PCFG models of linguistic tree representations

Mark Johnson*
Brown University

The kinds of tree representations used in a tree bank corpus can have a dramatic effect on performance of a parser based on the PCFG estimated from that corpus, causing the estimated likelihood of a tree to differ substantially from its frequency in the training corpus. This paper points out that the Penn II tree bank representations are of the kind predicted to have such an effect, and describes a simple node relabelling transformation which improves a tree bank PCFG based parser's average precision and recall by around 8%, or approximately half of the performance difference between a simple PCFG model and the best broad coverage parsers available today. This performance variation comes about because any PCFG, and hence the corpus of trees from which the PCFG is induced, embodies independence assumptions about the distribution of words and phrases. The particular independence assumptions implicit in a tree representation can be studied theoretically and investigated empirically by means of a tree transformation-detransformation process.

1 Introduction

Probabalistic Context Free Grammars (PCFGs) provide simple statistical models of natural languages. The relative frequency estimator provides a straight forward way of inducing these grammars from tree bank corpora, and a broad coverage parsing system can be obtained by using a parser to find a maximum likelihood parse tree for the input string with respect to such a tree bank grammar. PCFG parsing systems often perform as well as other simple broad coverage parsing system for predicting tree structure from Part of Speech (POS) tag sequences (Charniak, 1996). While PCFG models do not perform as well as models which are sensitive to a wider range of dependencies (Collins, 1996), their simplicity makes them straight forward to analyse both theoretically and empirically. Moreover, since more sophisticated systems can be viewed as refinements of

* Department of Cognitive and Linguistic Sciences, Box 1978, Providence, RI 02912

the basic PCFG model (Charniak, 1997), it seems reasonable to first attempt to better understand the properties of PCFG models themselves.

It is well known that natural language exhibits dependencies that Context Free Grammars (CFGs) cannot describe (Culy, 1985; Shieber, 1985). But the statistical independence assumptions embodied in a particular PCFG description of a particular natural language construction are in general much stronger than the requirement that the construction be generated by a CFG. We show below that the PCFG extension of what seems to be an adequate CFG description of PP attachment constructions performs no better than PCFG models estimated from non-CFG accounts of the same constructions.

More specifically, this paper studies the effect of varying the tree structure representation of PP modification from both a theoretical and an empirical point of view. It compares PCFG models induced from tree banks using several different tree representations, including the representation used in the Penn II tree bank corpora (Marcus, Santorini, and Marcinkiewicz, 1993) and the "Chomsky adjunction" representation now standardly assumed in generative linguistics.

One of the weaknesses of a PCFG model is that it is insensitive to non-local relationships between nodes. If these relationships are significant then a PCFG will be a poor language model. Indeed, the sense in which the set of trees generated by a CFG is "context free" is precisely that the label on a node completely characterizes the relationships between the subtree dominated by the node and the nodes that properly dominate this subtree.

Roughly speaking, the more nodes in the trees of the training corpus, the stronger the independence assumptions in the PCFG language model induced from those trees. For example, a PCFG induced from a corpus of completely flat trees (i.e., consisting of the root node immediately dominating a string of terminals) generates precisely the strings of training corpus with likelihoods equal to their relative frequencies in that corpus. Thus the location and labelling on the nonroot nonterminal nodes determine how a PCFG induced from a tree bank generalizes from that training data. Generally, one might expect that the fewer the nodes in the training corpus trees, the weaker the independence assumptions in the induced language model. For this reason, a "flat" tree representation of PP modification is investigated here as well.

A second method of relaxing the independence assumptions implicit in a PCFG is to encode more information in each node's label. Here the intuition is that the label on a node is a "communication channel" which conveys information between the subtree dominated by the node and the part of the tree not dominated by this node, so all other things being equal, appending additional information to the node's label about the context in which the node appears should make the independence assumptions implicit in the PCFG model weaker. The effect of adding a particularly simple kind of contextual information—the category of the node's parent—is also studied in this paper.

Whether either of these two PCFG models outperforms a PCFG induced from the original tree bank is a separate question. We face a classical 'bias versus variance' dilemma here (Geman, Bienenstock, and Doursat, 1992): as the independence assumptions implicit in the PCFG model are weakened, the number of parameters that must be estimated (i.e., the number of productions) increases. Thus while moving to a class of models with weaker independence assumptions permits us to more accurately describe a wider class of distributions (i.e., it reduces the *bias* implicit in the estimator), in general our estimate of these parameters will be less accurate simply because there are more of them to estimate from the same data (i.e., the *variance* in the estimator increases).

This paper studies the effects of these differing tree representations of PP modification theoretically by considering their effect on very simple corpora, and empirically by means of a tree transformation/detransformation methodology introduced below. The corpus used as the source for the empirical study is version II of the Wall Street Journal

(WSJ) corpus constructed at the University of Pennsylvania, modified as described in Charniak (1996), in that:

- root nodes (labelled 'ROOT') were inserted,
- the terminal or lexical items were deleted (i.e., the terminal items in the trees were POS tags),
- node labels consisted solely of syntactic category information, e.g., grammatical function and coindexation information was removed,
- the POS tag of auxiliary verbs was replaced with 'AUX',
- empty nodes (i.e., nodes dominating the empty string) were deleted, and
- any resulting unary branching nodes dominating a single child with the same node label (i.e., which are expanded by a production $X \to X$) were deleted.

2 PCFG models of tree structures

The theory of PCFGs is described elsewhere (e.g., Charniak (1993)), so it is only summarized here. A PCFG is a CFG in which each production $A \to \alpha$ in the grammar's set of productions P is associated with an emission probability $\mathrm{P}(A \to \alpha)$ that satisfies a normalization constraint

$$\sum_{\alpha : A \to \alpha \in P} \mathrm{P}(A \to \alpha) \; = \; 1$$

and a consistency or tightness constraint not discussed here, which PCFGs estimated from tree banks using the relative frequency estimator always satisfy (Chi and Geman, to appear).

A PCFG defines a probability distribution over the (finite) parse trees generated by the grammar, where the probability of a tree τ is given by

$$\mathrm{P}(\tau) \;=\; \prod_{A \to \alpha \in P} \mathrm{P}(A \to \alpha)^{\mathrm{C}_\tau(A \to \alpha)}$$

where $\mathrm{C}_\tau(A \to \alpha)$ is the number of times the production $A \to \alpha$ is used in the derivation τ.

The PCFG which assigns maximum likelihood to the sequence $\tilde{\tau}$ of trees in a tree bank corpus is given by the relative frequency estimator.

$$\hat{\mathrm{P}}_{\tilde{\tau}}(A \to \alpha) \;=\; \frac{\mathrm{C}_{\tilde{\tau}}(A \to \alpha)}{\sum_{\alpha' \in (N \cup T)^*} \mathrm{C}_{\tilde{\tau}}(A \to \alpha')}$$

Here $\mathrm{C}_{\tilde{\tau}}(A \to \alpha)$ is the number of times the production $A \to \alpha$ is used in derivations of the trees in $\tilde{\tau}$.

This estimation procedure can be used in a broad-coverage parsing procedure as follows. A PCFG G is estimated from a tree bank corpus $\tilde{\tau}$ of training data. In the work presented here the actual lexical items (words) are ignored, and the terminals of the trees are taken to be the Part of Speech (POS) tags assigned to the lexical items. Given a sequence of POS tags to be analysed, a dynamic programming method based on the CKY algorithm (Aho and Ullman, 1972) is used to search for a maximum likelihood parse using this PCFG.

