

Probabilistic Context-Free Grammars

Based on

“Foundations of Statistical NLP” by C. Manning & H. Schütze, ch. 11
MIT Press, 2002

A Sample PCFG

$S \rightarrow NP VP$	1.0	$NP \rightarrow NP PP$	0.4
$PP \rightarrow P NP$	1.0	$NP \rightarrow \text{astronomers}$	0.1
$VP \rightarrow V NP$	0.7	$NP \rightarrow \text{ears}$	0.18
$VP \rightarrow VP PP$	0.3	$NP \rightarrow \text{saw}$	0.04
$P \rightarrow \text{with}$	1.0	$NP \rightarrow \text{stars}$	0.18
$V \rightarrow \text{saw}$	1.0	$NP \rightarrow \text{telescopes}$	0.1

The Chomsky Normal Form of CFGs

CNF CFG: All non-terminals expand into either two or more non-terminals ($N \rightarrow X Y$) or a single terminal ($N \rightarrow w$).

Proposition: Any CFG can be converted into a “weakly equivalent” CNF CFG.

Definition: Two grammars are **weakly equivalent** if they generate the same language. They are **strongly equivalent** if they also assign the same structures to strings.

Example: CYK with Chart Representation

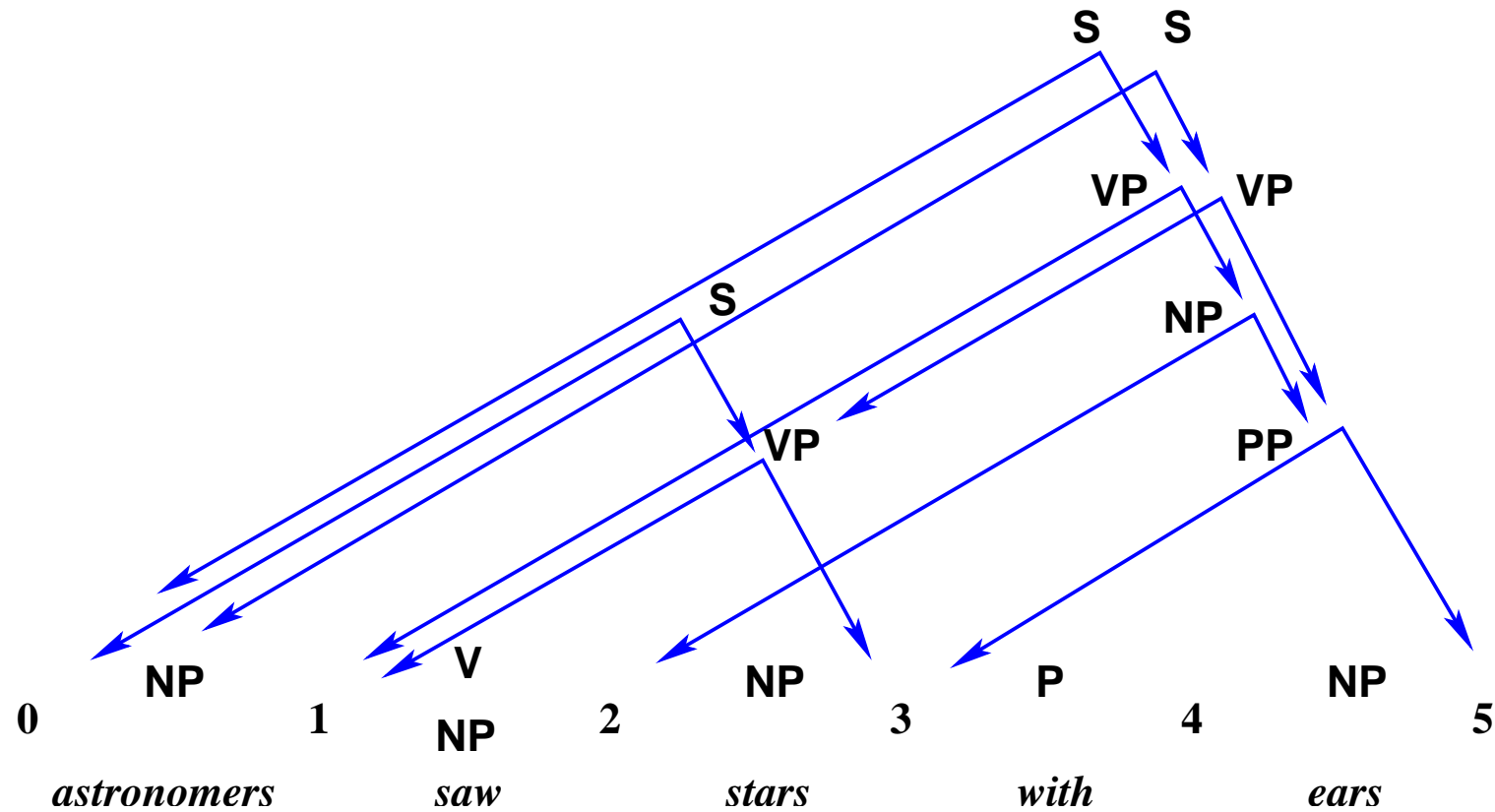
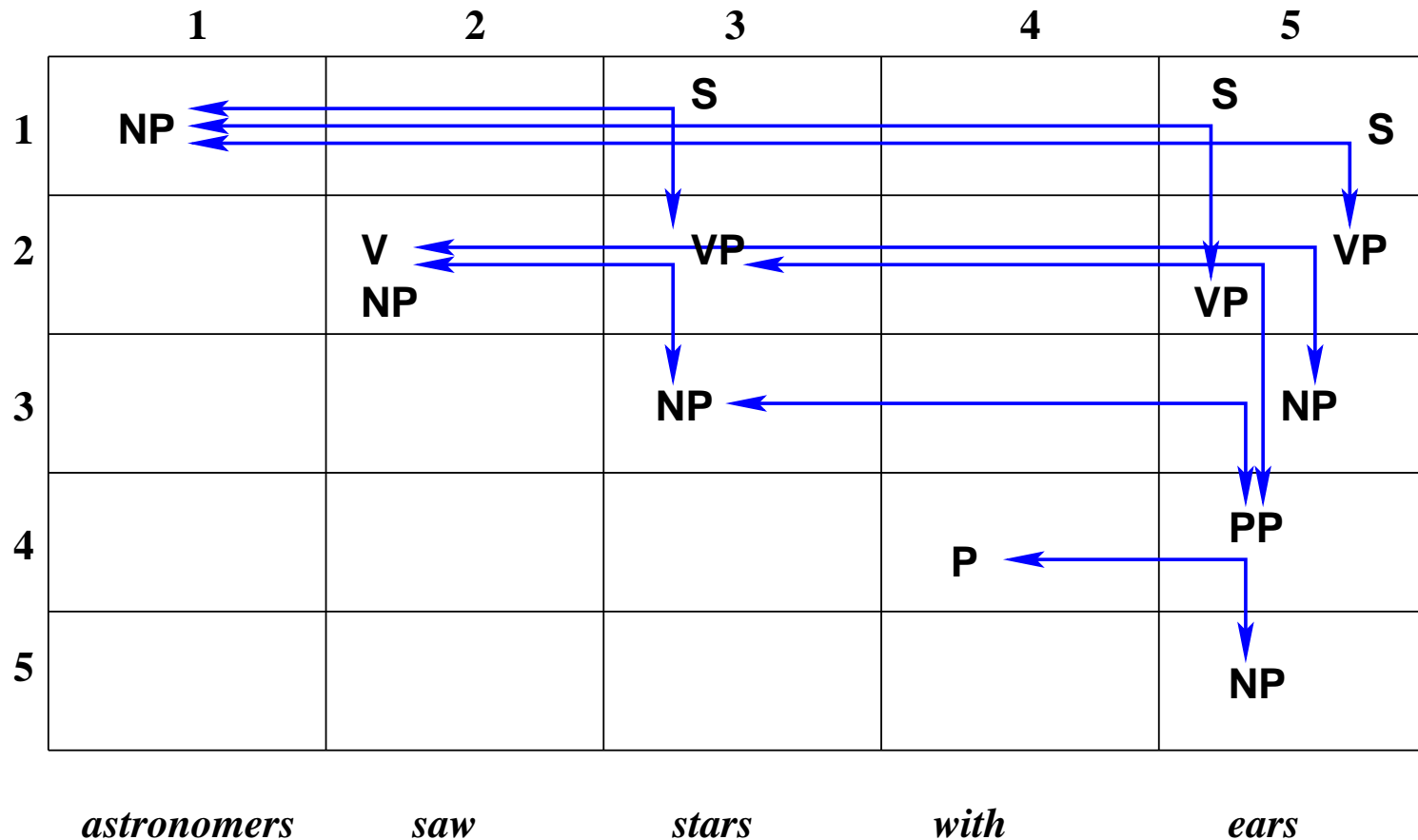


