Parse Selection on the Redwoods Corpus: 3rd Growth Results

Kristina Toutanova, Christopher D. Manning, Stephan Oepen, Dan Flickinger
{kristina|manning|oe|danf}@csli.stanford.edu

Department of Computer Science and
Center for the Study of Language and Information
Stanford University
Stanford, CA 94305-9040, USA

October 14, 2003

Abstract

This report details experimental results of using stochastic disambiguation models for parsing sentences from the Redwoods treebank (Oepen et al., 2002). The goals of this paper are two-fold: (i) to report accuracy results on the more highly ambiguous latest version of the treebank, as compared to already published results achieved by the same stochastic models on a previous version of the corpus, and (ii) to present some newly developed models using features from the HPSG signs, as well as the MRS dependency graphs.

1 Introduction

The Redwoods corpus has undergone several iterations of improvements. The latest corpus version ("the third growth") was created in January 2003. The latest export of the third growth is from August 2003; it fixes some deficiencies in the elementary MRS[1] dependency graphs which were detected in the original export. The third growth is much more ambiguous and contains more sentences than the previous versions. The quality of the grammar is improved and many annotation errors are fixed.

For users of the treebank, it is very helpful to have performance figures as baselines and reference. One of the goals of this report is to summarize model performance figures on the latest version of the treebank as compared to previous ones. In addition to running already developed models, in this work we look at including additional information from the HPSG signs and the improved MRS dependency graphs representation. We first present generative and discriminative model performance on the old and new versions of the treebank. Then we describe the new features considered and present model performance using the new features.

2 Comparative Results

Table 1 lists the characteristics of version 1.5 and version 3 of the Redwoods treebank. All sentences listed here have exactly one preferred analysis in the Redwoods treebank.[2] For each of the two corpora versions, we have listed statistics for all sentences (ambiguous and unambiguous), and ambiguous only. In testing, we only consider ambiguous sentences, while unambiguous ones may be used in training. Version 1.5 of the corpus dates from June 2002, and was used in the experiments reported in the papers (Toutanova, Manning, Flickinger, & Oepen, 2002), (Oepen et al., 2002), and (Toutanova, Mitchell, & Manning, 2003).

Comparative results of the major previously described generative and discriminative models follow. All models were trained and tested using 10-fold cross-validation. Each of the 10 folds was formed deterministically (not at random), by starting from sentence i, and placing every 10th sentence in the test set. Thus the union of the 10 test sets, for $i = 1, \ldots, 10$ is the complete corpus and they do not overlap. The unambiguous sentences were discarded form the

[1] See (Copestake, Flickinger, Sag, & Pollard, 1999) for an introduction to Minimum Recursion Semantics.

[2] Some sentences have not been fully disambiguated (and have more then one preferred analysis), or because of grammar limitations have no analysis. Both types are excluded in the present work. In version 3 of the corpus, 6939 sentences have exactly one preferred analysis. Additionally, sentences of the corpus can be marked as grammatically ill-formed. These are also excluded from the present experiments. A small number of sentences (63) are marked as ill-formed, but nevertheless are marked as having one preferred analysis. Excluding these gives the $6939 - 63 = 6876$ sentences that we report in Table 1.

Version	Sentences		Length	Struct Ambiguity
1.5	all	5307	6.8	8.0
	ambiguous	3829	7.8	10.8
3.0	all	6876	8.0	44.5
	ambiguous	5266	9.1	57.8

Table 1: Annotated corpora used in experiments: The columns are, from left to right, corpus version, the total number of sentences, average length, and average structural ambiguity

Method		Accuracy	
		version 1.5	version 3
Random		25.81	22.75
Tagger	trigram	47.74	41.87
	perfect	54.59	43.80
PCFG	PCFG-1P	67.40	61.60
	PCFG-3P	77.57	70.66

Table 2: Performance of generative models for the parse selection task on the two corpora (accuracy).

Method		Accuracy	
		version 1.5	version 3
LTagger	trigram	48.70	39.24
	perfect	54.59	43.80
LPCFG	LPCFG-1P	79.30	72.38
	LPCFG-3P	81.23	74.87

Table 3: Performance of conditional log-linear models for the parse selection task (accuracy).

test sets. The generative models use the unambiguous sentences for training, but the conditional log-linear models do not (the unambiguous sentences contribute a constant to the log-likelihood).

