AUTOMATICALLY ACQUIRING PHRASE STRUCTURE USING DISTRIBUTIONAL ANALYSIS

*Eric Brill and Mitchell Marcus**

Department of Computer Science
University of Pennsylvania
Philadelphia, Pa. 19104
brill@unagi.cis.upenn.edu, mitch@unagi.cis.upenn.edu

ABSTRACT

In this paper, we present evidence that the acquisition of the phrase structure of a natural language is possible without supervision and with a very small initial grammar. We describe a language learner that extracts distributional information from a corpus annotated with parts of speech and is able to use this extracted information to accurately parse short sentences. The phrase structure learner is part of an ongoing project to determine just how much knowledge of language can be learned solely through distributional analysis.

1. INTRODUCTION

This paper is an exploration into the possibility of automatically acquiring the phrase structure of a language. We use distributional analysis techniques similar to the techniques originally proposed by Zellig Harris [5] for structural linguists to use as an aid in uncovering the structure of a language. Harris intended his techniques to be carried out by linguists doing field work, as a substitute for what he perceived as unscientific information gathering by linguists at the time. The procedures Harris describes are intended to uncover "regularities [...] in the distributional relations among the features of speech in question" (page 5). To use distributional analysis to determine empirically whether *boy* and *girl* are in the same word class, the linguist would need to determine whether the two words are licensed to occur in the same environments. Harris presented algorithms linguists could use to detect distributionally similar entities.

Harris did not intend the procedures he proposed to be used as a model of child language acquisition or as a tool for computerized language learning. This would not be feasible because the method Harris describes for determining distributional similarity does not seem amenable to unsupervised acquisition. One way of determining whether *boy* and *girl* are in the same word class is to see whether it is the case that for all sentences that *boy* occurs in, the same sentence with *girl* substituted for *boy* is an allowable sentence. To do this automatically from text, one would need a prohibitively large corpus. This lack of sufficient data does not arise in field work because the linguist has access to informants, who are in effect infinite corpora. If one hears *the boy finished the homework*, the informant can be queried whether *the girl finished the homework* is also permissible.

The procedures Harris outlines for the linguist to use to discover linguistic structure could be used to automatically acquire grammatical information if it were possible to do away with the need for a human informant. It is possible that a variation of these procedures could extract information by observing distributional similarities in a sufficiently large corpus of unparsed text. In an earlier paper [2], we demonstrated that simple distributional analysis over a corpus can lead to the discovery of word classes. In this paper, we describe work in which we apply distributional analysis in an attempt to automatically acquire the phrase structure of a language.

We describe a system which automatically acquires English phrase structure, given only the tagged Brown Corpus [4] as input. The system acquires a context-free grammar where each rule is assigned a score. Once the grammar is learned, it can be used to find and score phrase structure analyses of a string of part of speech tags. The nonterminal nodes of the resulting phrase structure tree are not labelled. The system is able to assign a phrase structure analysis consistent with the string of part of speech tags with high accuracy.

There have been several other recent proposals for automatic phrase structure acquisition based on statistics gathered over large corpora. In [1, 9], a statistic based on mutual information is used to find phrase boundaries. [11] defines a function to score the quality of parse trees, and then uses simulated annealing to heuristically explore the entire space of possible parses for a given sentence. A number of papers describe results obtained using the Inside-Outside algorithm to train a probabilistic context-free grammar [10, 6, 8]. Below we describe an alternate method of phrase structure acquisition.

*This work was supported by DARPA and AFOSR jointly under grant No. AFOSR-90-0066, and by ARO grant No. DAAL 03-89-C0031 PRI.

2. HOW IT WORKS

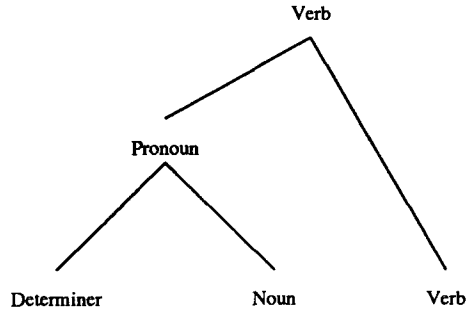
The system automatically acquires a grammar of scored context-free rules, where each rule is binary branching. Two sources of distributional information are used to acquire and score the rules. The score for the rule $tag_x \rightarrow tag_y\ tag_z$ is a function of:

1. The distributional similarity of the part of speech tag tag_x and the pair of tags $tag_y\ tag_z$.
2. A comparison of the entropy of the environment tag_y _ and $tag_y\ tag_z$ _. The entropy of environment tag_y _ is a measure of the randomness of the distribution of tags occurring immediately after tag_y in the corpus.