3 Tree representations of linguistic constructions

For something so apparently fundamental to syntactic research, there is considerable disagreement among linguists as to just what the right tree structure analysis of various linguistic constructions ought to be. Figure 1 shows some of the variation in PP modification structures postulated in generative syntactic approaches over the past 30 years.

The flat attachment structure was popular in the early days of transformational grammar, and is used to represent VPs in the WSJ corpus. In this representation both

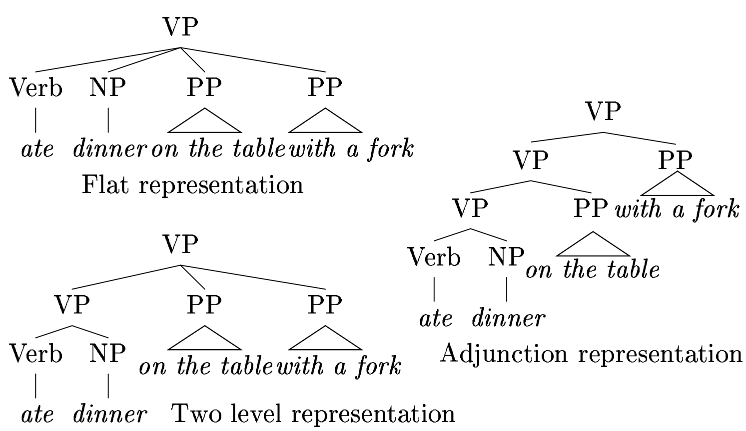


Figure 1
Different tree representations of PP modification.

arguments and adjuncts are sisters to the lexical head, and so are not directly distinguished in the tree structure.

The adjunction representation was introduced by Chomsky (it is often called "Chomsky adjunction"); in that representation arguments are sisters to the lexical head, while adjuncts are adjoined as sisters to a phrasal node: either a maximal projection (as shown in Figure 1) or a "1-bar" projection in the "X-bar" theory of grammar and its descendants.

The third representation depicted in Figure 1 is a mixed representation in which phrases with adjuncts have exactly two levels of phrasal projection. The lower level contains the lexical head, and all adjuncts are attached as sisters to a maximal projection at the higher level. To a first approximation, this is the representation used for NPs with PP modifiers or complements in the WSJ corpus used in this study.[1]

If the standard linguistic intuition that the number of PP modifiers permitted in natural language is unbounded is correct, then only the Chomsky adjunction representation trees can be generated by a CFG, as the other two representations depicted in Figure 1 require a different production for each possible number of PP modifiers. For example, the rule schema VP $\rightarrow$ V NP PP*, which generates the flat attachment structure, abbreviates an infinite number of CF productions.

In addition, if a tree bank using the two level representation contains at least one node with a single PP modifier, then the PCFG induced from it will generate Chomsky adjunction representations of multiple PP modification, in addition to the two level representations used in the tree bank. (Note that this is not a criticism of the use of

1 The Penn treebank annotation conventions are described in detail in Bies et al. (1995). The 'two level' representation arises from the conventions that "postmodifiers are Chomsky-adjoined to the phrase they modify" (11.2.1.1) and that "consecutive unrelated adjuncts are non-recursively attached to the NP they modify" (11.2.1.3.a) (parenthesis identify relevant subsections in Bies et al. (1995)). Arguments are not systematically distingushed adjunct PPs, and "only clausal complements of NP are placed inside [the innermost] NP" as a sister of the head noun. However, because certain constructions are encoded recursively, such as appositives, emphatic reflexives, phrasal titles, etc., it is possible for NPs with more than two levels of structure to appear.

this representation in a tree bank, but of modelling such a representation with a PCFG). This raises the question: how should a parse tree be interpreted that does not fit the representational scheme used to construct the tree bank training data?

As noted above, the WSJ corpus represents PP modification to NPs using the two-level representation. The PCFG estimated from sections 2–21 of this corpus contains the following two productions:

$$\hat{\mathrm{P}}(\mathrm{NP} \rightarrow \mathrm{NP\ PP}) = 0.112$$

$$\hat{\mathrm{P}}(\mathrm{NP} \rightarrow \mathrm{NP\ PP\ PP}) = 0.006$$

These productions generate the two-level representations of one and two PP adjunctions to NP, as explained above. However, the second of these productions will never be used in a maximum likelihood parse, as the the parse of sequence NP PP PP involving two applications of the first rule has a higher estimated likelihood.

In fact, *all* of the productions of the form NP $\rightarrow$ NP PP^n where $n > 1$ in the PCFG induced from sections 2–21 of the WSJ corpus are subsumed by the NP $\rightarrow$ NP PP production in this way. Thus PP adjunction to NP in the maximum likelihood parses using this PCFG always appear as Chomsky adjunctions, even though the original tree bank uses a two-level representation!

A large number of productions in the PCFG induced from sections 2–21 of the WSJ corpus are subsumed by higher likelihood combinations of shorter, higher probability productions. Of the 14,962 productions in the PCFG, 1,327 productions, or just under 9%, are subsumed by combinations of two or more productions.[2] Since the subsumed productions are never used to construct a maximum likelihood parse, they can be ignored if only maximum likelihood parses are required. Moreover, since these subsumed

2 These were found by parsing the right-hand side β of each production $A \rightarrow \beta$ with the treebank grammar: if a higher likelihood derivation $A \rightarrow^{+} \beta$ can be found then the production is subsumed. As a CL reviewer points out, Krotov et al. (1997) investigate rule redundancy in CFGs estimated from treebanks. They discussed, but did not investigate, rule subsumption in treebank PCFGs.

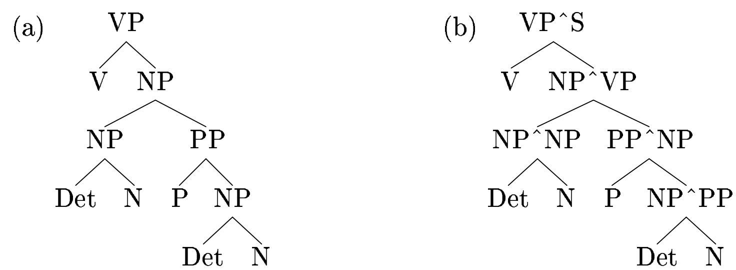


Figure 2
Trees before and after parent annotation.

Note that while the PCFG induced from tree (a) can generate Chomsky adjunction structures because it contains the production NP→NP PP, the PCFG induced from tree (b) can only generate 'two level' NPs.

production tend to be longer than the productions which subsume them, removing them from the grammar reduces the average parse time of the exhaustive PCFG parser used here by more than 9%.

Finally, note that the overgeneration of the PCFG model of the two level adjunction structures is due to an independence assumption implicit in the PCFG model; specifically, that the upper and lower NPs in the two level structure have the same expansions, and that these expansions have the same distributions. This assumption is clearly incorrect for the two level tree representations. If we systematically relabel one of these NPs with a fresh label then a PCFG induced from the resulting transformed tree bank no longer has this property. The 'parent annotation' transform discussed below, which appends the category of a parent node onto the label of all of its nonterminal children as sketched in Figure 2, has just this effect. Charniak and Carroll (1994) describe this transformation as adding "pseudo context-sensitivity" to the language model because the distribution of expansions of a node depends on non-local context, viz., the category of its parent. This non-local information is sufficient to distinguish the upper and lower NPs in the structures considered here.