Chart Representation as a Matrix



Assumptions of the PCFG Model

- $\forall i \sum_j P(N^i \rightarrow \nu^j \mid N^i) = 1$
- **Place invariance:**
the probability of a subtree does not depend on where in the string the words it dominates are
- **Context-free:**
the probability of a subtree does not depend on words not dominated by the subtree
- **Ancestor-free:**
the probability of a subtree does not depend on nodes outside of the subtree

Calculating the Probability of a Sentence

So, the probability of a sentence is

$$P(w_{1m}) = \sum_t P(w_{1m}, t) = \sum_{t: \text{yield}(t)=w_{1m}} P(t)$$

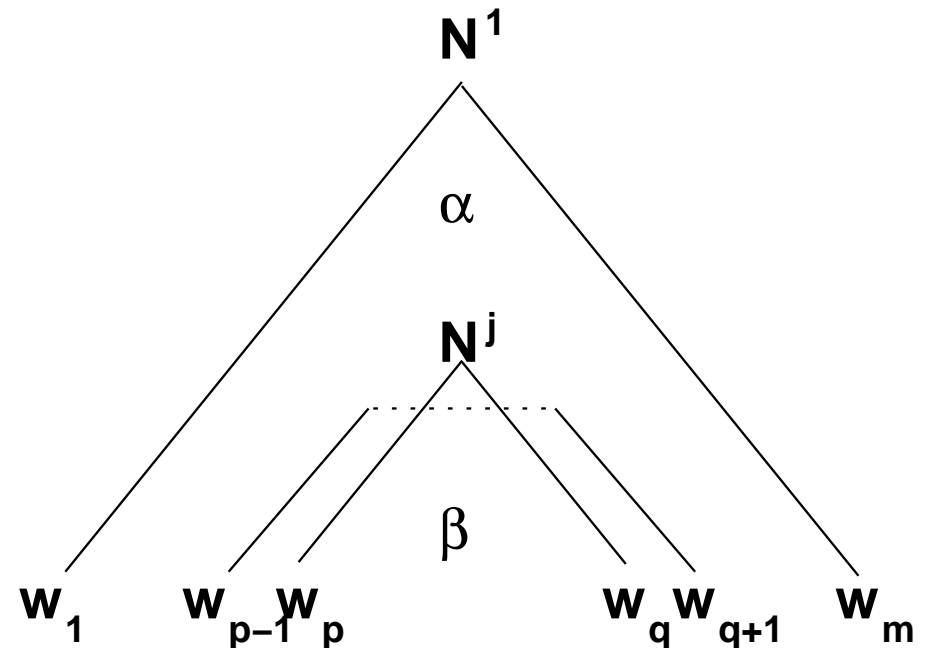
where t is a parse tree of the sentence.

To calculate the probability of a tree,
multiply the probabilities of all the rules it uses.

Inside and Outside Probabilities

Outside (α): the total probability of beginning in N^1 and generating N^j and the words outside p and q

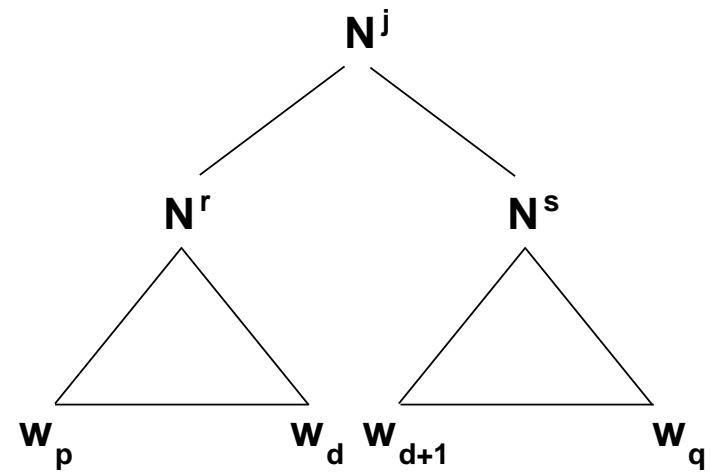
Inside (β): the total probability of generating the words from p to q given that we start at non-terminal N^j



$$\alpha^j(p, q) = P(w_{1(p-1)}, N_{pq}^j, w_{(q+1),m}) \quad \beta^j(p, q) = P(w_{pq} | N_{pq}^j)$$

