A Statistical Model for Parsing and Word-Sense Disambiguation

Daniel M. Bikel
Dept. of Computer & Information Science
University of Pennsylvania
200 South 33rd Street, Philadelphia, PA 19104-6389, U.S.A.
dbikel@cis.upenn.edu

Abstract

This paper describes a first attempt at a statistical model for simultaneous syntactic parsing and generalized word-sense disambiguation. On a new data set we have constructed for the task, while we were disappointed not to find parsing improvement over a traditional parsing model, our model achieves a recall of 84.0% and a precision of 67.3% of exact synset matches on our test corpus, where the gold standard has a reported inter-annotator agreement of 78.6%.

1 Introduction

In this paper we describe a generative, statistical model for simultaneously producing syntactic parses and word senses in sentences. We begin by motivating this new approach to these two, previously-separate problems, then, after reviewing previous work in these areas, we describe our model in detail. Finally, we will present the promising results of this, our first attempt, and the direction of future work.

2 Motivation for the Approach

2.1 Motivation from examples

Consider the following examples:

1. IBM bought Lotus for $200 million.
2. Sony widened its product line with personal computers.
3. The bank issued a check for $100,000.
4. Apple is expecting [NP strong results].
5. IBM expected [SBAR each employee to wear a shirt and tie].

With Example 1, the reading [IBM bought [Lotus for $200 million]] is nearly impossible, for the simple reason that a monetary amount is a likely instrument for buying and not for describing a company. Similarly, there is a reasonably strong preference in Example 2 for [PP with personal computers] to attach to *widened*, because personal computers are products with which a product line could be widened. As pointed out by (Stetina and Nagao, 1997), word sense information can be a proxy for the semantic- and world-knowledge we as humans bring to bear on attachment decisions such as these. This proxy effect is due to the "lightweight semantics" that word senses—in particular WordNet word senses—convey.

Conversely, both the syntactic and semantic context in Example 3 let us know that *bank* is not a river bank and that *check* is not a restaurant bill. In Examples 4 and 5, knowing that the complement of *expect* is an NP or an SBAR provides information as to whether the sense is "await" or "require". Thus, Examples 3–5 illustrate how the syntactic context of a word can help determine its meaning.

2.2 Motivation from previous work

2.2.1 Parsing

In recent years, the success of statistical parsing techniques can be attributed to several factors, such as the increasing size of computing machinery to accommodate larger models, the availability of resources such as the Penn Treebank (Marcus et al., 1993) and the success of machine learning techniques for lower-level NLP problems, such as part-of-speech tagging (Church, 1988; Brill, 1995), and PP-attachment (Brill and Resnik, 1994; Collins and Brooks, 1995). However, perhaps even more significant has been the lexicalization of the grammar formalisms being probabilistically modeled: crucially, all the recent, successful statistical parsers have in some way made use of bilexical dependencies. This includes both the parsers that attach probabilities to parser moves (Magerman, 1995; Ratnaparkhi, 1997), but also those of the lexicalized PCFG variety (Collins, 1997; Charniak, 1997).

Even more crucially, the bilexical dependencies involve head-modifier relations (hereafter referred to simply as "head relations"). The intuition behind the lexicalization of a grammar formalism is to capture lexical items' idiosyncratic parsing preferences. The intuition behind using heads as the members of the bilexical relations is twofold. First, many linguistic theories tell us that the head of a phrase projects the skeleton of that phrase, to be filled in by specifiers, complements and adjuncts; such a notion is captured quite directly by a formalism such as LTAG (Joshi and Schabes, 1997). Second, the head of a phrase usually conveys some large component of the semantics of that phrase.[1] In this way, using head-relation statistics encodes a bit of the predicate-argument structure in the syntactic model. While there are cases such as *John was believed to have been shot by Bill* where structural preference virtually eliminates one of the two semantically plausible analyses, it is quite clear that semantics—and, in particular, lexical head semantics—play a very important role in reducing parsing ambiguity. (See (Collins, 1999), pp. 207ff., for an excellent discussion of structural vs. semantic parsing preferences, including the above *John was believed...* example.)

Another motivation for incorporating word senses into a statistical parsing model has been to ameliorate the sparse data problem. Inspired by the PP-attachment work of (Stetina and Nagao, 1997), we use WordNet v1.6 (Miller et al., 1990) as our semantic dictionary, where the hypernym structure provides the basis for semantically-motivated soft clusters.[2] We discuss this benefit of word senses and the details of our implementation further in Section 4.