Indeed, even though the PCFG estimated from the trees obtained by applying the 'parent annotation' transformation to sections 2–21 of the WSJ corpus contains 22,773 productions (i.e., 7,811 more than the PCFG estimated from the untransformed corpus), only 965 of them, or just over 4%, are subsumed by two or more other productions.

4 A theoretical investigation of alternative tree structures

We can gain some theoretical insight into the effect that different tree representations have on PCFG language models by considering several artifical corpora whose estimated PCFGs are simple enough to study analytically. PP attachment was chosen for investigation here because the alternative structures are simple and clear, but presumably the same points could be made for any construction which has several alternative tree representations. Correctly resolving PP attachment ambiguities requires information, such as lexical information (Hindle and Rooth, 1993), which is simply not available to the PCFG models considered here. Still, one might hope that a PCFG model might be able to accurately reflect general statistical trends concerning attachment preferences in the training data, even if it lacks the information to correctly resolve individual cases. But as the analysis in this section makes clear, even this is not always obtained.

For example, suppose our corpora only contain two trees, both of which have yields "V Det N P Det N", are always analysed as a VP with a direct object NP and a PP, and differ only as to whether the PP modifies the NP or the VP. The corpora differ as to how these modifications are represented as trees. The dependencies in these corpora (specifically, the fact that the PP is either attached to the NP or to the VP) violate the independence assumptions implicit in a PCFG model, so one should not expect a PCFG model to exactly reproduce any of these corpora. As a CL reviewer points out, the results presented here depend on the assumption that there is exactly one PP. Never the less, the analysis of these corpora highlights two important points:

- the choice of tree representation can have a noticable effect on the performance

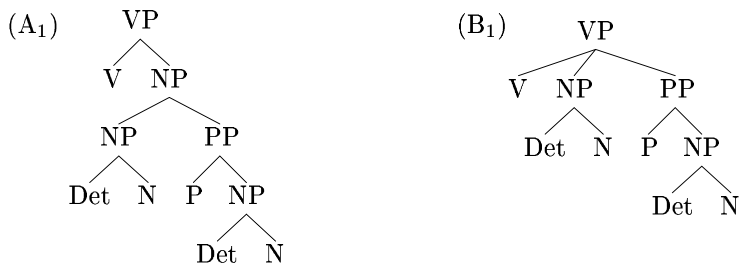


Figure 3
The training corpus $\tilde{\tau}_1$.
This corpus, which uses Penn II tree representations, consists of the trees (A$_1$) with relative frequency f and the trees (B$_1$) with relative frequence $1 - f$. The PCFG $\hat{\mathrm{P}}_1$ is estimated from this corpus.

of a PCFG language model, and

- the accuracy of a PCFG model can depend not just on the trees being modelled, but on their frequency.

4.1 The Penn II representations

Suppose we train a PCFG on a corpus $\tilde{\tau}_1$ consisting only of two different tree structures: the NP attachment structure labelled (A$_1$) and the VP attachment tree labelled (B$_1$) depicted in Figure 3. These trees are called the "Penn II" tree representations here because these are the representations used to encode PP modification in version II of the WSJ corpus constructed at the University of Pennsylvania. Suppose that (A$_1$) occurs in the corpus with relative frequency f and (B$_1$) occurs with relative frequency $(1 - f)$.

In fact, in the WSJ corpus, structure (A$_1$) occurs 7,033 times in sections 2–21 and 279 times in section 22, while structure (B$_1$) occurs 7,717 times in sections 2–21 and and 299 times in section 22. Thus $f \approx 0.48$ in both the F2-21 subcorpora and the F22 corpus.

Returning to the theoretical analysis, the relative frequency counts C_1 and the non-unit production probability estimates $\hat{\mathrm{P}}_1$ for the PCFG induced from this two-tree corpus are as follows:

R	$C_1(R)$	$\hat{\mathrm{P}}_1(R)$
VP $\rightarrow$ V NP	f	f
VP $\rightarrow$ V NP PP	$1-f$	$1-f$
NP $\rightarrow$ Det N	2	$2/(2+f)$
NP $\rightarrow$ NP PP	f	$f/(2+f)$

Of course, in a real tree bank the counts of all these productions would also include their occurences in other constructions, so the theoretical analysis presented here is but a crude idealization. Empirical studies using actual corpus data are presented in section 5.

Thus the estimated likelihoods using $\hat{\mathrm{P}}_1$ of the tree structures (A_1) and (B_1) are:

$$\begin{aligned}\hat{\mathrm{P}}_1(\mathrm{A}_1) &= \frac{4f^2}{(2+f)^3}\\ \hat{\mathrm{P}}_1(\mathrm{B}_1) &= \frac{4\,(1-f)}{(2+f)^2}.\end{aligned}$$

Clearly $\hat{\mathrm{P}}_1(\mathrm{A}_1) < f$ and $\hat{\mathrm{P}}_1(\mathrm{B}_1) < (1-f)$ except at $f=0$ and $f=1$, so in general the estimated frequencies using $\hat{\mathrm{P}}_1$ differ from the frequencies of A_1 and B_1 in the training corpus. This is not too surprising, as the PCFG $\hat{\mathrm{P}}_1$ assigns non-zero probability to trees not in the training corpus (e.g., to trees with more than one PP).

In any case, in the parsing applications mentioned earlier the absolute magnitude of the probability of a tree is of not of direct interest; rather we are concerned with its probability relative to the probabilities of other, alternative tree structures for the same yield. Thus it is arguably more reasonable to ignore the "spurious" tree structures generated by $\hat{\mathrm{P}}_1$ but not present in the training corpus, and compare the estimated relative frequencies of (A_1) and (B_1) under $\hat{\mathrm{P}}_1$ to their frequencies in the training data.

Ideally the estimated relative frequency $\hat{f}_1$ of (A_1)

$$\begin{aligned}\hat{f}_1 &= \hat{\mathrm{P}}_1(\tau = \mathrm{A}_1 : \tau \in \{\mathrm{A}_1, \mathrm{B}_1\})\\ &= \frac{\hat{\mathrm{P}}_1(\mathrm{A}_1)}{\hat{\mathrm{P}}_1(\mathrm{A}_1) + \hat{\mathrm{P}}_1(\mathrm{B}_1)}\\ &= \frac{f^2}{2-f}\end{aligned}$$

will be close to its actual frequency f in the training corpus. The relationship between f and $\hat{f}_1$ is plotted in Figure 4. As inspection of Figure 4 makes clear, the value of $\hat{f}_1$ can

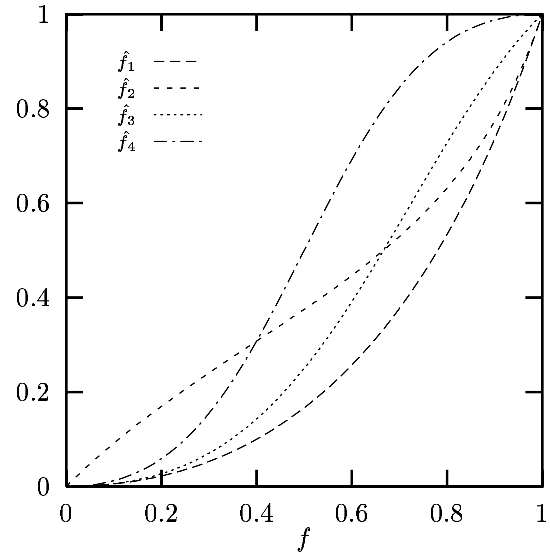


Figure 4
The estimated relative frequency $\hat{f}$ of NP attachment.
This graph shows $\hat{f}$ as a function of the relative frequency f of NP attachment in the training data for various models discussed in the text.

diverge substantially from f. For example, at $f = 0.48$ (the estimate obtained from the WSJ corpus presented above) $\hat{f}_1 = 0.15$. Thus a PCFG language model induced from the simple 2-tree corpus above can underestimate the relative frequency of NP attachment by a factor of more than 3.