Computing Inside Probabilities

Base case: $\beta^j(k, k) = P(N^j \rightarrow w^k | N^j)$



Induction step: $\beta^j(p, q) = P(w_{pq} | N_{pq}^j) =$

$$\sum_{r,s} \sum_{d=p}^{q-1} P_G(N^j \rightarrow N^r N^s | N^j) \beta^r(p, d) \beta^s(d+1, q)$$

Computing Inside Probabilities — Induction

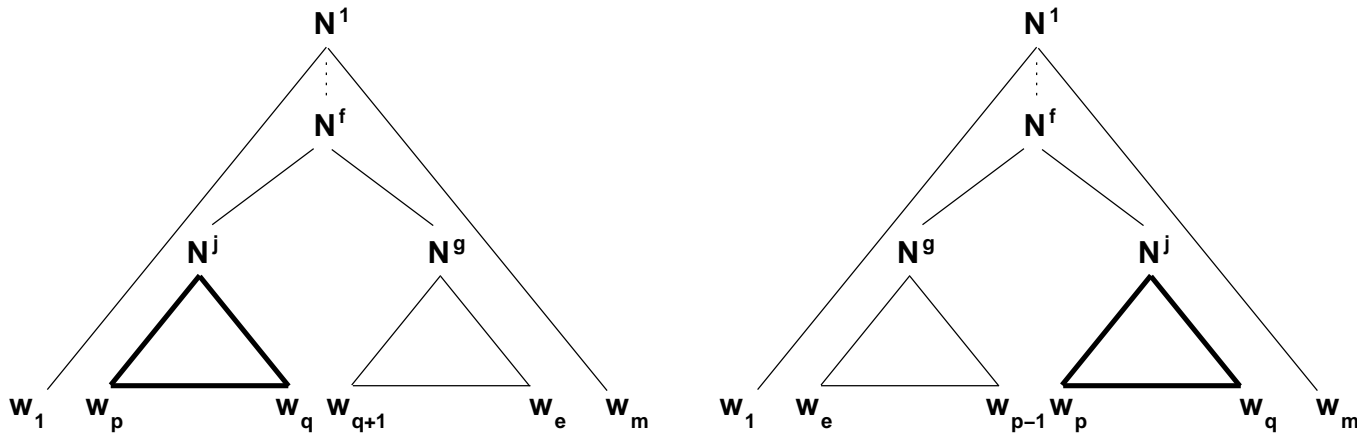
$$\begin{aligned}
 \beta^j(p, q) &= P(w_{pq} | N_{pq}^j) = \sum_{r,s} \sum_{d=p}^{q-1} P(w_{pd}, N_{pd}^r, w_{(d+1)q}, N_{(d+1)q}^s | N_{pq}^j) \\
 &= \sum_{r,s} \sum_{d=p}^{q-1} P(N_{pd}^r, N_{(d+1)q}^s | N_{pq}^j) P(w_{pd} | N_{pq}^j, N_{pd}^r, N_{(d+1)q}^s) \\
 &\quad \times P(w_{(d+1)q} | N_{pq}^j, N_{pd}^r, N_{(d+1)q}^s, w_{pd}) \\
 &= \sum_{r,s} \sum_{d=p}^{q-1} P(N_{pd}^r, N_{(d+1)q}^s | N_{pq}^j) P(w_{pd} | N_{pd}^r) P(w_{(d+1)q} | N_{(d+1)q}^s) \\
 &= \sum_{r,s} \sum_{d=p}^{q-1} P(N^j \rightarrow N^r N^s) \beta^r(p, d) \beta^s(d+1, q)
 \end{aligned}$$

