Probabilistic Context Free Grammars

Chapter 11 from the book

Foundations of Statistical Natural Language Processing

by Christopher D. Manning and Hinrich Schütze.

Presentation by Federico Cirett

Outline

- Introduction
- Probabilistic Context Free Grammars (PCFG)
- Questions for PCFGs
  – Probability of a sentence
  – Most likely parse for a sentence
  – Choose a rule to maximize the prob. of a sentence
- Training a PCFG
  – Inside-Outside algorithm

Introduction

- The quest for finding structure in language
  - Linguistics
  - Noam Chomsky 1950’s - 1960’s CFGs
  - Booth and Thomson 1969-1973 & others
- Uses of PCFGs
  - Speech recognition
  - Optical character recognition
  - Word grammar checker
  - Automatic translation
  - DNA sequencing

Quick Example of parse tree (CFG)

```
Start point  ------>  IP (or S)
                       /   \
Non-
terminal     ------>  NP    VP
                      |     |  \
Terminal     ------>  N     V   \
                      |     |    PP
                     She    is   /  \
                                /    \
                               P      NP
                               |      / \
                               on   Det  N
                                     |   |
                                    the  computer
```

- S = Start point
- IP = Inflectional phrase (sentence)
- NP = Noun phrase
- N = Noun
- VP = Verb phrase
- PP = Prepositional phrase
- P = Preposition
- Det = Determiner

Probabilistic or stochastic context-free grammars (PCFGs)

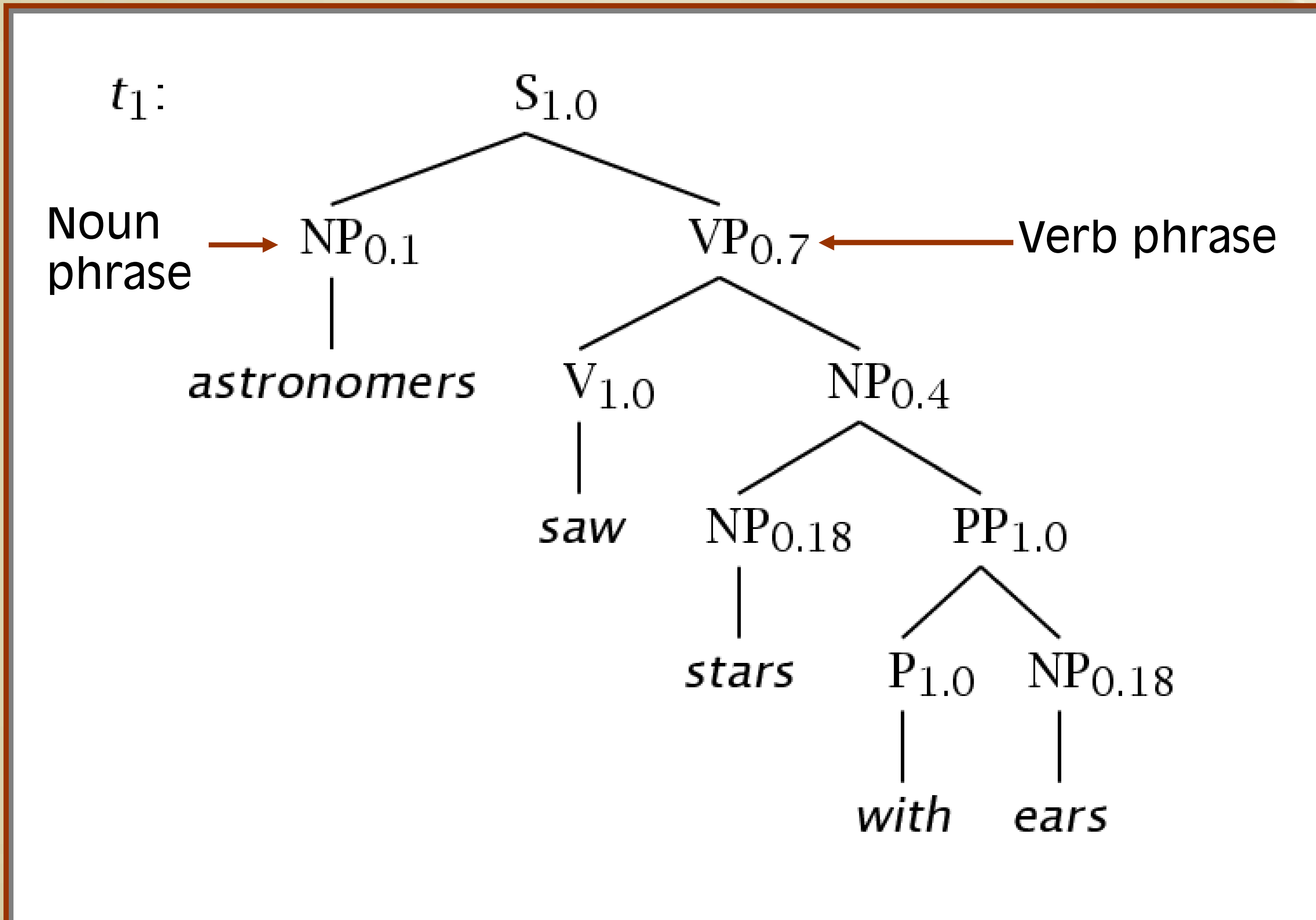
- G = (T, N, S, R, P)
  - T is set of terminals
  - N is set of nonterminals
  - S is the start symbol (one of the nonterminals)
  - R is rules/productions of the form X $\rightarrow \gamma$
  - P(R) gives the probability of each rule.