2.2.2 Word-sense disambiguation

While there has been much work in this area, let us examine the features used in recent statistical approaches. (Yarowsky, 1992) uses wide "bag-of-words" contexts with a naive Bayes classifier. (Yarowsky, 1995) also uses wide context, but incorporates the one-sense-per-discourse and one-sense-per-collocation constraints, using an unsupervised learning technique. The supervised technique in (Yarowsky, 1994) has a more specific notion of context, employing not just words that can appear within a window of $\pm k$, but crucially words that abut and fall in the ± 2 window of the target word. More recently, (Lin, 1997) has shown how syntactic context, and dependency structures in particular, can be successfully employed for word sense disambiguation. (Stetina and Nagao, 1997) have shown that by employing a fairly simple and somewhat ad-hoc unsupervised method of WSD using a WordNet-based similarity heuristic, they could enhance PP-attachment performance to a significantly higher level than systems that made no use of lexical semantics (88.1% accuracy). Most recently, in (Stetina et al., 1998), the authors made use of head-driven bilexical dependencies with syntactic relations to attack the problem of generalized word-sense disambiguation, precisely one of the two problems we are dealing with here.

3 The Model

3.1 Overview

The parsing model we started with was extracted from BBN's SIFT system (Miller et al., 1998), which we briefly present again here, using examples from Figure 1 to illustrate the model's parameters.[3]

The model generates the head of a constituent first, then each of the left- and right-modifiers, generating from the head outward, using a bigram model of node labels. Here are the first few elements generated by the model for the tree of Figure 1:

1. S and its head word and part of speech, *caught–VBD.*
2. The head constituent of S, VP.
3. The head word of the VP, *caught–VBD.*
4. The premodifier constituent ADVP.

[1] Heads originated this way, but it has become necessary to distinguish "semantic" heads, such as nouns and verbs, that correspond roughly to predicates and arguments, from "functional" heads, such as determiners, INFL's and complementizers, that correspond roughly to logical operators or are purely syntactic elements. In this paper, we almost always intend "head" to mean "semantic head".

[2] Soft clusters are sets where the elements have weights indicating the strength of their membership in the set, which in this case allows for a probability distribution to be defined over a word's membership in all the clusters.

[3] We began with the BBN parser because its authors were kind enough to allow us to extend it, and because its design allowed easy integration with our existing WordNet code.

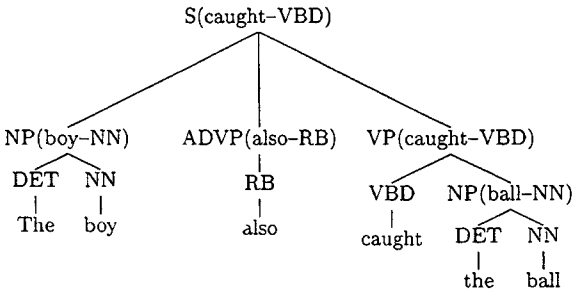

Figure 1: A sample sentence with parse tree.

5. The head word of the premodifier ADVP, *also–RB*.
6. The premodifier constituent NP.
7. The head word of the premodifier NP, *boy–NN*.
8. The +END+ (null) postmodifier constituent of the VP.

This process recurses on each of the modifier constituents (in this case, the subject NP and the VP) until all words have been generated. (Note that many words effectively get generated high up in the tree; in this example sentence, the last words to get generated are the two *the*'s)

More formally, the lexicalized PCFG that sits behind the parsing model has rules of the form

$$P \rightarrow L_n L_{n-1} \cdots L_1 H R_1 \cdots R_{n-1} R_n \quad (1)$$

where P, H, L_i and R_i are all lexicalized nonterminals, *i.e.*, of the form $X\langle w, t, f\rangle$, where X is a traditional CFG nonterminal and $\langle w, t, f\rangle$ is the word–part-of-speech–word-feature triple that is the head of the phrase denoted by X.[4] The lexicalized nonterminal H is so named because it is the *head constituent*, where P inherits its head triple from this head constituent. The constituents labeled L_i and R_i are left- and right-modifier constituents, respectively.

