCor and PTB data (discarding sentences where there was a difference in tokenisation), we were left with a total of 8,669 sentences containing 151,928 words. Note that this dataset is smaller than the one described by Bikel (2000) in a similar exercise, the reason being our simple and conservative approach taken when merging the resources.

We relied on this dataset alone for all the experiments in this paper. In order to maximise reproducibility and encourage further experimentation in the direction pioneered in this research, we partitioned the data into 3 sets: 80% training, 10% development and 10% test data. This dataset is available on request to the research community.

[2]OntoNotes (Hovy et al., 2006) includes large-scale treebank and (selective) sense data, which we plan to use for future experiments when it becomes fully available.

We evaluate the parsers via labelled bracketing recall ($\mathcal{R}$), precision ($\mathcal{P}$) and F-score ($\mathcal{F}_1$). We use Bikel's randomized parsing evaluation comparator[3] (with $p < 0.05$ throughout) to test the statistical significance of the results using word sense information, relative to the respective baseline parser using only lexical features.

4.2 PP attachment task

Following Atterer and Schütze (2007), we wrote a script that, given a parse tree, identifies instances of PP attachment ambiguity and outputs the `(v,n1,p,n2)` quadruple involved and the attachment decision. This extraction system uses Collins' rules (based on TREEP (Chiang and Bikel, 2002)) to locate the heads of phrases. Over the combined gold-standard parsing dataset, our script extracted a total of 2,541 PP attachment quadruples. As with the parsing data, we partitioned the data into 3 sets: 80% training, 10% development and 10% test data. Once again, this dataset and the script used to extract the quadruples are available on request to the research community.

In order to evaluate the PP attachment performance of a parser, we run our extraction script over the parser output in the same manner as for the gold-standard data, and compare the extracted quadruples to the gold-standard ones. Note that there is no guarantee of agreement in the quadruple membership between the extraction script and the gold standard, as the parser may have produced a parse which is incompatible with either attachment possibility. A quadruple is deemed correct if: (1) it exists in the gold standard, and (2) the attachment decision is correct. Conversely, it is deemed incorrect if: (1) it exists in the gold standard, and (2) the attachment decision is incorrect. Quadruples not found in the gold standard are discarded. Precision was measured as the number of correct quadruples divided by the total number of correct and incorrect quadruples (i.e. all quadruples which are not discarded), and recall as the number of correct quadruples divided by the total number of gold-standard quadruples in the test set. This evaluation methodology coincides with that of Atterer and Schütze (2007).

Statistical significance was calculated based on

[3]`www.cis.upenn.edu/~dbikel/software.html`

a modified version of the Bikel comparator (see above), once again with $p < 0.05$.

4.3 Semantic representation

We experimented with a range of semantic representations, all of which are based on WordNet 2.1. As mentioned above, words in WordNet are organised into sets of synonyms, called **synsets**. Each synset in turn belongs to a unique **semantic file** (SF). There are a total of 45 SFs (1 for adverbs, 3 for adjectives, 15 for verbs, and 26 for nouns), based on syntactic and semantic categories. A selection of SFs is presented in Table 1 for illustration purposes.

We experiment with both full synsets and SFs as instances of fine-grained and coarse-grained semantic representation, respectively. As an example of the difference in these two representations, *knife* in its tool sense is in the EDGE TOOL USED AS A CUTTING INSTRUMENT singleton synset, and also in the ARTIFACT SF along with thousands of other words including *cutter*. Note that these are the two extremes of semantic granularity in WordNet, and we plan to experiment with intermediate representation levels in future research (c.f. Li and Abe (1998), McCarthy and Carroll (2003), Xiong et al. (2005), Fujita et al. (2007)).

As a hybrid representation, we tested the effect of merging words with their corresponding SF (e.g. *knife*+ARTIFACT). This is a form of semantic specialisation rather than generalisation, and allows the parser to discriminate between the different senses of each word, but not generalise across words.