4.2 Chomsky adjunction representations

Now suppose that the corpus contains the following two trees (A_2) and (B_2) of Figure 5, which are the Chomsky adjunction representations of NP and VP attached PP's respectively, with relative frequencies f and $(1 - f)$ as before. Note that unlike the Penn II representations, the Chomsky adjunction representation represents NP and VP modification by PPs symmetrically.

The counts C_2 and the non-unit production probability estimates $\hat{\mathrm{P}}_2$ for the PCFG induced from this two-tree corpus are as follows:

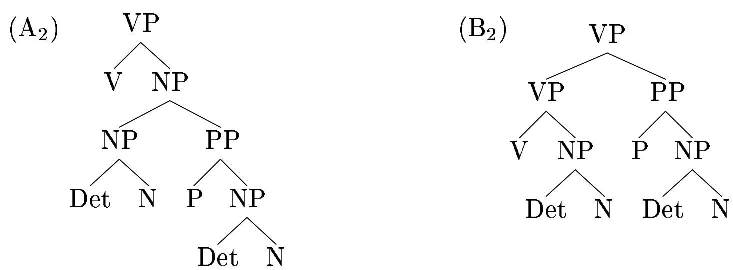


Figure 5
The training corpus $\tilde{\tau}_2$.
This 2-tree corpus, which uses Chomsky adjunction tree representations, consists of the trees (A) with relative frequency f and the trees (B) with relative frequence $1 - f$. The PCFG $\hat{\mathrm{P}}_2$ is estimated from this corpus.

R	$C_2(R)$	$\hat{\mathrm{P}}_2(R)$
VP $\rightarrow$ V NP	1	$1/(2-f)$
VP $\rightarrow$ VP PP	$1-f$	$(1-f)/(2-f)$
NP $\rightarrow$ Det N	2	$2/(2+f)$
NP $\rightarrow$ NP PP	f	$f/(2+f)$

The estimated likelihoods using $\hat{\mathrm{P}}_2$ of the tree structures (A_2) and (B_2) are:

$$\hat{\mathrm{P}}_2(\mathrm{A}_2) = \frac{4f}{(4-f^2)(2+f)^2}$$

$$\hat{\mathrm{P}}_2(\mathrm{B}_2) = \frac{4(1-f)}{(4-f^2)^2}$$

As in the previous subsection, $\hat{\mathrm{P}}_2(\mathrm{A}_2) < f$ and $\hat{\mathrm{P}}_2(\mathrm{B}_2) < (1-f)$ because the PCFG assigns non-zero probability to trees not in the training corpus. Again, we calculate the estimated relative frequencies of (A_2) and (B_2) under $\hat{\mathrm{P}}_2$.

$$\begin{aligned} \hat{f}_2 &= \hat{\mathrm{P}}_2(\tau = \mathrm{A}_2 : \tau \in \{\mathrm{A}_2, \mathrm{B}_2\}) \\ &= \frac{f^2 - 2f}{2f^2 - f - 2} \end{aligned}$$

The relationship between f and $\hat{f}_2$ is also plotted in Figure 4. The value of $\hat{f}_2$ can diverge from f, although not as widely as $\hat{f}_1$. For example, at $f = 0.48$ $\hat{f}_2 = 0.36$. Thus the precise tree structure representations used to train a PCFG can have a marked effect on the probability distribution that it generates.

4.3 Flattened tree representations

The previous subsection showed that inserting additional nodes into the tree structure can result in a PCFG language model which better models the distribution of trees in the training corpus. This subsection investigates the effect of removing the lower NP node in the WSJ NP modification structure, again resulting in a pair of more symmetric tree structures, as shown in Figure 6. As explained in section 1, flattening the tree structures in general corresponds to weakening the independence assumptions in the induced PCFG models, so one might expect this to improve the induced language model.

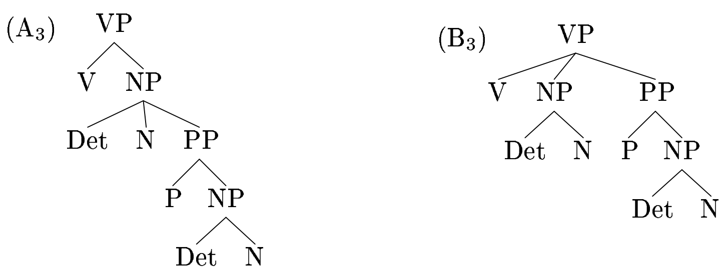


Figure 6
The training corpus $\tilde{\tau}_3$.
The NP modification tree representation used in the Penn II WSJ corpus is ‘flattened’ to make it similiar to the VP modification representation. The PCFG $\hat{P}_3$ is estimated from this corpus.

The counts C_3 and the non-unit production probability estimates $\hat{\mathrm{P}}_3$ for the PCFG induced from this two-tree corpus are as follows:

R	$C_3(R)$	$\hat{\mathrm{P}}_3(R)$
VP $\rightarrow$ V NP	f	f
VP $\rightarrow$ VP NP PP	$1-f$	$1-f$
NP $\rightarrow$ Det N	$2-f$	$1-f/2$
NP $\rightarrow$ Det N PP	f	$f/2$

The estimated likelihoods using $\hat{\mathrm{P}}_3$ of the tree structures (A_3) and (B_3) are:

$$\hat{\mathrm{P}}_3(\mathrm{A}_3) = \frac{f^2}{2\,(1-f/2)}$$

$$\hat{\mathrm{P}}_3(\mathrm{B}_3) = (1-f)\,(1-f/2)^2$$

As before $\hat{\mathrm{P}}_3(\mathrm{A}_3) < f$ and $\hat{\mathrm{P}}_3(\mathrm{B}_3) < (1-f)$, again because the PCFG assigns non-zero probability to trees not in the training corpus. The estimated relative frequency $\hat{f}_3$ of (A_3) relative to (B_3) under $\hat{\mathrm{P}}_3$ is:

$$\hat{f}_3 = \hat{\mathrm{P}}_3(\tau = \mathrm{A}_3 : \tau \in \{\mathrm{A}_3, \mathrm{B}_3\}) = \frac{f^2}{2-3f+2f^2}$$

The relationship between f and $\hat{f}_3$ is also plotted in Figure 4. The value of $\hat{f}_3$ diverges from f, as before: at $f = 0.48$ $\hat{f}_3 = 0.23$. As Figure 4 shows, the estimated relative frequency $\hat{f}_3$ using the flattened tree representations is always closer to f than the estimated relative frequency $\hat{f}_1$ using the Penn II representations, but is only closer to f than the estimated relative frequency $\hat{f}_2$ using the Chomsky adjunction representations for f greater than approximately 0.7.