Computing Inside Probabilities

	1	2	3	4	5
1	$\beta^{\text{NP}} = 0.1$		$\beta^{\text{S}} = 0.0126$		$\beta^{\text{S}} = 0.0015876$
2		$\beta^{\text{NP}} = 0.04$ $\beta^{\text{V}} = 1.0$	$\beta^{\text{VP}} = 0.126$		$\beta^{\text{VP}} = 0.015876$
3			$\beta^{\text{NP}} = 0.18$		$\beta^{\text{NP}} = 0.01296$
4				$\beta^{\text{P}} = 1.0$	$\beta^{\text{PP}} = 0.18$
5					$\beta^{\text{NP}} = 0.18$
	<i>astronomers</i>	<i>saw</i>	<i>stars</i>	<i>with</i>	<i>ears</i>

Computing Outside Probabilities

Base case: $\alpha^1(1, m) = 1$, and $\alpha^j(1, m) = 0$ for $j \neq 1$



Induction step:

$$\alpha^j(p, q) = \sum_{f,g} \sum_{e=q+1}^m \alpha^f(p, e) P_G(N^f \rightarrow N^j N^g \mid N^f) \beta^g(q+1, e) + \sum_{f,g} \sum_{e=1}^{p-1} \alpha^f(e, q) P_G(N^f \rightarrow N^g N^j \mid N^f) \beta^g(e, p-1)$$

Computing Outside Probabilities — Induction

$$\begin{aligned}
\alpha^j(p, q) &= \sum_{f,g} \sum_{e=q+1}^m P(w_{1(p-1)}, w_{(q+1)m}, N_{pe}^f, N_{pq}^j, N_{(q+1)e}^g) + \\
&\quad \sum_{f,g} \sum_{e=1}^{p-1} P(w_{1(p-1)}, w_{(q+1)m}, N_{eq}^f, N_{e(p-1)}^g, N_{pq}^j) \\
&= \sum_{f,g} \sum_{e=q+1}^m P(w_{1(p-1)}, w_{(e+1)m}, N_{pe}^f) P(N_{pq}^j, N_{(q+1)e}^g | N_{pe}^f) P(w_{(q+1)e} | N_{(q+1)e}^g) + \\
&\quad \sum_{f,g} \sum_{e=1}^{p-1} P(w_{1(e-1)}, w_{(q+1)m}, N_{eq}^f) P(N_{e(p-1)}^g, N_{pq}^j | N_{eq}^f) P(w_{e(p-1)} | N_{e(p-1)}^g) \\
&= \sum_{f,g} \sum_{e=q+1}^m \alpha^f(p, e) P_G(N^f \rightarrow N^j N^g | N^f) \beta^g(q+1, e) + \\
&\quad \sum_{f,g} \sum_{e=1}^{p-1} \alpha^f(e, q) P_G(N^f \rightarrow N^g N^j | N^f) \beta^g(e, p-1)
\end{aligned}$$

Finding the Most Likely Parse Sequence

Viterbi Algorithm

Base case: $\delta^i(p, p) = P(N^i \rightarrow w^p | N^i)$

Induction step:

$$\delta^i(p, q) = \max_{1 \leq j, k \leq n; p \leq r < q} P_G(N^i \rightarrow N^j N^k | N^i) \delta^j(p, r) \delta^k(r + 1, q)$$

$$\psi^i(p, q) = \operatorname{argmax}_{(j, k, r)} P_G(N^i \rightarrow N^j N^k | N^i) \delta^j(p, r) \delta^k(r + 1, q)$$

Termination:

$$P_G(\hat{t}) = \delta^1(1, m)$$

Path readout (by backtracing):

if $\hat{X}_\chi = N_{pq}^i$ is in the Viterbi parse, and $\psi_i(p, q) = (j, k, r)$,

then $\operatorname{left}(\hat{X}_\chi) = N_{pr}^j$, $\operatorname{right}(\hat{X}_\chi) = N_{(r+1)q}^k$

(N_{1m}^1 is the root node of the Viterbi parse.)