2.1. Substitutability

The system is based upon the assumption that if two adjacent part of speech tags are distributionally similar to some single tag, then it is probable that the two tags form a constituent. If tag_x is distributionally similar to $tag_y\ tag_z$, then tag_x can be substituted for $tag_y\ tag_z$ in many environments. If a single tag is substitutable for a pair of adjacent tags, it is highly likely that that pair of tags makes up a syntactically significant entity, i.e. a phrase.

For example, words labelled with the tag *Pronoun* and words labelled with the tag pair *Determiner Noun* are distributionally similar. Distributionally, *Pronoun* can occur in almost all environments in which *Determiner Noun* can occur. In the tag sequence *Determiner Noun Verb*, we could discover that *Determiner Noun* is a constituent and *Noun Verb* is not, since no single lexical item has distributional behavior similar to the pair of tags *Noun Verb*. Once we know these distributional facts, as well as the fact that the single tag *Verb* and the tag pair *Pronoun Verb* distribute similarly (*eat* fish :: *we eat* fish), we can find the structure of the tag sequence *Determiner Noun Verb* by recursively substituting single part of speech tags for pairs of tags. This would result in the structurally correct (ignore the nonterminal labels):



To carry out the above analysis, we made use of our knowledge of the language to determine that the tag *Pronoun* is distributionally similar to (substitutable for) the pair of tags *Determiner Noun*. Unfortunately, the system does not have access to such knowledge. However, an approximation to this knowledge can be learned. For each possible context-free rule $tag_x \rightarrow tag_y\ tag_z$, the system assigns a value indicating the distributional similarity of tag_x to the pair of tags $tag_y\ tag_z$. The measure used to compute the similarity of tag_x to $tag_y\ tag_z$ is known as *divergence* [7].

Let P_1 and P_2 be two probability distributions over environments. The relative entropy between P_1 and P_2 is:

$$D(P_1||P_2) = \sum_{x \in Environments} P_1(x) * log\frac{P_1(x)}{P_2(x)}$$

Relative entropy $D(P_1||P_2)$ is a measure of the amount of extra information beyond P_2 needed to describe P_1. The *divergence* between P_1 and P_2 is defined as $D(P_1||P_2) + D(P_2||P_1)$, and is a measure of how difficult it is to distinguish between the two distributions. Two entities will be considered to distribute similarly, and therefore be substitutable, if the divergence of their probability distributions over environments is low. In part, this work is an attempt to test the claim that a very local definition of environment is sufficient for determining distributional similarity.[1]

We will now describe how we can use the distributional similarity measure to extract a binary context-free grammar with scored rules from a corpus. Statistics of the following form are collected:

1. $word_1\ tag_x\ word_2\ number$
2. $word_1\ tag_y\ tag_z\ word_2\ number$

where in (1), *number* is the number of times in the corpus the word between words $word_1$ and $word_2$ is tagged with tag_x, and in (2), *number* is the number of times that the pair of words between $word_1$ and $word_2$ is tagged with tag_y, tag_z. For instance, in the Brown Corpus, the part of speech tag NP[2] appears between the words *gave* and *a* three times, and the tags $AT\ NN$[3] occur six times in this environment.

[1] Evidence that this claim is valid for word class discovery is presented in [1, 2, 3].

[2] NP = proper noun.

[3] AT = article, NN = sing. noun.

From this, we obtain a set of context-free rules $tag_x \rightarrow tag_y\ tag_z$, scored by the distributional similarity of tag_x and $tag_y\ tag_z$. The score given to the rule is the divergence between the probability distributions of tag_x and $tag_y\ tag_z$ over environments, where an environment is of the form **word ___ word**.

Below are the five single tags found to be distributionally most similar to the pair of tags AT NN, found by measuring divergence of distributions over the environments **word ___ word**: .

1. NP (Proper Noun)
2. CD (Number)
3. NN (Sing. Noun)
4. NNS (Plural Noun)
5. PPO (Object Personal Pronoun)

Of all rules with AT NN on the right hand side, the rule $NP \rightarrow AT\ NN$ would be given the best score. Below are the five tag pairs found to be closest to the single tag NP. Of all rules with NP on the left hand side, $NP \rightarrow NP\ NP$ is given the best score.