4.4 Penn II representations with parent annotation

As mentioned in section 1, another way of relaxing the independence assumptions implicit in a PCFG model is to systematically encode more information in node labels about their context. This subsection explores a particularly simple kind of contextual encoding: the

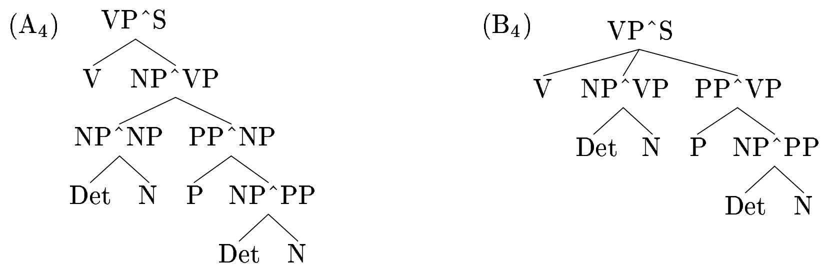


Figure 7
The training corpus $\tilde{\tau}_4$.
This corpus, which uses Penn II tree bank tree representations in which each preterminal node's parent's category is appended onto its own label, consists of the trees (A_4) with relative frequency f and the trees (B_4) with relative frequence $1 - f$. The PCFG $\hat{P}_4$ is estimated from this corpus.

label of the parent of each non-root nonpreterminal node is appended to that node's label. The labels of the root node and the terminal and preterminal nodes are left unchanged.

For example, assuming that the Penn II format trees (A_1) and (B_1) of subsection 4.1 are immediately dominated by a node labelled S, this relabelling applied to those trees produces the trees (A_4) and (B_4) depicted in Figure 7.

We can perform the same theoretical analysis on this two tree corpus that we applied to the previous corpora to investigate the effect of this relabelling on the PCFG modelling of PP attachment structures.

The counts C_4 and the non-unit production probability estimates $\hat{P}_4$ for the PCFG induced from this two-tree corpus are as follows:

R	$C_4(R)$	$\hat{P}_4(R)$
VP^S → V NP^VP	f	f
VP^S → V NP^VP PP^VP	$1 - f$	$1 - f$
NP^VP → Det N	$1 - f$	$1 - f$
NP^VP → NP^NP PP^NP	f	f

The estimated likelihoods using $\hat{P}_4$ of the tree structures (A_4) and (B_4) are:

$$\hat{P}_4(A_4) = f^2$$

$$\hat{P}_4(B_4) = (1 - f)^2$$

As in the previous subsection $\hat{P}_4(A_4) < f$ and $\hat{P}_4(B_4) < (1 - f)$. Again, we calculate the estimated relative frequencies of (A_4) and (B_4) under $\hat{P}_4$.

$$\begin{aligned}\hat{f}_4 &= \hat{P}_4(\tau = A_4 : \tau \in \{A_4, B_4\}) \\ &= \frac{f^2}{f^2 + (1-f)^2}\end{aligned}$$

The relationship between f and $\hat{f}_4$ is plotted in Figure 4. The value of $\hat{f}_4$ can diverge from f, just like the other estimates. For example, at $f = 0.48$ $\hat{f}_4 = 0.46$, which is closer to f than any of the other relative frequency estimates presented earlier. (However, for f less than approximately 0.38, the relative frequency estimate using the Chomsky adjunction representations $\hat{f}_2$ is closer to f than $\hat{f}_4$). Thus as expected, increasing the context information in the form of an enriched node labelling scheme can improve the performance of a PCFG language model.

5 Empirical investigation of different tree representations

The previous section presented theoretical evidence that varying the tree representations used to estimate a PCFG language model can have a noticable impact on that model's performance. However, as anyone working with statistical language models knows, the actual performance of a language model on real language data can often differ dramatically from one's expectations, even when it has an apparently impeccable theoretical basis. For example, on the basis of the theoretical models presented in the last section (and, undoubtedly, a background in theoretical linguistics) I expected that PCFG models induced from Chomksy adjunction tree representations would perform better than models induced from the Penn II representations. However, as shown in this section, this is not the case, but some of the other tree representations investigated here induce PCFGs which do perform noticably better than the Penn II representations.

It is fairly straight forward to mechanically transform the Penn II tree representations in the WSJ corpus into something close to the alternative tree representations described above, although the diversity of local trees in the WSJ corpus makes this task more difficult. For example, what is the Chomsky adjunction representation of a VP with no apparent verbal head? In addition, the Chomsky adjunction representation requires argument PPs to be attached as sisters of the lexical head, while adjunct PPs are attached as sisters of a non-lexical projection. Argument PPs are not systematically distinguished from adjunct PPs in the Penn II tree representations, and reliably determining whether a particular PP is an argument or an adjunct is extremely difficult, even for trained linguists. Never the less, the tree transformations investigated below should give at least an initial idea as to the influence of different kinds of tree representation on the induced PCFG language models.

5.1 The tree transformations

The tree transformations investigated in this section are listed below. Each is given a short name which is used to identify it in the rest of the paper. Designing the tree transformations is complicated by the fact that there are in general many different tree transformations which correctly transform the simple cases discussed in section 4, but behave differently on more complex constructions that appear in the WSJ corpus. The actual transformations investigated here have the advantage of simplicity, but many other different transformations would correctly transform the trees discussed in sections 3 and 4 and be just as linguistically plausible as the transforms below, yet would presumably induce PCFGs with very different properties.

Id is an identity transformation, i.e., it does not modify the trees at all. This condition studies the behaviour of the Penn II tree representation used in the WSJ corpus.

NP-VP produces trees which represent PP modification of both NPs and VPs using Chomsky adjunction. The NP-VP transform is the result of exhaustively applying all four of the tree transforms depicted in Figure 8. The first and fourth trans-

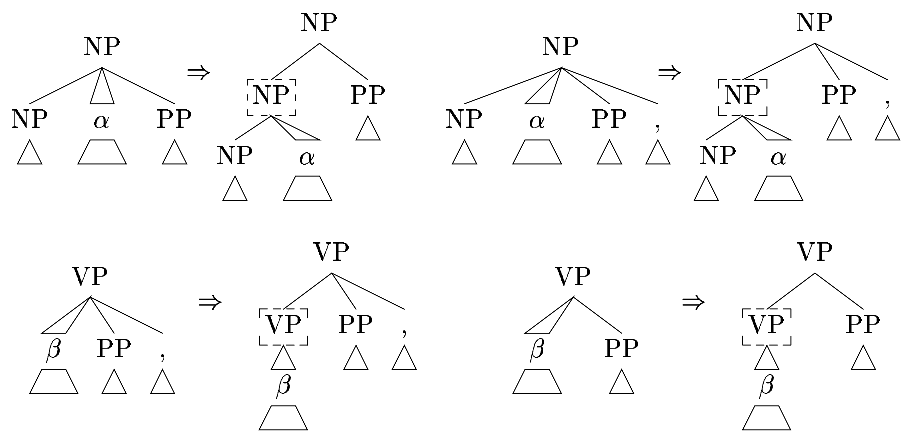
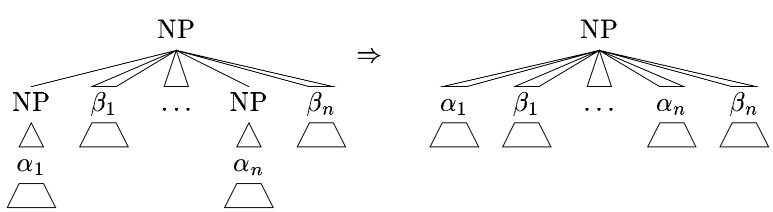


Figure 8
Producing Chomsky adjunction tree representations.
The four tree transforms depicted here are exhaustively reapplied to produce the Chomsky adjunction tree representations from Penn II tree representations in the **NP-VP** transformation. In the **N′-V′** transformation the boxed NP and VP nodes are relabelled with N′ and V′ respectively. In these schema α is a sequence of trees of length 1 or greater and β is a sequence of trees of length 2 or greater.

forms transform NP and VP nodes whose rightmost child is a PP into Chomsky adjunction structures, and the second and third transforms adjoin final PPs with a following comma punctuation into Chomsky adjunction structures. The constraints that $\alpha > 1$ and $\beta > 2$ ensures that these transforms will only apply a finite number of time to any given subtree.