3.2 Probability structure of the original model

We use p to denote the unlexicalized nonterminal corresponding to P, and similarly for l_i, r_i and h. We now present the top-level generation probabilities, along with examples from Figure 1. For brevity, we omit the smoothing details of BBN's model (see (Miller et al., 1998) for a complete description); we note that all smoothing weights are computed via the technique described in (Bikel et al., 1997).

The probability of generating p as the root label is predicted conditioning on only +TOP+, which is the hidden root of all parse trees:

$$P(p \mid +\mathrm{TOP}+), \textit{ e.g., } P(\mathrm{S} \mid +\mathrm{TOP}+). \quad (2)$$

The probability of generating a head node h with a parent p is

$$P(h \mid p), \textit{ e.g., } P(\mathrm{VP} \mid \mathrm{S}). \quad (3)$$

The probability of generating a left-modifier l_i is

$$P_L(l_i \mid l_{i-1}, p, h, w_h), \textit{ e.g., } P_L(\mathrm{NP} \mid \mathrm{ADVP}, \mathrm{S}, \mathrm{VP}, \mathrm{caught}) \quad (4)$$

when generating the NP for NP(boy–NN), and the probability of generating a right modifier r_i is

$$P_R(r_i \mid r_{i-1}, p, h, w_h), \textit{ e.g., } P_R(\mathrm{NP} \mid +\mathrm{BEGIN}+, \mathrm{VP}, \mathrm{VBD}, \mathrm{caught}) \quad (5)$$

when generating the NP for NP(ball–NN).[5]

The probabilities for generating lexical elements (part-of-speech tags, words and word features) are as follows. The part of speech tag of the head of the entire sentence, t_h, is

[4] The inclusion of the word feature in the BBN model was due to the work described in (Weischedel et al., 1993), where word features helped reduce part of speech ambiguity for unknown words.

[5] The hidden nonterminal +BEGIN+ is used to provide a convenient mechanism for determining the initial probability of the underlying Markov process generating the modifying nonterminals; the hidden nonterminal +END+ is used to provide consistency to the underlying Markov process, *i.e.*, so that the probabilities of all possible nonterminal sequences sum to 1.

computed conditioning only on the top-most symbol p:[6]

$$P(t_h \mid p). \tag{6}$$

Part of speech tags of modifier constituents, t_{l_i} and t_{r_i}, are predicted conditioning on the modifier constituent l_i or r_i, the tag of the head constituent, t_h, and the word of the head constituent, w_h:

$$P(t_{l_i} \mid l_i, t_h, w_h) \text{ and } P(t_{r_i} \mid r_i, t_h, w_h). \tag{7}$$

The head word of the entire sentence, w_h, is predicted conditioning only on the top-most symbol p and t_h.

$$P(w_h \mid t_h, p). \tag{8}$$

Head words of modifier constituents, w_{l_i} and w_{r_i}, are predicted conditioning on all the context used for predicting parts of speech in (7), as well as the parts of speech themsleves

$$\begin{aligned} &P(w_{l_i} \mid t_{l_i}, l_i, t_h, w_h) \\ &\quad \text{and} \quad P(w_{r_i} \mid t_{r_i}, r_i, t_h, w_h). \end{aligned} \tag{9}$$

The word feature of the head of the entire sentence, f_h, is predicted conditioning on the top-most symbol p, its head word, w_h, and its head tag, t_h:

$$P(f_h \mid w_h, t_h, p). \tag{10}$$

Finally, the word features for the head words of modifier constituents, f_{l_i} and f_{r_i}, are predicted conditioning on all the context used to predict modifier head words in (9), as well as the modifier head words themselves:

$$\begin{aligned} &P(f_{l_i} \mid \text{known}(w_{l_i}), t_{l_i}, l_i, t_h, w_h) \\ &\quad \text{and} \quad P(f_{r_i} \mid \text{known}(w_{r_i}), t_{r_i}, r_i, t_h, w_h) \end{aligned} \tag{11}$$

where known(x) is a predicate returning *true* if the word x was observed more than 4 times in the training data.

The probability of an entire parse tree is the product of the probabilities of generating all of the elements of that parse tree, where an element is either a constituent label, a part of speech tag, a word or a word feature. We obtain maximum-likelihood estimates of the parameters of this model using frequencies gathered from the training data.

[6]This is the one place where we have altered the original model, as the lexical components of the head of the entire sentence were all being estimated incorrectly, causing an inconsistency in the model. We have corrected the estimation of t_h, w_h and f_h in our implementation.