1. NP NP (Robert/NP Snodgrass/NP)
2. PP$ NN (his/PP$ staff/NN)
3. NN NNS (city/NN employees/NNS)
4. NP$ NN (Gladden's/NP$ wife/NN)
5. AT NN (the/AT man/NN)

Once the scored context-free grammar is learned, there are a number of ways to use that grammar to search for the correct phrase structure analysis of a sentence. For the results reported at the end of the paper, we used the simplest method: find the best set of rules that allow the part of speech string to be reduced to a single part of speech. The best set is that set of rules whose scores sum to the lowest number. In other words, we search for the set of rules with the lowest total divergence between the pair of tags on the right hand side of the rule and the single tag these two tags will be reduced to. The structure assigned by this set of rules, ignoring nonterminal labels, is output as the structural description of the sentence.

2.2. Adjusting Scores

The scored CFG described above works fairly well, but makes a number of errors. There are a number of cases where a phrase is posited when the pair of symbols do not really constitute a phrase. For instance, *VBD* and *VBD IN*[4] have similar distributional behavior. (John and Mary **kissed/VBD in/IN** the car *vs.* John and Mary **bought/VBD** the car). If we had access to lexical information, this would not be a problem. The problem results from discarding the lexical items and replacing them with their part of speech tags. If we are to continue our analysis on part of speech tags, a different information source is needed to recognize problematic rules such as **VBD → VBD IN** which are incorrectly given a good score. We extract more n-gram statistics, this time of the form:

1. $tag_x\ tag_y\ number$
2. $tag_x\ tag_y\ tag_z\ number$

which is a file of pairs and triples of part of speech tags and the number of times the tag strings occur in the corpus. The entropy of the position after tag_x in the corpus is a measure of how constrained that position is. This entropy (H) is computed as:

$$H(tag_x\_) = -\sum_{tag_y \in TagSet} p(tag_y \mid tag_x) * log_2 p(tag_y \mid tag_x)$$

Likewise, we can compute the entropy of the position following the pair of tags tag_x and tag_y. If $tag_x\ tag_y$ is indeed a constituent, we would expect:

$$H(tag_x\ \_) < H(tag_x\ tag_y\ \_)$$

This is because a phrase internal position in a sentence is more constrained as to what can follow than a phrase boundary position. We can use this information to readjust the scores in the grammar. The score of each rule of the form $tag_z \rightarrow tag_x\ tag_y$ is multiplied by a function of $Entropy(tag_x\ tag_y\ \_) - Entropy(tag_x\ \_)$, to reward those rules for which the entropy-based metric indicates that they span a true constituent and to penalize those involving nonconstituents. For instance, the measure $Entropy(tag_x\ tag_y\ \_) - Entropy(tag_x\ \_)$ has a value of 1.4 for the pair of tags *AT NN*[5], and a value of -0.8 for the pair of tags *VBD IN*, the troublesome tag pair mentioned above.

[4] VBD = past verb, IN = preposition.

[5] AT NN = Determiner Noun - a true phrase.

At this point the learner makes one major mistake on short sentences. Sometimes, but not always, the subject or some part of the subject is joined to the verb before the object is. For example, the system assigns a slightly better score to the parse ((PPS VBD) PPO)[6] than to the correct parse (PPS (VBD PPO)). To remedy this, we need a rule specifying that a matrix verb must join with its object before joining with its subject.

3. RESULTS

After running this learning procedure on the Brown Corpus, a grammar of 41,000 rules was acquired. We took a subset of these rules (about 7,500), choosing the fifteen best scoring rules for all tag pairs appearing on the right hand side of some rule.

The parser is given a string of part of speech tags as input and uses its automatically acquired grammar to output an unlabelled binary-branching syntactic tree for the string. Since lexical information is thrown away, a correct answer is considered to be an analysis that is consistent with the tag set. The goal of this work is to automatically create from a tagged corpus a corpus of simple sentences annotated with phrase structure. In the next phase of the project, we plan to extract a richer grammar from the corpus of trees. Therefore, we were not concerned when no answer was returned by the parser, as long as this did not happen with high probability. If the parser fails to parse a sentence, that sentence would not be present in the corpus of trees. However, if the parser incorrectly parses a sentence, the error will be entered into the corpus. The higher the error rate of this corpus, the more difficult the next stage of acquisition would be.