For each of these three semantic representations, we experimented with substituting each of: (1) all open-class POSs (nouns, verbs, adjectives and adverbs), (2) nouns only, and (3) verbs only. There are thus a total of 9 combinations of representation type and target POS.

4.4 Disambiguation methods

For a given semantic representation, we need some form of WSD to determine the semantics of each token occurrence of a target word. We experimented with three options:

1. **Gold-standard:** Gold-standard annotations from SemCor. This gives us the upper bound performance of the semantic representation.

SF ID	DEFINITION
adj.all	all adjective clusters
adj.pert	relational adjectives (pertainyms)
adj.ppl	participial adjectives
adv.all	all adverbs
noun.act	nouns denoting acts or actions
noun.animal	nouns denoting animals
noun.artifact	nouns denoting man-made objects
...	
verb.consumption	verbs of eating and drinking
verb.emotion	verbs of feeling
verb.perception	verbs of seeing, hearing, feeling
...	

Table 1: A selection of WordNet SFs

2. **First Sense (1ST):** All token instances of a given word are tagged with their most frequent sense in WordNet.[4] Note that the first sense predictions are based largely on the same dataset as we use in our evaluation, such that the predictions are tuned to our dataset and not fully unsupervised.
3. **Automatic Sense Ranking (ASR):** First sense tagging as for First Sense above, except that an unsupervised system is used to automatically predict the most frequent sense for each word based on an independent corpus. The method we use to predict the first sense is that of McCarthy et al. (2004), which was obtained using a thesaurus automatically created from the British National Corpus (BNC) applying the method of Lin (1998), coupled with WordNet-based similarity measures. This method is fully unsupervised and completely unreliant on any annotations from our dataset.

In the case of SFs, we perform full synset WSD based on one of the above options, and then map the prediction onto the corresponding (unique) SF.

5 Results

We present the results for each disambiguation approach in turn, analysing the results for parsing and PP attachment separately.

[4]There are some differences with the most frequent sense in SemCor, due to extra corpora used in WordNet development, and also changes in WordNet from the original version used for the SemCor tagging.

System	Charniak			Bikel		
	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$
Baseline	.857	.808	.832	.837	.845	.841
SF	.855	.809	.831	**.847***	.854*	.850*
SF_n	.860	.808	.833	**.847***	.853*	.850*
SF_v	.861	.811	.835	**.847***	**.856***	**.851***
word + SF	**.865***	**.814***	**.839***	.837	.846	.842
word + SF_n	.862	.809	.835	.841*	.850*	.846*
word + SF_v	.862	.810	.835	.840	.851	.845
Syn	.863*	.812	.837	.845*	.853*	.849*
Syn_n	.860	.807	.832	.841	.849	.845
Syn_v	.863*	.813*	.837*	.843*	.851*	.847*

Table 2: Parsing results with gold-standard senses (* indicates that the recall or precision is significantly better than baseline; the best performing method in each column is shown in **bold**)

System	Charniak			Bikel		
	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$
Baseline	.667	.798	.727	.659	.820	.730
SF	.710	.808	.756	.714*	.809	.758
SF_n	.671	.792	.726	.706	.818	.758
SF_v	**.729***	**.823**	**.773***	**.733***	.827	**.778***
word + SF	.710*	.801	.753	.706*	**.837**	.766*
word + SF_n	.698*	.813	.751	.706*	.829	.763*
word + SF_v	.714*	.805	.757*	.706*	**.837***	.766*
Syn	.722*	.814	.765*	.702*	.825	.758
Syn_n	.678	.805	.736	.690	.822	.751
Syn_v	.702*	.817*	.755*	.690*	.834	.755*

Table 3: PP attachment results with gold-standard senses (* indicates that the recall or precision is significantly better than baseline; the best performing method in each column is shown in **bold**)

5.1 Gold standard

We disambiguated each token instance in our corpus according to the gold-standard sense data, and trained both the Charniak and Bikel parsers over each semantic representation. We evaluated the parsers in full parsing and PP attachment contexts.