4 Word-sense Extensions to the Lexical Model

The desired output structure of our combined parser/word-sense disambiguator is a standard, Treebank-style parse tree, where the words not only have parts of speech, but also WordNet synsets. Incorporating synsets into the lexical part of the model is fairly straightforward: a synset is yet another element to be generated. The question is when to generate it. The lexical model has decomposed the generation of the $\langle w, t, f \rangle$ triple into three steps, each conditioning on all the history of the previous step. While it is probabilistically identical to predict synsets at any of the four possible points if we continue to condition on all the history at each step, we would like to pick the point that is most well-founded both in terms of the underlying linguistic structure and in terms of what can be well-estimated. In Section 2.2.1 we mentioned the soft-clustering aspect of synsets; in fact, they have a duality. On the one hand, they serve to add specificity to what might otherwise be an ambiguous lexical item; on the other, they are *sets*, clustering lexical items that have similar meanings. Even further, noun and verb synsets form a *concept taxonomy*, the hypernym relation forming a partial ordering on the lemmas contained in WordNet. The former aspect corresponds roughly to what we as human listeners or readers do: we hear or see a sequence of words in context, and determine incrementally the particular meaning of each of those words. The latter aspect corresponds more closely to a mental model of generation: we have a desire or intention to convey, we choose the appropriate concepts with which to convey it, and we realize that desire or intention with the most felicitous syntactic structure and lexical realizations of those concepts. As this is a generative model, we generate a word's synset after generating the part of speech tag but *before* generating the word itself.[7]

The synset of the head of the entire sentence, s_h is predicted conditioning only on the top-most symbol p and the head tag, t_h:

$$P(s_h \mid t_h, p). \tag{12}$$

We accordingly changed the probability of

[7]We believe that synsets and parts of speech are largely orthogonal with respect to their lexical information, and thus their relative order of prediction was not a concern.

generating the head word of the entire sentence to be

$$P(w_h \mid s_h, t_h, p). \tag{13}$$

The probability estimates for (12) and (13) are not smoothed.

The probability model for generating synsets of modifier constituents m_i, complete with smoothing components, is as follows:

$$\begin{aligned}\hat{P}(s_{m_i} \mid t_{m_i}, m_i, w_h, s_h) = \quad & (14)\\ \lambda_0 \hat{P}(s_{m_i} \mid t_{m_i}, m_i, w_h, s_h) & \\ + \ \lambda_1 \hat{P}(s_{m_i} \mid t_{m_i}, m_i, s_h) & \\ + \ \lambda_2 \hat{P}(s_{m_i} \mid t_{m_i}, m_i, @^1(s_h)) & \\ + \ \cdots & \\ + \ \lambda_{n+1} \hat{P}(s_{m_i} \mid t_{m_i}, m_i, @^n(s_h)) & \\ + \ \lambda_{n+2} \hat{P}(s_{m_i} \mid t_{m_i}, m_i) & \\ + \ \lambda_{n+3} \hat{P}(s_{m_i} \mid t_{m_i}) &\end{aligned}$$

where $@^i(s_h)$ is the i^{th} hypernym of s_h. The WordNet hypernym relations, however, do not form a tree, but a DAG, so whenever there are multiple hypernyms, the uniformly-weighted mean is taken of the probabilities conditioning on each of the hypernyms. That is,

$$\begin{aligned}\hat{P}(s_{m_i} \mid t_{m_i}, m_i, @^j(s_h)) = \quad & (15)\\ \frac{1}{n} \sum_{k=1}^{n} \hat{P}(s_{m_i} \mid t_{m_i}, m_i, @_k^j(s_h)) & \\ \text{when } @^j(s_h) = \{@_1^j(s_h), \ldots, @_n^j(s_h)\}. &\end{aligned}$$

Note that in the first level of back-off, we no longer condition on the head word, but strictly on its synset, and thereafter on hypernyms of that synset; these models, then, get at the heart of our approach, which is to abstract away from *lexical* head relations, and move to the more general *lexico-semantic* relations, here represented by synset relations.