N′-V′ produces trees which represent PP modification of NPs and VPs with a Chomsky adjunction representation that uses an 'intermediate' level of X′ structure. This is the result of repeatedly applying the four transformations depicted in Figure 8 as in the NP-VP transform, with the modification that the new non-maximal nodes are labelled N′ or V′ as appropriate (rather than NP or VP).

Flatten produces trees in which NPs have a flatter structure than the 'two level' representation of NPs used in the Penn II tree bank. Only subtrees consisting of a parent node labelled NP whose first child is also labelled NP are affected by this transformation. The effect of this transformation is to excise all the children nodes labelled NP from the tree, and to attach their children as direct descendants of the parent node, as depicted in the schema below.



Parent appends to each nonroot nonterminal node's label its parent's category. The effect of this transformation is to produce trees of the kind discussed in subsection 4.4.

5.2 Evaluation of parse trees

It is straightforward to estimate PCFGs using the relative frequency estimator from the sequences of trees produced by applying these transforms to the WSJ corpus. We turn now to the question of evaluating the different PCFGs so obtained.

None of the PCFGs induced the various tree representations discussed here reliably identifies the correct tree representations on sentences from held-out data. It is standard to evaluate broad coverage parsers using less stringent criteria which measure how similiar the trees produced by the parser are to the 'correct' analysis trees in a portion of the tree bank held out for testing purposes. This study uses the 1,578 sentences section 22 of the WSJ corpus of length 40 or less for this purpose.

The labelled *precision* and *recall* figures are obtained by regarding the sequence of trees $\tilde{\tau}$ produced by a parser as a multiset or bag $E(\tilde{\tau})$ of edges, i.e., triples $\langle N, l, r \rangle$ where N is a nonterminal label and l and r are left and right string positions in yield of the entire corpus. (Root nodes and preterminal nodes are not included in these edge sets, as they are given as input to the parser). Relative to a 'test sequence' of trees $\tilde{\tau}'$ (here section 22 of the WSJ corpus) the labelled precision and recall of a sequence of trees $\tilde{\tau}$ with the same yield as $\tilde{\tau}'$ are calculated as follows, where the '$\cap$' operation denotes multiset intersection.

$$\begin{aligned} \text{Precision}(\tilde{\tau}) &= \frac{|E(\tilde{\tau}) \cap E(\tilde{\tau}')|}{|E(\tilde{\tau})|} \\ \text{Recall}(\tilde{\tau}) &= \frac{|E(\tilde{\tau}) \cap E(\tilde{\tau}')|}{|E(\tilde{\tau}')|} \end{aligned}$$

Thus precision is the fraction of edges in the tree sequence to be evaluated which also appear in the test tree sequence, and recall is the fraction of edges in the test tree sequence which also appear in tree sequence to be evaluated.

It is straight forward to use the PCFG estimation techniques described in section 2 to estimate PCFGs from the result of applying these transformations to sections 2–21 of the Penn II WSJ corpus. The resulting PCFGs can be used with a parser to obtain maximum likelihood parse trees for the POS tag yields of the trees of the held out test corpus (section 22 of the WSJ corpus). While the resulting parse trees can be compared to the trees in the test corpus using the precision and recall measures described above, the results would not be meaningful as the parse trees reflect a different tree representation to that used in the test corpus, and thus are not directly comparable with the test corpus trees. For example, the node labels used in the PCFG induced from trees produced by applying the 'parent' transform are pairs of categories from the original Penn II WSJ tree bank, and so the *labelled* precision and recall measures obtained by comparing the parse trees obtained using this PCFG with the trees from the tree bank would be close to zero.

One might try to overcome this by applying the same transformation to the test trees as was used to obtain the training trees for the PCFG, but then the resulting precision and recall measures would not be comparable across transformations. For example, as two different Penn II format trees may map to the same 'flattened' tree, the 'flatten' transformation is in general not invertible. Thus a parsing system which produces perfect flat tree representations provides less information than one which produces perfect Penn II tree representations, and one might expect that all else being equal, a parsing system using flat representations will score higher (or at least differently) in terms of precision and recall than an equivalent one producing Penn II representations.

The approach developed here overcomes this problem by applying an additional tree transformation step which converts the parse trees produced using the PCFG back to the Penn II tree representations, and compares these trees to the held-out test trees using the labelled precision and recall trees. This transformation-detransformation process is depicted in Figure 9. It has the virtue that all precision and recall measures involve trees

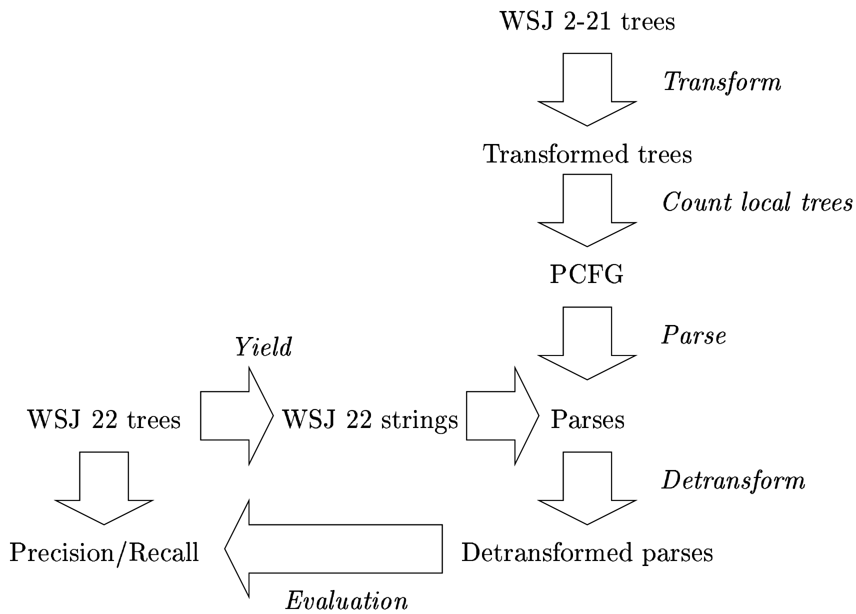


Figure 9
The tree transformation/detransformation process.

using the Penn II tree representations, but it does involve an additional detransformation step.

It is straight forward to define detransformers for all of the tree transformations described in this section except for the 'flattening' transform. The difficulty in this case is that several different Penn II format trees may map onto the same 'flattened' tree, as mentioned above. The detransformer for the 'flattening' transform was obtained by recording for each distinct local tree in the flattened tree representation of the training corpus the various tree fragments in the Penn II format training corpus it could have been derived from. The detransformation of a 'flattened' tree is effected by replacing each local tree in the parse tree with its most frequently occuring Penn II format fragment.

