# Improved Statistical Language Models

Eugene Charniak
Department of Computer Science, Brown University

**CONTACT INFORMATION**
Department of Computer Science
Brown University Box 1910
Providence RI 02912-1910

ec@cs.brown.edu
(401) 863 7601 (phone), -7657 (fax)

**WWW PAGE** http://www.cs.brown.edu/people/ec/home.html

**PROGRAM AREA** Speech and Natural Language Understanding

**KEYWORDS:** Syntactic Parsing, Statistical Language Processing, Language Models, Syntactic Disambiguation, Language Learning, Corpus-based Linguistics

**PROJECT SUMMARY**

As explained in the "area background," a good way to think about syntactic disambiguation is to find the parse $\pi$ which is most probable for the sentence $s$. As noted there, this is equivalent to maximizing $p(s, \pi)$.

Our research seeks to find new and better ways of computing this quantity, and then to see how this influences the syntactic disambiguation choices. We start with a comparatively simple probabilistic context-free grammar (PCFG) automatically induced from the Brown Corpus using a program developed earlier. While PCFGs assign probabilities to parses and thus can compute $p(s, \pi)$, generally they serve only to distinguish common from uncommon constructions, and when things get complicated they typically provide only modest help. Figure 1 gives a bad case. In particular note the bad prepositional phrase "of bankers Pres. Reagan said profits." While some prepositions, e.g. "because," introduce sentences, "of" does not. Thus we might improve this parse if our system knew the probabilities of rules conditioned on the key word in the phrase, such as the preposition in a prepositional phrase. In our work we have gathered statistics on this and many other lexical relations in order to do a better job of parsing.

A nice feature of our general approach is that we have an independent measure of the goodness of a particular probability model, its *per-word cross entropy*. We do not go into this here, but simply note that we would expect a good probability model tested on a representative sample of English to assign a higher probability to the corpus than a poor one. (After all, we should assign high probability to things we are likely to see, and if this is a representative corpus, it should contain lots of things we are likely to see.) Cross entropy is defined so that it goes down as the probability goes up: lower cross entropy is equivalent to a better probability model. Our current results show that by this measure, as the probability

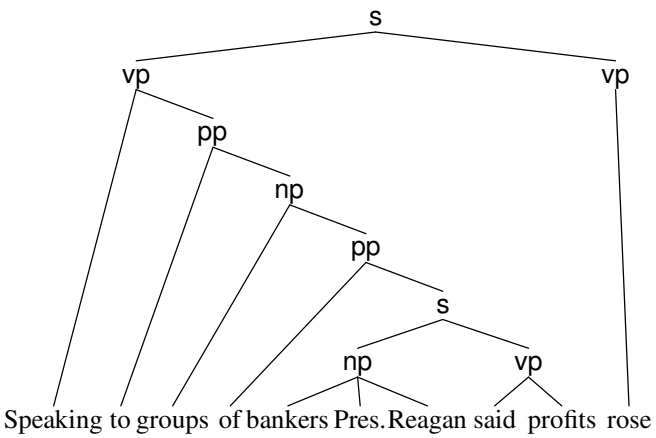


Figure 1: A badly parsed sentence

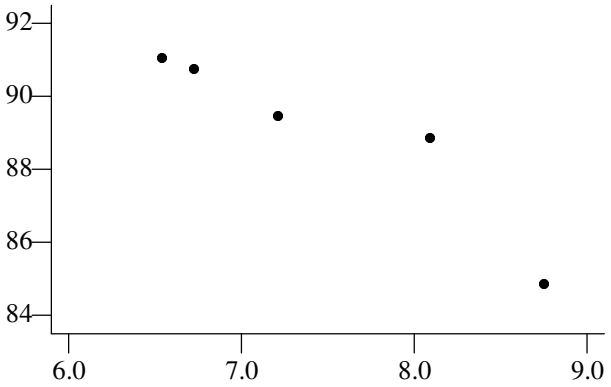


Figure 2: Percentage parsing accuracy as a function of cross entropy

model gets better, the syntactic disambiguation gets better as well. Figure 2 plots our results of parsing accuracy as a function of cross entropy. The connection we postulated clearly holds for our data. The simple PCFG model used as the starting point of the research corresponds to the poorest of the models shown here, and the full model is the best. The three intermediate levels correspond to deleting portions of the full model.

We are also interested in the absolute values of the parsing accuracy. While our best accuracy (91.1%) is reasonably good, it is not quite as good as the best values reported in the literature. At the moment we have no explanation for these differences. (Ultimately we are also interested in the absolute values of the per-word cross entropy, but those familiar with this measure should note that the experiment behind Fig. 2 re-

stricted consideration to sentences with only high-frequency words, so the absolute cross-entropy values are not for general English.)