Now that we generate synsets for words using (14), we can also change the word generation model to have synsets in its history:

$$\begin{aligned}\hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}, m_i, w_h, s_h) = \quad & (16)\\ \lambda_0 \hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}, m_i, w_h) & \\ + \ \lambda_1 \hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}, m_i, s_h) & \\ + \ \lambda_2 \hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}, m_i, @^1(s_h)) & \\ + \ \cdots & \\ + \ \lambda_{n+1} \hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}, m_i, @^n(s_h)) & \\ + \ \lambda_{n+2} \hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}, m_i) & \\ + \ \lambda_{n+3} \hat{P}(w_{m_i} \mid s_{m_i}, t_{m_i}) & \\ + \ \lambda_{n+4} \hat{P}(w_{m_i} \mid s_{m_i}) &\end{aligned}$$

where once again, $@^i(s_h)$ is the i^{th} hypernym of s_h. For both the word and synset prediction models, by backing off up the hypernym chain, there is an appropriate conflation of similar head relations. For example, if in training the verb phrase [strike the target] had been seen, if the unseen verb phrase [attack the target] appeared during testing, then the training from the semantically-similar training phrase could be used, since this sense of *attack* is the hypernym of this sense of *strike*.

Finally, we note that both of these synset- and word-prediction probability estimates contain an enormous number of back-off levels for nouns and verbs, corresponding to the head word's depth in the synset hierarchy. A valid concern would be that the model might be backing off using histories that are far too general, so we experimented with limiting the hypernym back-off to only two, three and four levels. This change produced a negligible difference in parsing performance.[8]

5 A New Approach, A New Data Set

Ideally, the well-established gold standard for syntax, the Penn Treebank, would have a parallel word-sense–annotated corpus; unfortunately, no such word-sense corpus exists. However, we do have SemCor (Miller et al., 1994), where every noun, verb, adjective and adverb from a 455k word portion of the Brown Corpus has been assigned a WordNet synset. While all of the Brown Corpus was annotated in the style of Treebank I, a great deal was also more recently annotated in Treebank II format, and this corpus has recently been released by the Linguistic Data Consortium.[9] As it happens, the intersection between the Treebank-II-annotated Brown and SemCor comprises some 220k words, most of which is fiction, with some nonfiction and humor writing as well.

We went through all 220k words of the corpora, synchronizing them. That is, we made sure that the corpora were identical up to the spelling of individual tokens, correcting all

[8] We aim to investigate the precise effects of our back-off strategy in the next version of our combined parsing/WSD model.

[9] We were given permission to use a pre-release version of this Treebank II–style corpus.

tokenization and sentence-breaking discrepancies. This correcton task ranged from the simple, such as connecting two sentences in one corpus that were erroneously broken, to the middling, such as joining two tokens in SemCor that comprised a hyphenate in Brown, to the difficult, such as correcting egregious parse annotation errors, or annotating entire sentences that were omitted from SemCor. In particular, the case of hyphenates was quite frequent, as it was the default in SemCor to split up all such words and assign them their individual word senses (synsets). In general, we attempted to make SemCor look as much as possible like the Treebank II–annotated Brown, and we used the following guidelines for assigning word senses to hyphenates:

1. Assign the word sense of the head of the hyphenate. *E.g.*, both *twelve-foot* and *ten-foot* get the word sense of `foot_1` (the unit of measure equal to 12 inches).
2. If there is no clear head, then attempt to annotate with the word sense of the hypernym of the senses of the hyphenate components. *E.g.*, *U.S.-Soviet* gets the word sense of `country_2` (a state or nation).
3. If options 1 and 2 are not possible, the hyphenate is split in the Treebank II file.
4. If the hyphenate has the prefix *non-* or *anti-*, annotate with the word sense of that which follows, with the understanding that a post-processing step could recover the antonymous word sense, if necessary.

After three passes through the corpora, they were perfectly synchronized. We are seeking permission to make this data set available to any who already have access to both SemCor and the Treebank II version of Brown.

After this synchronization process, we merged the word-sense annotations of our corrected SemCor with the tokens of our corrected version of the Treebank II Brown data. Here we were forced to make two decisions. First, SemCor allows multiple synsets to be assigned to a particular word; in these cases, we simply discard all but the first assigned synset. Second, WordNet has collocations, whereas Treebank does not. To deal with this disparity, we re-analyze annotated collocations as a sequence of separate words that have all been assigned the same synset as was assigned the collocation as a whole. This is not as unreasonable as it may sound; for example, `vice_president` is a lemma in WordNet and appears in SemCor, so the merged corpus has instances where the word *president* has the synset `vice_president_1`, but only when preceded by the word *vice*. The cost of this decision is an increase in average polysemy.