This detransformation step is in principle an additional additional source of error, in that a parser could produce flawless parse trees in its particular tree representation, but the transformation to the corresponding Penn II tree representations might itself introduce errors. For example, it might be that several different Penn II tree represen-

Table 1
The results of an empirical study of the effect of tree structure on PCFG models.
Each column corresponds to a sequence of trees, either consisting of section 22 of the WSJ corpus or transforms of the maximum likelihood parses of the yields of the section 22 subcorpus with respect to different PCFGs, as explained in the text. The first row reports the number of productions in these PCFGs, and the next two rows give the labelled precision and recall of these sequences of trees. The last four rows report the number of times particular kinds of subtrees appear in these sequences of trees, as explained in the text.

	22	**22 Id**	**Id**	**NP-VP**	**N′-V′**	**Flatten**	**Parent**
No. of rules		2,269	14,962	14,297	14,697	22,652	22,773
Precision	1	0.772	0.735	0.730	0.735	0.745	0.800
Recall	1	0.728	0.697	0.705	0.701	0.723	0.792
NP attachments	279	0	67	330	69	154	217
VP attachments	299	424	384	0	503	392	351
NP* attachments	339	3	67	399	69	161	223
VP* attachments	412	668	662	150	643	509	462

tations can correspond to a single parse tree, as is the case with a parser producing ‘flattened’ tree representations. To determine if detransformation can be done reliably, for each tree transformation labelled precision and recall measures were calculated comparing the result of applying the transformation and the corresponding detransformation to the test corpus trees with the original trees of the test corpus. In all cases except for the ‘flattening’ transform these precision and recall measures were always greater than 99.5%, indicating that the transformation-detransformation process is quite reliable. For the ‘flattening’ transform the measures were greater than 97.5%, suggesting that while the error introduced by this process is noticable, the transformation-detransformation process does not introduce a very large error on its own.

5.3 Results

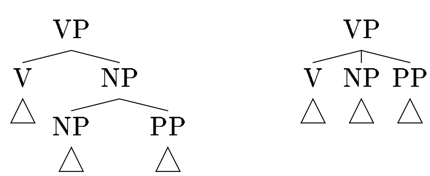
Table 1 presents an analysis of the sequences of trees produced via this detransformation process applied to the maximum likelihood parse trees. The columns of this table correspond to sequences of parse trees for section 22 of the WSJ corpus. The column labelled “22” describes the trees given in section 22 of the WSJ corpus, and the column labelled

"22 Id" describes the maximum likelihood parse trees of section 22 of the WSJ corpus using the PCFG induced from those very trees. This is thus an example of 'training on the test data', and is often assumed to provide an upper bound on the performance of a learning algorithm. The remaining columns describe the sequences of trees produced using the transformation-detransformation process described above.

The first three rows of that table show the number of productions in each PCFG (which is the number of distinct local trees in the corresponding transformed training corpus), and the labelled precision and recall measures for the detransformed parse trees.

Randomization tests for paired sample data were performed to assess the significance of the difference between the labelled precision and recall scores for the output of the 'Id' PCFG and the other PCFGs (Cohen, 1995). The labelled precision and recall scores for the 'Flatten' and 'Parent' transforms differed significantly from each other and also from the 'Id' transform at the 0.01 level, while neither the 'NP-VP' nor the 'N′-V′' transform differed significantly from each other or the 'Id' transform at the 0.1 level.

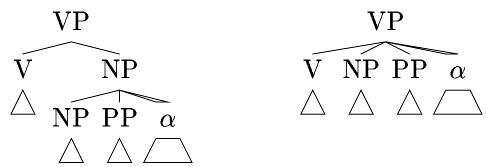
The remaining rows of Table 1 show the number of times certain tree schema appear in these (detransformed) tree sequences. The rows labelled NP attachments and VP attachments provide the number of times the following tree schema, which represent a single PP attachment, match the tree sequence.[3] In these schema, V can be instantiated by any of the verbal preterminal tags used in the Penn II corpus.



The rows labelled NP* attachments and VP* attachments provide the number of

3 The Penn II markup scheme permits a 'pseudo-attachment' notation for indicating ambiguous attachment. However, this is only used relatively infrequently—the pseudo-attachment markup only appears 27 times in the entire Penn II tree bank—and was ignored here. Pseudo-attachment structures count as VP attachment structures here.

times that the following more relaxed schema match the tree sequence. Here α can be instantiated by any sequence of trees, and V can be instantiated by the same range of preterminal tags as above.



5.4 Discussion

As expected, the PCFGs induced from the output of the 'Flatten' transform and 'Parent' transform significantly improve precision and recall over the original tree bank PCFG (i.e., the PCFG induced from the output of the 'Id' transform). The PCFG induced from the output of the 'Parent' transform performed significantly better than any other PCFG investigated here. As discussed above, both the 'Parent' and the 'Flatten' transforms induce PCFGs that are sensitive to what would be non-CF dependencies in the original tree bank trees, which perhaps accounts for their superior performance. Both the 'Flatten' and 'Parent' transforms induced PCFGs which have substantially more productions than the original tree bank grammar, perhaps reflecting the fact that they encode more contextual information than the original tree bank grammar, albeit in different ways. Their superior performance suggests that the reduction in bias obtained by the weakening of independence assumptions that these transformations induce more than outweighs any associated increase in variance.

The various adjunction transformations only had minimal effect on labelled precision and recall. Perhaps this is because PP attachment ambiguities, despite their important role in linguistic and parsing theory, are just one source of ambiguity among many in real language, and the effect of the alternative representations has only minor effect.

Indeed, moving to the purportedly linguistically more realistic tree Chomsky adjunction representations did not improve performance on these measures. On reflection, perhaps this should not be surprising. The Chomsky adjunction representations are motivated within the theoretical framework of Transformational Grammar, which explicitly argues for nonlocal, indeed, non context free, dependencies. Thus its poor performance when used as input to a statistical model which is insensitive to such dependencies is perhaps to be expected. Indeed, it might be the case that inserting the additional adjunction nodes inserted by the 'NP-VP' and 'N′-V′′' transformations above have the effect of converting a local dependency (which can be described by a PCFG) into a nonlocal dependency (which cannot).

Another initially surprising property of the tree sequences produced by the PCFGs is that they do not reflect at all well the frequency of the different kinds of PP attachment found in the Penn II corpus. This is in fact to be expected, since the sequences consist of *maximum likelihood* parses. To see this, consider any of the examples analysed in section 4. In all of these cases, the corpora contained two tree structures, and the induced PCFG associates each with an estimated likelihood. If these likelihoods differ, then a maximum likelihood parser will always return the same maximum likelihood tree structure each time it is presented with its yield, and will never return the tree structure with lower likelihood, even though the PCFG assigns it a nonzero likelihood.

Thus the surprising fact is that these PCFG parsers ever produce a nonzero number of NP attachments and VP attachments in the same tree sequence. This is possible because the node label V in the attachment schema above abbreviates several different preterminal labels (i.e., the set of all verbal tags). Further investigation shows that once the V label in NP and VP attachment schemas is instantiated with a particular verbal tag, only either the relevant NP attachment schema or the VP attachment schema appears in the tree sequence. For instance, in the Id tree sequence (i.e., produced by the standard tree bank grammar) the 67 NP attachments all occured with the V label instantiated to the verbal tag AUX.[4]

It is worth noting that the 8% improvement in average precision and recall obtained by the parent annotation transform is approximately half of the performance difference between a parser using a PCFG induced directly from the tree bank (i.e., using the 'Id' transform above) and best currently available broad coverage parsing systems, which exploit lexical as well purely syntactic information (Charniak, 1997).