2.1 Generative Models

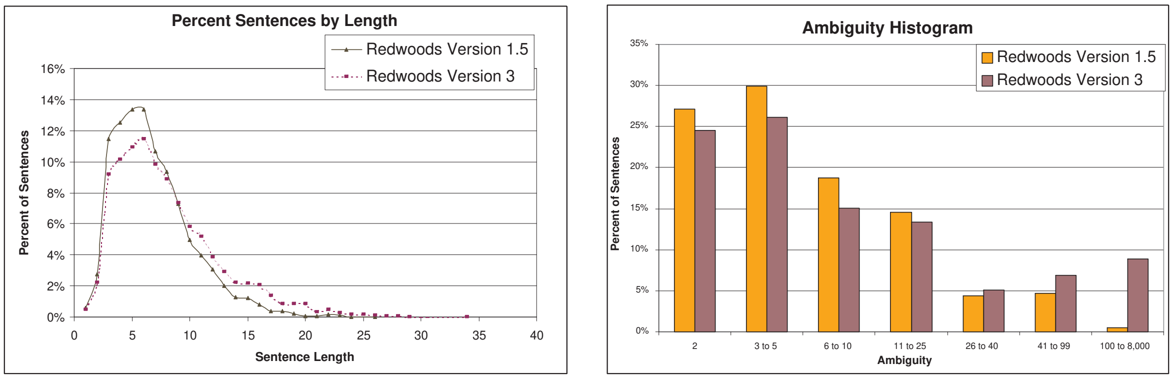
In Table 2, the parse selection accuracy of previously described models for the two corpus versions are listed. Most of the models are described at length in (Toutanova et al., 2002). A brief description follows here. Some accuracy figures differ slightly from the ones reported in (Toutanova et al., 2002). This is due to an improved version of smoothing in the generative models (add-α smoothing).

- Random is a model that assigns probability $1/m$ to each parse in a sentence with m parses.
- Tagger assigns probabilities to derivation trees by looking only at the preterminals (lexical labels) and the words of the trees.
- PCFG-1P is a simple PCFG model, equivalent to PCFG-S from (Toutanova et al., 2002). PCFG-3P is a model using the current node, its parent and its grandparent node as conditioning features.

From the table we can note that the greater structural ambiguity outweighs the improved quality of the treebank, and the results are considerably lower. Further discussion of the comparative results can be found in Section 3.

2.2 Conditional Log-linear Models

A conditional log-linear model for estimating the probability of an HPSG analysis given a sentence has a set of features $\{f_1, \ldots, f_m\}$ defined over analyses and a set of corresponding weights $\{\lambda_1, \ldots, \lambda_m\}$ for them.



(a) **(b)**

Figure 1: Percentage of sentences by length (a) and percentage of sentences by ambiguity level ranges (b) for versions 1.5 and 3 of the Redwoods corpus.

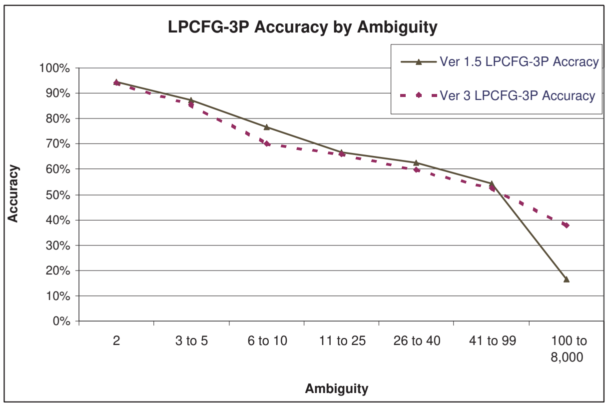


Figure 2: Accuracy of LPCFG-3P on the two corpora versions by ambiguity.

For a sentence s with possible analyses $t_1, \ldots, t_k$, the conditional probability for analysis t_i is given by:

$$P(t_i|s) = \frac{e^{\sum_{j=1\ldots m} f_j(t_i)\lambda_j}}{\sum_{i'=1,\ldots,k} e^{\sum_{j=1\ldots m} f_j(t_{i'})\lambda_j}} \quad (1)$$

For the three generative models described in the previous section, we built conditional log-linear models using the same features. We refer to the log-linear models as LTrigram, LPCFG-1P, and LPCFG-3P. These models correspond to the generative models Trigram, PCFG-1P and PCFG-3P respectively. Table 3 shows the accuracy of parse selection using the conditional log-linear models.