6 Training and Decoding

Using this merged corpus, actual training of our model proceeds in an identical fashion to training the non-WordNet–extended model, except that for each lexical relation, the hypernym chain of the parent head is followed to derive counts for the various back-off levels described in Section 4. We also developed a "plug-'n'-play" lexical model system to facilitate experimentation with various word- and synset-prediction models and back-off strategies.

Even though the model is a top-down, generative one, parsing proceeds bottom-up. The model is searched via a modified version of CKY, where candidate parse trees that cover the same span of words are ranked against each other. In the unextended parsing model, the cells corresponding to spans of length one are seeded with $\langle w, t, f \rangle$ triples, with every possible tag t for a given word w (the word-feature f is computed deterministically for w); this step introduces the first degree of ambiguity in the decoding process. Our WordNet-extended model adds to this initial ambiguity, for each cell is seeded with $\langle w, t, f, s \rangle$ quadruples, with every possible synset s for a given word-tag pair.

During decoding, two forms of pruning are employed: a beam is applied to each cell in the chart, pruning away all parses whose ranking score is not within a factor of e^{-k} of the top-ranked parse, and only the top-ranked n subtrees are maintained, and the rest are pruned away. The "out-of-the-box" BBN program uses values of -5 and 25 for k and n, respectively. We changed these to default to -9 and 50, because generating additional unseen items (in our case, synsets) will necessarily lower intermediate ranking scores.

7 Experiments and Results

7.1 Parsing

Initially, we created a small test set, blindly choosing the last 117 sentences, or 1%, of

our 220k word corpus, sentences which were, as it happens, from section "r" of the Brown Corpus. After some disappointing parsing results using both the regular parser and our WordNet-extended version, we peeked in (Francis and Kučera, 1979) and discovered this was the humor writing section; our initial test corpus was literally a joke. To create a more representative test set, we sampled every 100th sentence to create a new 117-sentence test set that spanned the entire range of styles in the 220k words; we put all other sentences in the training set.[10] For the sake of comparison, we present results for both test sets (from section "r" and the balanced test set) and both the standard model (Norm) and our WN-extended model (WN-ext) in Table 1.[11] We note that after we switched to the balanced test set, we did not use the "out-of-the-box" version of the BBN parser, as its default settings for pruning away low-count items and the threshold at which to count a word as "unknown" were too high to yield decent results. Instead, we used precisely the same settings as for our WordNet-extended version, complete with the larger beam width discussed in the previous section.[12]

The reader will note that our extended model performs at roughly the same level as the unextended version with respect to parsing—a shave better with the "r" test set, and slightly worse on the balanced test set. Recall, however, that this is in spite of adding more intermediate ambiguity during the decoding process, and yet using the same beam width. Furthermore, our extensions have occurred strictly within the framework of the original model, but we believe that for the true advantages of synsets to become apparent, we must use trilexical or even tetralexical dependencies. Whereas such long-range dependencies might cripple a standard generative model, the soft-clustering aspects of synsets should offset the sparse data problem. As an example of the lack of such dependencies, in the current model when predicting the attachment of [bought company [for million]], there is no current dependence between the verb *bought* and the object of the preposition *million*—a dependence shown to be useful in virtually all the PP attachment work, and particularly in (Stetina and Nagao, 1997). Related to this issue, we note that the head rules, which were nearly identical to those used in (Collins, 1997), have not been tuned at all to this task. For example, in the sentence in Figure 2, the subject *Jane* is predicted conditioning on the head of the VP, which is the modal *will*, as opposed to the more semantically-content-rich *kill*. So, while the head relations provide a very useful structure for many syntactic decisions the parser needs to make, it is quite possible that the synset relations of this model would require additional or different de-

Model, test set	≤40 words				
	LR	LP	$\overline{\text{CB}}$	0CB	≤2CB
Norm, "r"*	69.7	72.6	2.93	31.9	55.0
WN-ext, "r"	**69.7**	**72.7**	**2.86**	**30.8**	**56.0**
Norm, bal	83.1	85.0	0.82	75.9	85.7
WN-ext, bal	**82.9**	**84.0**	**1.02**	**70.5**	**81.3**
	All sentences				
	LR	LP	$\overline{\text{CB}}$	0CB	≤2CB
Norm, "r"*	68.6	71.2	3.83	25.9	44.8
WN-ext, "r"	**69.7**	**71.5**	**3.77**	**25.0**	**45.7**
Norm, bal	82.0	84.4	1.00	73.5	83.8
WN-ext, bal	**80.5**	**82.2**	**1.43**	**68.4**	**78.6**

Table 1: Results for both parsing models on both test sets. All results are percentages, except for those in the CB column. *See footnote 11.