The results for parsing are given in Table 2. The rows represent the three semantic representations (including whether we substitute only nouns, only verbs or all POS). We can see that in almost all cases the semantically-enriched representations improve over the baseline parsers. These results are statistically significant in some cases (as indicated by *). The SF_v representation produces the best results for Bikel (F-score 0.010 above baseline), while for Charniak the best performance is obtained with word+SF (F-score 0.007 above baseline). Comparing the two baseline parsers, Bikel achieves better precision and Charniak better recall. Overall, Bikel obtains a superior F-score in all configurations.

The results for the PP attachment experiments using gold-standard senses are given in Table 3, both for the Charniak and Bikel parsers. Again, the F-score for the semantic representations is better than the baseline in all cases. We see that the improvement is significant for recall in most cases (particularly when using verbs), but not for precision (only Charniak over Syn_v and word+SF_v for Bikel). For both parsers the best results are achieved with SF_v, which was also the best configuration for parsing with Bikel. The performance gain obtained here is larger than in parsing, which is in accordance with the findings of Stetina and Nagao that lexical semantics has a considerable effect on PP attachment performance. As in full-parsing, Bikel outperforms Charniak, but in this case the difference in the baselines is not statistically significant.

5.2 First sense (1ST)

For this experiment, we use the first sense data from WordNet for disambiguation. The results for full parsing are given in Table 4. Again, the performance is significantly better than baseline in most cases, and surprisingly the results are even better than gold-standard in some cases. We hypothesise that this is due to the avoidance of excessive fragmentation, as occurs with fine-grained senses. The results are significantly better for nouns, with SF_n performing best. Verbs seem to suffer from lack of disambiguation precision, especially for Bikel. Here again, Charniak trails behind Bikel.

The results for the PP attachment task are shown in Table 5. The behaviour is slightly different here, with Charniak obtaining better results than Bikel in most cases. As was the case for parsing, the performance with 1ST reaches and in many instances surpasses gold-standard levels, achieving statistical significance over the baseline in places. Comparing the semantic representations, the best results are achieved with SF_v, as we saw in the gold-standard PP-attachment case.

5.3 Automatic sense ranking (ASR)

The final option for WSD is automatic sense ranking, which indicates how well our method performs in a completely unsupervised setting.

The parsing results are given in Table 6. We can see that the scores are very similar to those from

System	Charniak			Bikel		
	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$
Baseline	.857	.807	.832	.837	.845	.841
SF	.851	.804	.827	.843	.850	.846
SF_n	**.863***	**.813**	**.837***	**.850***	**.854***	**.852***
SF_v	.857	.808	.832	.843	.853*	.848
word + SF	.859	.810	.834	.833	.841	.837
word + SF_n	.862*	.811	.836	.844*	.851*	.848*
word + SF_v	.857	.808	.832	.831	.839	.835
Syn	.857	.810	.833	.837	.844	.840
Syn_n	**.863***	.812	**.837***	.844*	.851*	.848*
Syn_v	.860	.810	.834	.836	.844	.840

Table 4: Parsing results with 1st (* indicates that the recall or precision is significantly better than baseline; the best performing method in each column is shown in **bold)**

System	Charniak			Bikel		
	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$
Baseline	.857	.807	.832	.837	.845	.841
SF	.863	**.815***	**.838**	.845*	.852	.849
SF_n	.862	.810	.835	.845*	.850	.847*
SF_v	.859	.810	.833	**.846***	**.856***	**.851***
word + SF	.859	.810	.834	.836	.844	.840
word + SF_n	**.865***	.813*	**.838***	.844*	.852*	.848*
word + SF_v	.856	.806	.830	.832	.839	.836
Syn	.856	.807	.831	.840	.847	.843
Syn_n	.864*	.813*	**.838***	.844*	.851*	.847*
Syn_v	.857	.806	.831	.837	.845	.841

Table 6: Parsing results with ASR (* indicates that the recall or precision is significantly better than baseline; the best performing method in each column is shown in **bold)**