Learning PCFGs:

The Inside-Outside (EM) Algorithm

Combining inside and outside probabilities:

$$\begin{aligned}\alpha^j(p, q)\beta^j(p, q) &= P_G(N^1 \Rightarrow^* w_{1m}, N^j \Rightarrow^* w_{pq}) \\ &= P_G(N^1 \Rightarrow^* w_{1m})P_G(N^j \Rightarrow^* w_{pq} \mid N^1 \Rightarrow^* w_{1m})\end{aligned}$$

Denoting $\pi = P_G(N^1 \Rightarrow^* w_{1m})$, it follows that

$$P_G(N^j \Rightarrow^* w_{pq} \mid N^1 \Rightarrow^* w_{1m}) = \frac{1}{\pi} \alpha^j(p, q)\beta^j(p, q)$$

$$\begin{aligned}P_G(N^j \rightarrow N^r N^s \Rightarrow^* w_{pq} \mid N^1 \Rightarrow^* w_{1m}) \\ = \frac{1}{\pi} \sum_{d=p}^{q-1} \alpha^j(p, q)P_G(N^j \rightarrow N^r N^s \mid N^j)\beta^r(p, d)\beta^s(d+1, q)\end{aligned}$$

$$\begin{aligned}P_G(N^j \rightarrow w^k \mid N^1 \Rightarrow^* w_{1m}, w^k = w_h) \\ = \frac{1}{\pi} \alpha^j(h, h)P(w^k = w_h)\beta^j(h, h)\end{aligned}$$

The Inside-Outside Algorithm: E-step

16.

Assume that we have a set of sentences $W = \{W_1, \dots, W_\omega\}$

$$f_i(p, q, j, r, s) = \frac{1}{\pi_i} \sum_{d=p}^{q-1} \alpha_i^j(p, q) P_G(N^j \rightarrow N^r N^s \mid N^j) \beta_i^r(p, d) \beta_i^s(d+1, q)$$

$$g_i(h, j, k) = \frac{1}{\pi_i} \alpha_i^j(h, h) P(w^k = w_h) \beta_i^j(h, h)$$

$$h_i(p, q, j) = \frac{1}{\pi_i} \alpha_i^j(p, q) \beta_i^j(p, q) \text{ with } \pi_i = P_G(N^1 \Rightarrow^* W_i)$$

$$\hat{P}_G(N^j \rightarrow N^r N^s) = \sum_{i=1}^{\omega} \sum_{p=1}^{m_i-1} \sum_{q=p+1}^{m_i} f_i(p, q, j, r, s)$$

$$\hat{P}_G(N^j \rightarrow w^k) = \sum_{i=1}^{\omega} \sum_{h=1}^{m_i} g_i(h, j, k)$$

$$\hat{P}_G(N^j) = \sum_{i=1}^{\omega} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p, q, j)$$

The Inside-Outside Algorithm: M-step

$$\begin{aligned}
 P_{G'}(N^j \rightarrow N^r N^s \mid N^j) &= \frac{\hat{P}_G(N^j \rightarrow N^r N^s)}{\hat{P}_G(N^j)} \\
 &= \frac{\sum_{i=1}^{\omega} \sum_{p=1}^{m_i-1} \sum_{q=p+1}^{m_i} f_i(p, q, j, r, s)}{\sum_{i=1}^{\omega} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p, q, j)}
 \end{aligned}$$

$$\begin{aligned}
 P_{G'}(N^j \rightarrow w^k \mid N^j) &= \frac{\hat{P}_G(N^j \rightarrow w^k)}{\hat{P}_G(N^j)} \\
 &= \frac{\sum_{i=1}^{\omega} \sum_{h=1}^{m_i} g_i(h, j, k)}{\sum_{i=1}^{\omega} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p, q, j)}
 \end{aligned}$$

$P(W|G') \geq P(W|G)$ (**Baum-Welch**)

Problems with the Inside-Outside Algorithm

1. It is much *slower* than linear models like HMMs:
For each sentence of length m , the training is $O(mn)$,
where n is the number of nonterminals in G .
2. The algorithm is very sensitive to the *initialization*:
[Chiarniak, 1993] reports finding different local maxima
for each of 300 trials of a PCFG on artificial data!!
Proposed solutions: [Lari & Young, 1990]
3. Experiments suggest that satisfactory PCFG learning re-
quires *many more nonterminals* (i.e., about 3 times) than
are theoretically needed to describe the language.

“Problems” with the Learned PCFGs (Contin.)

4. There is no guarantee that the learned nonterminals will bear any resemblance to linguistically-motivated nonterminals we would use to write the grammar by hand...
5. Even if the grammar is initialized with such nonterminals, the training process may completely change the *meaning* of those nonterminals.
6. Thus, while grammar induction from unannotated corpora is possible with PCFGs, it is extremely difficult.