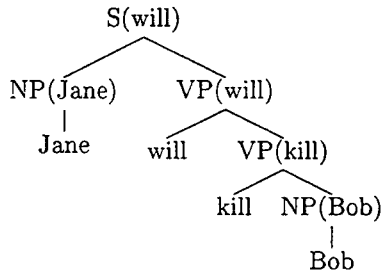

Figure 2: Head rules are tuned for syntax, not semantics.

[10]We realize these are very small test sets, but we presume they are large enough to at least give a good indicator of performance on the tasks evaluated. They were kept small to allow for a rapid train-test-analyze cycle, i.e., they were actually used as development test sets. With the completion of these initial experiments, we are going to designate a proper three-way divsion of training, devtest and test set of this new merged corpus.

[11]The scores in the rows labeled *Norm, "r"*, indicating the performance of the standard BBN model on the "r" test set, are actually scores based on 116 of the 117 sentences, as one sentence did not get parsed due to a timeout in the program.

[12]This is partly an unfair comparison, then, since ours is a larger model, but we wanted to give the standard model every conceivable advantage.

	Recall	Precision
Noun	86.5%	70.9%
Verb	84.0%	59.5%
Adj	80.2%	70.4%
Adv	78.5%	75.8%
Total	**84.0%**	**67.3%**

Table 2: Word sense disambiguation results for balanced test set.

pendencies that would help in the prediction of correct synsets, and in turn help further reduce certain syntactic ambiguities, such as PP attachment. This is because the "lightweight semantics" offered by synset relations can provide selectional and world-knowledge restrictions that simple lexicalized nonterminal relations cannot.

7.2 Word-sense disambiguation

The WSD results on the balanced test set are shown in Table 2. A few important points must be made when evaluating these results. First, almost all other WSD approaches are aimed at distinguishing homonyms, as opposed to the type of fine-grained distinctions that can be made by WordNet. Second, almost all other WSD approaches attempt to disambiguate a small set of such homonymous terms, whereas here we are attacking the *generalized* word-sense disambiguation problem. Third, we call attention to the fact that SemCor has a reported inter-annotator agreement of 78.6% overall, and as low as 70% for words with polysemy of 8 or above (Fellbaum et al., 1998), so it is with this upper bound in mind that one must consider the precision of any generalized WSD system. Finally, we note that the scores in Table 2 are for *exact synset matches*; that is, if our program delivers a synset that is, say, the hypernym or sibling of the correct answer, no credit is given.

While it is tempting to compare these results to those of (Stetina et al., 1998), who reported 79.4% overall accuracy on a different, larger test set using their non-discourse model, we note that that was more of an upper-bound study, examining how well a WSD algorithm could perform if it had access to gold-standard–perfect parse trees.[13] By way of further comparison, that algorithm has a feature space similar to the synset-prediction components of our model, but the steps used to rank possible answers are based largely on heuristics; in contrast, our model is based entirely on maximum-likelihood probability estimates.

A final note on the scores of Table 2: given the fact that there is not a deterministic mapping between the 50-odd Treebank and 4 WordNet parts of speech, when our program delivers a synset for a WordNet part of speech that is different from our gold file, we have called this a recall error, as this is consistent with all other WSD work, where part of speech ambiguity is not a component of an algorithm's precision.

[13] It is not clear how or why the results of (Stetina et al., 1998) exceeded the reported inter-annotator agreement of the entire corpus.