System	Charniak			Bikel		
	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$
Baseline	.667	.798	.727	.659	.820	.730
SF	.710	.808	.756	.702	.806	.751
SF_n	.671	.781	.722	.702	**.829**	.760
SF_v	**.737***	**.836***	**.783***	**.718***	.821	**.766***
word + SF	.706	.811	.755	.694	.823	.753
word + SF_n	.690	.815	.747	.667	.810	.731
word + SF_v	.714*	.805	.757*	.710*	.819	.761*
Syn	.725*	.833*	.776*	.698	.828	.757
Syn_n	.698	.828*	.757*	.667	.817	.734
Syn_v	.722*	.811	.763*	.706*	.818	.758*

Table 5: PP attachment results with 1st (* indicates that the recall or precision is significantly better than baseline; the best performing method in each column is shown in **bold)**

System	Charniak			Bikel		
	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{F}_1$
Baseline	.667	.798	.727	.659	.820	.730
SF	**.733***	.824	.776*	.698	.805	.748
SF_n	.682	.791	.733	.671	.807	.732
SF_v	**.733***	.813	.771*	.710*	.812	.757*
word + SF	.714*	.798	.754	.675	.800	.732
word + SF_n	.690	.807	.744	.659	.804	.724
word + SF_v	.706*	.800	.750	.702*	.814	.754*
Syn	**.733***	**.827**	**.778***	.694	.805	.745
Syn_n	.686	.810	.743	.667	.806	.730
Syn_v	.714*	.816	.762*	**.714***	**.816**	**.762***

Table 7: PP attachment results with ASR (* indicates that the recall or precision is significantly better than baseline; the best performance in each column is shown in **bold)**

1st, with improvements in some cases, particularly for Charniak. Again, the results are better for nouns, except for the case of SF_v with Bikel. Bikel outperforms Charniak in terms of F-score in all cases.

The PP attachment results are given in Table 7. The results are similar to 1st, with significant improvements for verbs. In this case, synsets slightly outperform SF. Charniak performs better than Bikel, and the results for Syn_v are higher than the best obtained using gold-standard senses.

6 Discussion

The results of the previous section show that the improvements in parsing results are small but significant, for all three word sense disambiguation strategies (gold-standard, 1st and ASR). Table 8 summarises the results, showing that the error reduction rate (ERR) over the parsing F-score is up to 6.9%, which is remarkable given the relatively superficial strategy for incorporating sense information into the parser. Note also that our baseline results for the dataset are almost the same as previous work parsing the Brown corpus with similar models (Gildea, 2001), which suggests that our dataset is representative of this corpus.

The improvement in PP attachment was larger (20.5% ERR), and also statistically significant. The results for PP attachment are especially important, as we demonstrate that the sense information has high utility when embedded within a parser, where the parser needs to first identify the ambiguity and heads correctly. Note that Atterer and Schütze (2007) have shown that the Bikel parser performs as well as the state-of-the-art in PP attachment, which suggests our method improves over the current state-of-the-art. The fact that the improvement is larger for PP attachment than for full parsing is suggestive of PP attachment being a parsing subtask where lexical semantic information is particularly important, supporting the findings of Stetina and Nagao (1997) over a standalone PP attachment task. We also observed that while better PP-attachment usually improves parsing, there is some small variation. This

WSD	TASK	PAR	BASE	SEM	ERR	BEST
Gold-standard	Pars.	C	.832	.839*	4.2%	word+SF
		B	.841	.851*	6.3%	SF_v
	PP	C	.727	.773*	16.9%	SF_v
		B	.730	.778*	17.8%	SF_v
1ST	Pars.	C	.832	.837*	3.0%	SF_n, Syn_n
		B	.841	.852*	6.9%	SF_n
	PP	C	.727	.783*	20.5%	SF_v
		B	.730	.766*	13.3%	SF_v
ASR	Pars.	C	.832	.838*	3.6%	SF, word+SF_n, Syn_n
		B	.841	.851*	6.3%	SF_v
	PP	C	.727	.778*	18.7%	Syn
		B	.730	.762*	11.9%	Syn_v