In order to better understand just why the parent annotation transform performs so much better than the other transforms, transformation-detransformation experiments were performed in which the parent annotation transform was performed selectively either on all nodes with a given category label, or all nodes with a given category label and parent category label. Figure 10 depicts the effect of selective application of the parent annotation transform on the change of the average of precision and recall with respect to the 'Id' transform. It is clear that distinguishing the context of NP and S nodes is responsible for an important part of the improvement in performance. Merely distinguishing root from non-root S nodes—a distinction made in early transformational grammar but ignored in more recent work—improves average precision and recall by approximately 3%. Thus it is possible that the performance gains achieved by the parent annotation transform have little to do with PP attachment.

6 Conclusion

This paper has presented theoretical and empirical evidence that the choice of tree representation can make a significant difference to the performance of a PCFG based parsing system. What makes a tree representation a 'good choice' for PCFG modelling seems to be quite different to what makes it a good choice for a representation of a linguistic

4 This tag was introduced by (Charniak, 1996) to distinguish auxiliary verbs from main verbs.

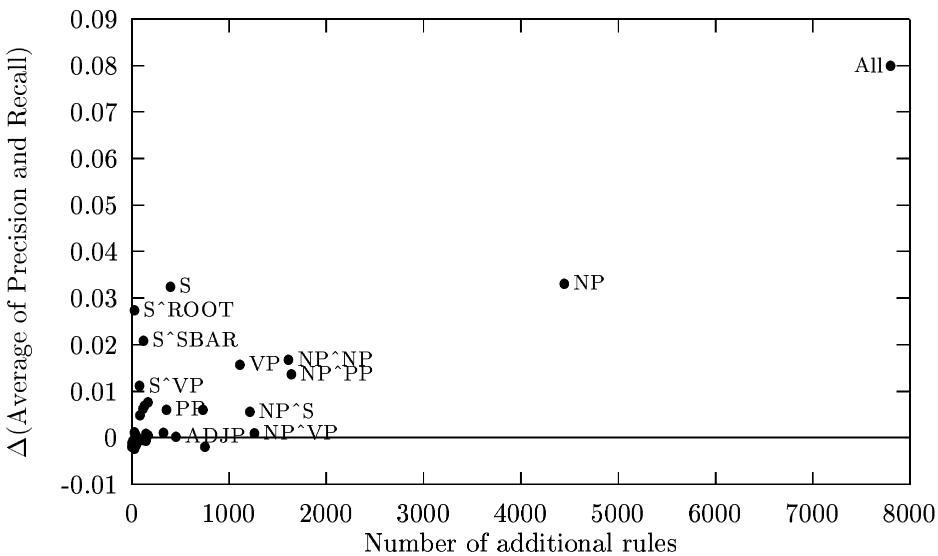


Figure 10
The effects of selective application of the 'Parent' transform.
Each point corresponds to a PCFG induced after selective application of the 'Parent' transform. The point labelled 'All' corresponds to the PCFG induced after the 'Parent' transform to all nonroot nonterminal nodes, as before. Points labelled with a single category A correspond to PCFGs induced after applying the 'parent' transform to just those nodes labelled A, while points labelled with a pair of categories $A\hat{\ }B$ correspond to PCFGs induced applying the 'Parent' transform to nodes labelled A with parents labelled B. (Some labels are elided to make the remaining labels legible). The x axis shows the difference in number of productions in the PCFG after selective parent transform and the untransformed treebank PCFG, and the y axis shows the difference in the average of the precision and recall scores.

theory. In conventional linguistic theories the choice of rules, and hence trees, is usually influenced by considerations of parsimony; thus the Chomsky adjunction representation of PP modification may be preferred because it requires only a single context free rule, rather than a rule schema abbreviating a potentially unbounded number of rules that would be required in 'flat' tree representations of adjunction. But in a PCFG model the additional nodes required by the Chomsky adjunction representation represent independence assumptions which seem not to be justified. In general, in selecting a tree structure one faces a bias/variance trade-off, in that tree structures with fewer nodes and/or richer node labels reduce bias, but possibly at the expense of an increase in variance. A tree transformation-detransformation methodology for empirically evaluating the effect of different tree representations on parsing systems was developed in this paper. The results presented earlier show that the tree representations investigated here which incorporated weaker independence assumptions performed signficantly better in the empirical studies than the more linguistically motivated Chomsky adjunction structures.

Of course, there is nothing particularly special about the particular tree transformations studied in this paper: other transforms could—and should—be studied in exactly the same manner. For example, I am currently using this methodology to study the interaction between tree structure and a "slash category" node labelling in tree representations with empty categories (Gazdar et al., 1985). While the work presented here focussed on PCFG parsing models, it seems that the general transformation-detransformation approach can be applied to a wider range of problems. For example, it would be interesting to know to what extent the performance of more sophisticated parsing systems, such as those described by Collins (1996) and Charniak (1997), depends on the particular tree representations they are trained on.

Acknowledgments
I would like to thank Eugene Charniak and my other colleagues at Brown, as well as the CL reviewers for their excellent advice in this research. This material is based on work supported by the National Science Foundation under Grant No. SBR-9720368.

References

Aho, Alfred V. and Jeffery D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling; Volume 1: Parsing.* Prentice-Hall, Englewood Cliffs, New Jersey.

Bies, Ann, Mark Ferguson, Karen Katz, and Robert MacIntyre, 1995. *Bracketting Guideliness for Treebank II style Penn Treebank Project.* Linguistic Data Consortium.

Charniak, Eugene. 1993. *Statistical Language Learning.* The MIT Press, Cambridge, Massachusetts.

Charniak, Eugene. 1996. Tree-bank grammars. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 1031–1036, Menlo Park. AAAI Press/MIT Press.

Charniak, Eugene. 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.

Charniak, Eugene and Glenn Carroll. 1994. Context-sensitive statistics for improved grammatical language models. In *Proceedings of AAAI '94*, pages 728–733.

Chi, Zhiyi and Stuart Geman. to appear. Estimation of probabilistic context-free grammars. *Computational Linguistics.*

Cohen, Paul R. 1995. *Empirical Methods for Artificial Intelligence.* The MIT Press, Cambridge, Massachusetts.

Collins, Michael John. 1996. A new statistical parser based on bigram lexical dependencies. In *The Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 184–191, San Francisco. The Association for Computational Linguistics, Morgan Kaufmann.

Culy, Christopher. 1985. The complexity of the vocabulary of Bambara. *Linguistics and Philosophy*, 8(3):345–352.

Gazdar, Gerald, Ewan Klein, Geoffrey Pullum, and Ivan Sag. 1985. *Generalized Phrase Structure Grammar.* Basil Blackwell, Oxford.

Geman, Stuart, Elie Bienenstock, and René Doursat. 1992. Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58.

Hindle, Donald and Mats Rooth. 1993. Structural ambiguity and lexical relations. *Computational Linguistics*, 19(1):103–120.

Krotov, Alexander, Robert Gaizauskas, Mark Hepple, and Yorik Wilks. 1997. Compacting the Penn Treebank grammar. Technical report, Department of Computer Science, Sheffield University. Available at `http://www.dcs.shef.ac.uk/~alexk/auto_papers.html`.

Marcus, Michell P., Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Shieber, Stuart M. 1985. Evidence against the Context-Freeness of natural language. *Linguistics and Philosophy*, 8(3):333–344.