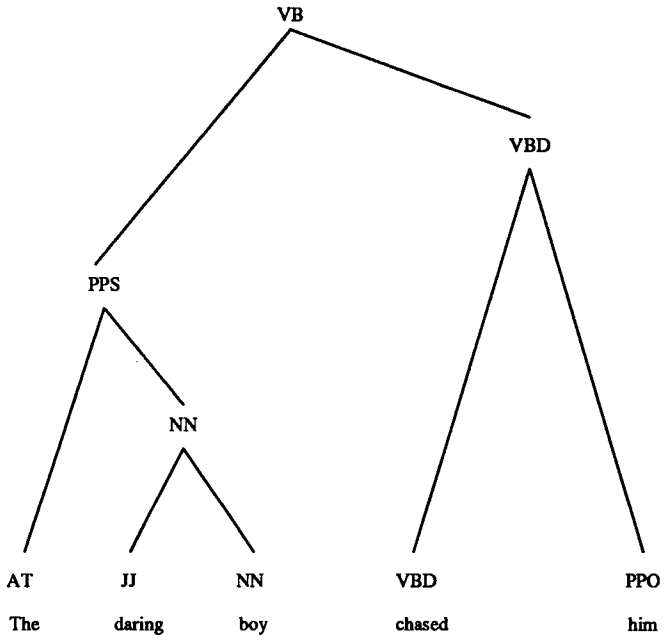
The table below shows the results obtained by testing the system on *simple* sentences. A *simple* sentence is defined as a sentence with between five and fourteen words, containing no coordinates, quotations, or commas.

	Correct	*Close*	*Wrong*
No Unparsed Sents	71%	11%	18%
With Unparsed Sents	62%	10%	28%

Table 1: Summary of Acquisition and Parsing Accuracy

In the table, *correct* means that the parse was a valid parse for the string of tags, *close* means that by performing the operation of moving one bracket and then balancing brackets, the parse can be made correct. *Wrong* means that the parse was more than one simple operation away from being correct. Of all test sentences, 15% were not parsed by the system. Of those sentences, many failed because the beam search we implemented to speed up parsing does not explore the entire space of parses allowed by the grammar. Presumably, many of these sentences could be parsed by widening the beam when a sentence fails to parse.

One question that remains to be answered is whether there is a way to label the nonterminals in the trees output by the system. The tree below was given the best score for that particular part of speech tag sequence.



If all part of speech tags are assigned a particular nonterminal label (PPS and NN would be classed as **NP**. VB, VBD would be classed as **VP**)[7] and replaced the tags with their nonterminal labels, we would get a properly labelled tree for the above structure. It remains to be seen whether this idea can be extended to accurately assign nonterminal labels to the trees output by the parser.

4. CONCLUSION

We believe that these results are evidence that automatic phrase structure acquisition is feasible. In addition to the problem of labelling nonterminals, we are currently working on expanding the learner so it can handle more complex sentences and take lexical information into account when parsing a sentence.

[6]PPS = subject pers. pron., VBD = past verb, PPO = obj. pers. pron.

[7]PPS = 3rd sing. nom. pronoun, NN = sing. noun, VB = verb, VBD = past verb

References

1. Brill, E., Magerman, D., Marcus, M., and Santorini, B. (1990) Deducing linguistic structure from the statistics of large corpora. In *Proceedings of the DARPA Speech and Natural Language Workshop*, Morgan Kaufmann, 1990.
2. Brill, Eric. (1991) Discovering the lexical features of a language. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, Berkeley, CA.
3. Brown, P., Della Pietra, V., Della Pietra, S. and Mercer, R. (1990) Class-based n-gram models of natural language. In *Proceedings of the IBM Natural Language ITL, pp. 283-298*, Paris, France.
4. Francis, W. Nelson and Kučera, Henry, *Frequency analysis of English usage. Lexicon and grammar.* Houghton Mifflin, Boston, 1982.
5. Harris, Zellig. (1951) *Structural Linguistics.* Chicago: University of Chicago Press.
6. Jelinek, F., Lafferty, J., and Mercer, R. (1990) Basic methods of probabilistic context free grammars. Technical Report RC 16374 (72684), IBM, Yorktown Heights, New York 10598.
7. Kullback, Solomon. (1959) *Information Theory and Statistics.* New York: John Wiley and Sons.
8. Lari, K. and Young, S. (1990) The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56.
9. Magerman, D. and Marcus, M. (1990) Parsing a natural language using mutual information statistics, *Proceedings, Eighth National Conference on Artificial Intelligence (AAAI 90)*, 1990.
10. Pereira, F. and Schabes, Y. (1992) Inside-outside reestimation from partially bracketed corpora. Also in these proceedings.
11. Sampson, G. (1986) A stochastic approach to parsing. In *Proceedings of COLING 1986*, Bonn.