Table 8: Summary of F-score results with error reduction rates and the best semantic representation(s) for each setting (C = Charniak, B = Bikel)

means that the best configuration for PP-attachment does not always produce the best results for parsing

One surprising finding was the strong performance of the automatic WSD systems, actually outperforming the gold-standard annotation overall. Our interpretation of this result is that the approach of annotating all occurrences of the same word with the same sense allows the model to avoid the data sparseness associated with the gold-standard distinctions, as well as supporting the merging of different words into single semantic classes. While the results for gold-standard senses were intended as an upper bound for WordNet-based sense information, in practice there was very little difference between gold-standard senses and automatic WSD in all cases barring the Bikel parser and PP attachment.

Comparing the two parsers, Charniak performs better than Bikel on PP attachment when automatic WSD is used, while Bikel performs better on parsing overall. Regarding the choice of WSD system, the results for both approaches are very similar, showing that ASR performs well, even if it does not require sense frequency information.

The analysis of performance according to the semantic representation is not so clear cut. Generalising only verbs to semantic files (SF_v) was the best option in most of the experiments, particularly for PP-attachment. This could indicate that semantic generalisation is particularly important for verbs, more so than nouns.

Our hope is that this paper serves as the bridgehead for a new line of research into the impact of lexical semantics on parsing. Notably, more could be done to fine-tune the semantic representation between the two extremes of full synsets and SFs. One could also imagine that the appropriate level of generalisation differs across POS and even the relative syntactic role, e.g. finer-grained semantics are needed for the objects than subjects of verbs.

On the other hand, the parsing strategy is very simple, as we just substitute words by their semantic class and then train statistical parsers on the transformed input. The semantic class should be an information source that the parsers take into account in addition to analysing the actual words used. Tighter integration of semantics into the parsing models, possibly in the form of discriminative reranking models (Collins and Koo, 2005; Charniak and Johnson, 2005; McClosky et al., 2006), is a promising way forward in this regard.

7 Conclusions

In this work we have trained two state-of-the-art statistical parsers on semantically-enriched input, where content words have been substituted with their semantic classes. This simple method allows us to incorporate lexical semantic information into the parser, without having to reimplement a full statistical parser. We tested the two parsers in both a full parsing and a PP attachment context.

This paper shows that semantic classes achieve significant improvement both on full parsing and PP attachment tasks relative to the baseline parsers. PP attachment achieves a 20.5% ERR, and parsing 6.9% without requiring hand-tagged data.

The results are highly significant in demonstrating that a simplistic approach to incorporating lexical semantics into a parser significantly improves parser performance. As far as we know, these are the first results over both WordNet and the Penn Treebank to show that semantic processing helps parsing.

Acknowledgements

We wish to thank Diana McCarthy for providing us with the sense rank for the target words. This work was partially funded by the Education Ministry (project KNOW TIN2006-15049), the Basque Government (IT-397-07), and the Australian Research Council (grant no. DP0663879). Eneko Agirre participated in this research while visiting the University of Melbourne, based on joint funding from the Basque Government and HCSNet.

References

Eneko Agirre and Philip Edmonds, editors. 2006. *Word Sense Disambiguation: Algorithms and Applications*. Springer, Dordrecht, Netherlands.

Michaela Atterer and Hinrich Schütze. 2007. Prepositional phrase attachment without oracles. *Computational Linguistics*, 33(4):469–476.

Daniel M. Bikel. 2000. A statistical model for parsing and word-sense disambiguation. In *Proc. of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, pages 155–63, Hong Kong, China.

Daniel M. Bikel. 2004. Intricacies of Collins' parsing model. *Computational Linguistics*, 30(4):479–511.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proc. of the 43rd Annual Meeting of the ACL*, pages 173–80, Ann Arbor, USA.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proc. of the 15th Annual Conference on Artificial Intelligence (AAAI-97)*, pages 598–603, Stanford, USA.

Eugene Charniak. 2000. A maximum entropy-based parser. In *Proc. of the 1st Annual Meeting of the North American Chapter of Association for Computational Linguistics (NAACL2000)*, Seattle, USA.