3 Discussion

It is useful to look at accuracy results from models on the two corpora versions depending on the ambiguity level of sentences. Since the 3rd growth is more ambiguous, the overall accuracy figures may be misleading. To illustrate the distribution of ambiguity levels in the two versions of the corpus, as well as the related distribution of number of words per sentence, Figure 1 shows histograms of the number of analyses per sentence in (b) as well as the percentage of sentences by sentence length in (a). Figure 2 shows the breakdown on accuracy of model LPCFG-3P for several sentence ambiguity categories. The two figures indicate that the major cause for the lower average accuracy on the third growth is the different distribution of number of analyses per sentence.

As can be seen from Figure 2, the accuracy of LPCFG-3P across most ambiguity levels is very similar for the two corpora versions. For sentences with more than 100 analyses the accuracy for version 3 is better. Overall, since the

No.	Path
1	synsem.local.cat.head.prd
2	synsem.local.cat.head.aux
3	sysnem.local.cat.head.vform
4	sysnem.local.cat.head.mod
5	sysnem.local.cat.val.comps
6	sysnem.local.cat.val.spr
7	sysnem.local.cat.val.subj
8	sysnem.nonlocal.slash
9	synsem.nonlocal.que
10	synsem.nonlocal.rel
11	synsem.local.cat.posthead
12	synsem.local.png.pn

Table 4: Feature Paths Selected from the HPSG Signs.

proportion of highly ambiguous sentences e.g., $\geq$ 26 analyses is much higher in Version 3 of the treebank (21% in Version 3 versus 9.6% in Version 1.5), the final average accuracy for Version 3 is much lower.

For the same ambiguity level, e.g., exactly 2 analyses, the accuracy for Version 1.5 is slightly better. Our initial hope was that the improved version of the treebank would be easier to disambiguate; however, the present results do not support this. This may indicate that more careful checking of the corpus actually leads to more cases where uncommon analyses are given to words or constructions.

4 Using Additional Features

By following paths in the feature structures of the HPSG signs, a large number of features are available at each node. To test the usefulness of various features, we performed an initial experiment using 12 feature paths.

These 12 feature paths were identified as promising by looking at statistics on how often failure of unification was due to these paths and by inspecting sentences for which the model made wrong predictions. The selected feature paths are listed in Table 4 (see (Pollard & Sag, 1994) for an explanation of almost all of these features).

For each of the selected feature paths, a conditional log-linear model was trained for which the features in the model were (*i*) all features from LPCFG-1P, and (*ii*) features that include the type of the feature structure which is the value of the specified path , e.g. if A$\rightarrow$B C is a local configuration in the derivation trees, A:VA$\rightarrow$ B:VB C:VC will be a feature in the model; VA, VB, and VC are the types of the values of the selected path at the corresponding nodes in the HPSG sign.

An analysis of useful distinctions between values was not performed. One would expect that looking at the types of the feature structures at the selected paths would often be too coarse. For example, if the value is a list, the type will only tell us whether the list is empty or not, rather than the number of elements or the types of the elements themselves. (Examples of list-valued features are the specifiers, complements, and subject features.)

Table 5 shows the accuracy for each of the paths. The baseline LPCFG-1P is also included. Many of the features performed at similar levels. Most helpful were features 1 (synsem.local.cat.head.prd), 7 (sysnem.local.cat.val.subj), 8 (sysnem.nonlocal.slash), and 10 (synsem.nonlocal.rel). If the effect of adding multiple of these features is nearly as good as adding them in isolation, we could expect big improvements from incorporating multiple feature paths.

To test this hypothesis, we trained a conditional log-linear model that contains the features of LPCFG-1P and feature paths 1, 7, 8, and 10 (the best-performing ones in bold in Table 5). The accuracy of this model was 74%. This is a promising result, but lower than might be expected. In future research, we will explore feature selection and methods of model combination using this vast space of available feature-path values.

Model	Accuracy
LPCFG-1P	72.38
LPCFG-1P + feat 1	**73.54**
LPCFG-1P + feat 2	73.03
LPCFG-1P + feat 3	73.35
LPCFG-1P + feat 4	72.78
LPCFG-1P + feat 5	73.34
LPCFG-1P + feat 6	73.18
LPCFG-1P + feat 7	**73.47**
LPCFG-1P + feat 8	**73.46**
LPCFG-1P + feat 9	73.04
LPCFG-1P + feat 10	**73.46**
LPCFG-1P + feat 11	72.83
LPCFG-1P + feat 12	72.95
LPCFG-1P + feats 1, 7, 8, 10	74.01

Table 5: Accuracy of Models Augmented by Feature Paths.