8 Future Work

This paper represents a first attempt at a combined parsing/word sense disambiguation model. Although it has been very useful to work with the BBN model, we are currently implementing and hope to augment a more state-of-the-art model, *viz.*, Models 2 and 3 of (Collins, 1997). We would also like to explore the use of a more radical model, where nonterminals *only* have synsets as their heads, and words are generated strictly at the leaves. We would also like to incorporate long-distance context in the model as an aid to WSD, a demonstrably effective feature in virtually all the recent, statistical WSD work. Also, as mentioned earlier, we believe there are several features that would allow significant parsing improvement. Finally, we would like to investigate the incorporation of unsupervised methods for WSD, such as the heuristically-based methods of (Stetina and Nagao, 1997) and (Stetina et al., 1998), and the theoretically purer bootstrapping method of (Yarowsky, 1995). Bolstered by the success of (Stetina and Nagao, 1997), (Lin, 1997) and especially (Stetina et al., 1998), we believe there is great promise the incorporation of word-sense into a probabilistic parsing model.

9 Acknowledgements

I would like to greatly acknowledge the researchers at BBN who allowed me to use and abuse their parser and who fostered the beginning of this research effort: Scott Miller, Lance Ramshaw, Heidi Fox, Sean Boisen and Ralph Weischedel. Thanks to Michelle Engel, who helped enormously with the task of preparing the new data set. Finally, I would like to thank my advisor Mitch Marcus for his invaluable technical advice and support.

References

Daniel M. Bikel, Richard Schwartz, Ralph Weischedel, and Scott Miller. 1997. Nymble: A high-performance learning name-finder. In *Fifth Conference on Applied Natural Language Processing*, pages 194–201, , Washington, D.C.

E. Brill and P. Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *Fifteenth International Conference on Computational Linguistics (COLING-1994)*.

Eric Brill. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, Menlo Park. AAAI Press/MIT Press.

Kenneth Church. 1988. A stochastic parts program and noun phrase parser for unrestricted text. In *Second Conference on Applied Natural Language Processing*, pages 136–143, Austin, Texas.

M. Collins and J. Brooks. 1995. Prepositional phrase attachment through a backed-off model. In *Third Workshop on Very Large Corpora*, pages 27–38.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of ACL-EACL '97*, pages 16–23.

Michael John Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing.* Ph.D. thesis, University of Pennsylvania.

Christiane Fellbaum, Jaochim Grabowski, and Shari Landes. 1998. Performance and confidence in a semantic annotation task. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, chapter 9. MIT Press, Cambridge, Massachusetts.

W. N. Francis and H. Kučera. 1979. *Manual of Information to accompany A Standard Corpus of Present-Day Edited American English, for use with Digital Computers.* Department of Linguistics, Brown University, Providence, Rhode Island.

Aravind K. Joshi and Yves Schabes. 1997. Tree-adjoining grammars. In A. Salomma and G. Rosenberg, editors, *Handbook of Formal Languages and Automata*, volume 3, pages 69–124. Springer-Verlag, Heidelberg.

Dekang Lin. 1997. Using syntactic dependency as local context to resolve word sense ambiguity. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, Madrid, Spain.

D. Magerman. 1995. Statistical decision tree models for parsing. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 276–283, Cambridge, Massachusetts. Morgan Kaufmann Publishers.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19:313–330.

George A. Miller, Richard T. Beckwith, Christiane D. Fellbaum, Derek Gross, and Katherine J. Miller. 1990. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–244.

George A. Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G. Thomas. 1994. Using a semantic concordance for sense identification. In *Proceedings of the ARPA Human Language Technology Workshop.*

Scott Miller, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. 1998. SIFT – Statistically-derived Information From Text. In *Seventh Message Understanding Conference (MUC-7)*, Washington, D.C.

Adwait Ratnaparkhi. 1997. A linear observed time statistical parser based on maximum entropy models. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, Brown University, Providence, Rhode Island.

Jiri Stetina and Makoto Nagao. 1997. Corpus based PP attachment ambiguity resolution with a semantic dictionary. In *Fifth Workshop on Very Large Corpora*, pages 66–80, Beijing.

Jiri Stetina, Sadao Kurohashi, and Makoto Nagao. 1998. General word sense disambiguation method based on a full sentential context. In *COLING-ACL '98 Workshop: Usage of WordNet in Natural Language Processing Systems*, Montréal, Canada, August.

R. Weischedel, M. Meteer, R. Schwartz, L. Ramshaw, and J. Palmucci. 1993. Coping with ambiguity and unknown words through probabilistic methods. *Computational Linguistics*, 19(2):359–382.

David Yarowsky. 1992. Word-sense disambiguation using statistical models of roget's categories trained on large corpora. In *Fourteenth International Conference on Computational Linguistics (COLING)*, pages 454–460.

David Yarowsky. 1994. Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 88–95.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196.