**REFERENCES**

1. Carroll, G. and Charniak, E., Two experiments on learning probabilistic dependency grammars from corpora. In *Workshop Notes, Statistically-Based NLP Techniques*, AAAI (1992) 1-13

2. Charniak, E. and Carroll, G., Context-sensitive statistics for improved grammatical language models. In *Proc. Twelfth National Conference on Artificial Intelligence*, Menlo Park: AAAI Press/MIT Press (1994) 728-733

3. Charniak, Eugene, *Parsing with context-free grammars and word statistics*. TR CS-95-28, Dept. of Computer Science, Brown University (1995)

4. Charniak, E., Carroll, G., Adcock, J., Cassandra, A., Gotoh, Y., Katz, J., Littman, M., and McCann, J., Taggers for parsers. *Artificial Intelligence* (forthcoming)

5. Ersan, M., and Charniak, E., *A statistical syntactic disambiguation program and what it learns*. TR CS-95-29 Brown University, Dept. of Computer Science, (1995) (To appear in *Symbolic, Connectionist, and Statistical Approaches to Learning for Natural Language Processing*. S. Wermter, E. Riloff, and G. Scheler Eds., New York: Springer (forthcoming))


**AREA BACKGROUND**

The area in question is a cross between the language modeling work common in the speech-recognition community and the language understanding work common in the AI community. From language modeling work we take the idea of a *language model*: a mechanism that assigns a probability to each string in a language. (For our purposes we take such strings to be sentences of the language.) So, with a sentence $s$ the language model associates a probability $p(s)$. If the language model is a good one then relatively common sentences like "I am late" are given a higher probability than less common ones like "I crashed into people," which in turn should be more probable than ungrammatical strings like "People into crashed I." People not familiar with such work initially have two qualms: first, how can one assign such probabilities, and second, why would one want to do so in the first place.

As for how to assign such probabilities, if you are willing to do a poor job it is easy. To make things slightly easier, assume we have a complete list of English words. If there are $n$ of them and we assign each a probability of $1/n$ and make the probability of a sequence of words the product of the probability of each word, we are done. We could do better by collecting large amounts of text and getting a better estimate of the probability of individual words. We can do still better if, as in speech recognition systems, we condition each probability on the last two words. For example, in the sentence "I crashed into people" we would need the probability of "into" given that the last two words are "I crashed." This, of course, would be reasonably high, whereas in the garbled sentence the probability of "crashed" given the last two words "people into" would be very low.

But why care about language models? There are two sorts of answers. One is that for certain tasks like speech recognition they are important. We ignore this answer. The second is that they suggest new and interesting ways to perform tasks more common in AI, such as syntactic disambiguation. Syntactic disambiguation is the process of finding the intended syntactic structure of a sentence. In Fig. 1 the structure found is far from that intended. To see how language models can help in disambiguation, suppose that somehow (we discuss how in the "project summary") we can calculate the probability of a sentence $s$ under a particular parse $\pi$, $p(s, \pi)$. Then let $\pi$ vary over all the parses of the sentence $s$:

$$p(s) = \sum_{\pi} p(s, \pi) \tag{1}$$

That is, we get our language model by summing the probability of the sentence over all of its possible parses.

What is interesting about this is that the term $p(s, \pi)$ is a crucial one for syntactic disambiguation, since a good way to disambiguate a sentence is to find the parse that maximizes this quantity. To see this, agree that we want to find the parse which maximizes $p(\pi \mid s)$, the most probable parse given the sentence. But by Bayes' law this is equivalent to maximizing $p(s, \pi)$:

$$\arg\max_{\pi} p(\pi \mid s) = \arg\max_{\pi} \frac{p(\pi, s)}{p(s)} \tag{2}$$

$$= \arg\max_{\pi} p(\pi, s) \tag{3}$$

Here $\arg\max_{\pi}$ says find the $\pi$ that maximizes the quantity that follows. Equation 2 just uses Bayes' law, while 3 depends on the fact that $p(s)$ remains constant over all parses, so it can be ignored when computing the maximum. The research upon which I am engaged starts from this last equation.


**REFERENCES**

1. Bod, A. R., Using an annotated language corpus as a virtual stochastic grammar. In *Proc. Eleventh National Conference on Artificial Intelligence*, Menlo Park: AAAI Press/MIT Press (1993) 778-783

2. Charniak, E., *Statistical Language Learning*. Cambridge: MIT Press (1993)

3. Church, K. W., and Mercer, R. I. (Eds.), Special issue on using large corpora. *Computational Linguistics* **19** Nos 1,2 (1993)

4. Magerman, D. M., Statistical decision-tree models for parsing. In *Proc. 33rd Annual Meeting of the Association for Computational Linguistics* (1995) 276-283

5. Yarowsky, D., and Church, K. (Eds.), *Proc. Third Workshop on Very Large Corpora*. (1995)


**RELATED PROGRAM AREAS** Adaptive Human Interfaces, Intelligent Interactive Systems for Persons with Disabilities