David Chiang and David M. Bikel. 2002. Recovering latent information in treebanks. In *Proc. of the 19th International Conference on Computational Linguistics (COLING 2002)*, pages 183–9, Taipei, Taiwan.

Michael Collins and Terry Koo. 2005. Discriminative reranking for natural language parsing. *Computational Linguistics*, 31(1):25–69.

Michael J. Collins. 1996. A new statistical parser based on lexical dependencies. In *Proc. of the 34th Annual Meeting of the ACL*, pages 184–91, Santa Cruz, USA.

Michael Collins. 2003. Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637.

John Dowding, Robert Moore, François Andry, and Douglas Moran. 1994. Interleaving syntax and semantics in an efficient bottom-up parser. In *Proc. of the 32nd Annual Meeting of the ACL*, pages 110–6, Las Cruces, USA.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, USA.

Sanae Fujita, Francis Bond, Stephan Oepen, and Takaaki Tanaka. 2007. Exploiting semantic information for HPSG parse selection. In *Proc. of the ACL 2007 Workshop on Deep Linguistic Processing*, pages 25–32, Prague, Czech Republic.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proc. of the 6th Conference on Empirical Methods in Natural Language Processing (EMNLP 2001)*, pages 167–202, Pittsburgh, USA.

Erik Hektoen. 1997. Probabilistic parse selection based on semantic cooccurrences. In *Proc. of the 5th International Workshop on Parsing Technologies (IWPT-1997)*, pages 113–122, Boston, USA.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In *Proc. of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA.

Shari Landes, Claudia Leacock, and Randee I. Tengi. 1998. Building semantic concordances. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, USA.

Hang Li and Naoki Abe. 1998. Generalising case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–44.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proc. of the 36th Annual Meeting of the ACL and 17th International Conference on Computational Linguistics: COLING/ACL-98*, pages 768–774, Montreal, Canada.

David M. Magerman. 1995. Statistical decision-tree models for parsing. In *Proc. of the 33rd Annual Meeting of the ACL*, pages 276–83, Cambridge, USA.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn treebank. *Computational Linguistics*, 19(2):313–30.

Diana McCarthy and John Carroll. 2003. Disambiguating nouns, verbs and adjectives using automatically acquired selectional preferences. *Computational Linguistics*, 29(4):639–654.

Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. 2004. Finding predominant senses in untagged text. In *Proc. of the 42nd Annual Meeting of the ACL*, pages 280–7, Barcelona, Spain.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Effective self-training for parsing. In *Proc. of the Human Language Technology Conference of the NAACL (NAACL2006)*, pages 152–159, New York City, USA.

Brian Mitchell. 2004. *Prepositional Phrase Attachment using Machine Learning Algorithms*. Ph.D. thesis, University of Sheffield.

Martha Palmer, Hoa Dang, and Christiane Fellbaum. 2006. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering*, 13(2):137–63.

Adwait Ratnaparkhi, Jeff Reynar, and Salim Roukos. 1994. A maximum entropy model for prepositional phrase attachment. In *HLT '94: Proceedings of the Workshop on Human Language Technology*, pages 250–255, Plainsboro, USA.

Philip Resnik. 2006. WSD in NLP applications. In Eneko Agirre and Philip Edmonds, editors, *Word Sense Disambiguation: Algorithms and Applications*, chapter 11, pages 303–40. Springer, Dordrecht, Netherlands.

Jiri Stetina and Makoto Nagao. 1997. Corpus based PP attachment ambiguity resolution with a semantic dictionary. In *Proc. of the 5th Annual Workshop on Very Large Corpora*, pages 66–80, Hong Kong, China.

Deyi Xiong, Shuanglong Li, Qun Liu, Shouxun Lin, and Yueliang Qian. 2005. Parsing the Penn Chinese Treebank with semantic knowledge. In *Proc. of the 2nd International Joint Conference on Natural Language Processing (IJCNLP-05)*, pages 70–81, Jeju Island, Korea.