Model	Acc Version 1.5	Acc Version 3
LDep	67.44%	61.55%

Table 6: Accuracy of log-linear model over elementary dependency graphs for the two corpus versions.

5 Elementary Dependency Graphs

Here we report on a simple experiment of using elementary dependency graphs for disambiguation. We built a log-linear model over the dependency graphs corresponding to sentence analyses.

The features included in the model were very simple. For every elementary predication of the form:

label:rel_name[ARG1 label_1:rel_name_1, ... ARGK label_k:rel_name_k]

we added the following two feature kinds:

- [rel_name ARG1 ... ARGK]
- [rel_name ARG1 rel_name_1] ... [rel_name ARGK rel_name_k]

For example, for the elementary predication e2:_want2_rel[ARG1 x4:pron_rel, ARG4 _2:hypo_rel], the features (*i*) [_want2_rel ARG1 ARG4] and (*ii*) [_want2_rel ARG1 pron_rel] [_want2_rel ARG4 hypo_rel] were added.

As for the other conditional log-linear models we have built, the value of a feature for an analysis is equal to the number of times the specified configuration occurs in the elementary dependency graph.

Table 6 shows the accuracy of this simple model on the two versions of the corpus. The accuracy is lower than that of the simplest conditional log-linear model over derivation trees. Therefore more work is needed for finding a good semantic model. The difference in model accuracy between the two corpora versions is similar to the difference for other models.

6 Error Analysis

We performed a more extensive error analysis for the model LPCFG-3P on all errors in one of the CDs – CD32. Since the annotation consistency of the third growth was improved, we hoped that the fraction of errors due to wrong annotation would diminish compared to growth 1.5, and this was indeed the case.

For the sentences from CD32, looking across folds, the model made an error in parse selection 165 times. The error analysis suggests the following breakdown:

- For about 26% of errors, the annotation in the treebank was wrong
- For about 12% of the errors, both the treebank and the model were wrong
- About 62% of the errors were real errors and we could hope to get them right

The number of annotation errors is down to 26% from the previously reported 50% figure in (Toutanova et al., 2002). The number of errors due to problems with the grammar is also down. This shows the quality of the treebank is indeed improved. Correspondingly, the percent of real errors is up from 20% to 62%.

A more detailed break-down of the real errors (103 out of 165), in which the treebank was right and the model was wrong, follows:

- 27 (about 26.2%) are PP-attachment errors.
- 21 (about 20.4%) are errors in choosing the correct lexical sign.
- 15 (about 14.6%) are other modifier attachment errors.
- 13 (about 12.6%) are coordination errors.
- 9 (about 8.7%) are errors in the complement/adjunct distinctions.
- 18 (about 17.5%) are other errors.

The types of most common errors are similar to the ones observed in Penn Treebank parsing. Since the Redwoods treebank makes finer-grained distinctions, there are additional error types.

The model LPCFG-3P is unlexicalized. Even though in the past we have not had great improvements from lexicalization, it is a promising approach for, for example, the PP and other modifier attachment errors. We plan to explore different ways/amounts of lexicalization.

The values of features in the HPSG signs should be helpful for resolving complement/adjunct ambiguity and propagating coordination information. We have just started to experiment with these features as reported in Section 4, and will continue to work further along these lines.

7 Acknowledgements

This work was carried out under the Edinburgh-Stanford Link programme, funded by Scottish Enterprise, ROSIE project R36763.

References

Copestake, A., Flickinger, D. P., Sag, I. A., & Pollard, C. (1999). *Minimal Recursion Semantics. An introduction.* in preparation, Center for the Study of Language and Information, Stanford, CA.

Oepen, S., Toutanova, K., Shieber, S., Manning, C., Flickinger, D., & Brants, T. (2002). The LinGo Redwoods treebank: Motivation and preliminary apllications. In *COLING 19.* Taipei, Taiwan.

Pollard, C., & Sag, I. A. (1994). *Head-driven phrase structure grammar.* University of Chicago Press.

Toutanova, K., Manning, C. D., Flickinger, D., & Oepen, S. (2002). Parse disambiguation for a rich HPSG grammar. In *Treebanks and Linguistic Theories.* Sozopol, Bulgaria.

Toutanova, K., Mitchell, M., & Manning, C. (2003). Optimizing local probability models for statistical parsing. In *Proceedings of the 14th European Conference on Machine Learning (ECML).* Dubrovnik, Croatia.