$$\forall X \in N, \sum_{X \rightarrow \gamma \in R} P(X \rightarrow \gamma) = 1$$

A Simple PCFG (in Chomsky Normal Form)

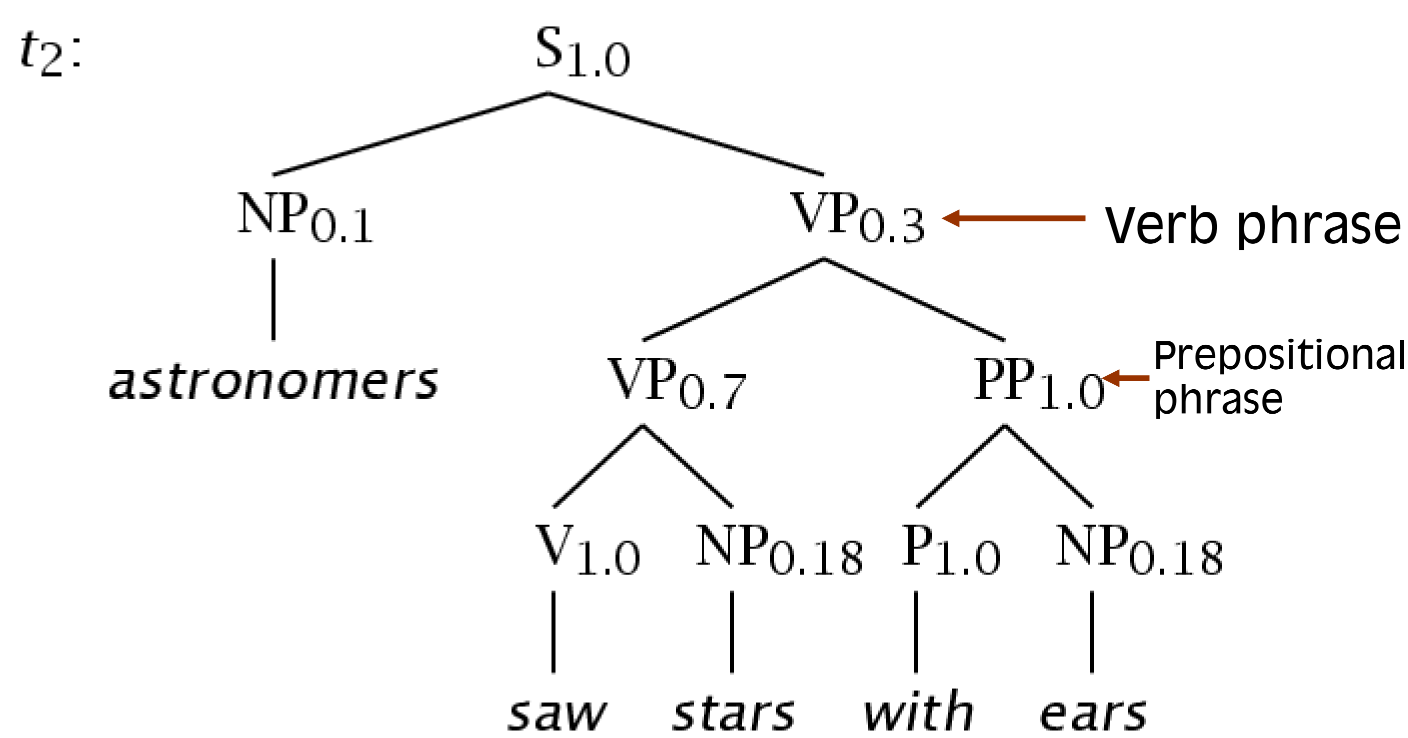
S	$\rightarrow$	NP VP	1.0	NP	$\rightarrow$	NP PP	0.4
VP	$\rightarrow$	V NP	0.7	NP	$\rightarrow$	*astronomers*	0.1
VP	$\rightarrow$	VP PP	0.3	NP	$\rightarrow$	*ears*	0.18
PP	$\rightarrow$	P NP	1.0	NP	$\rightarrow$	*saw*	0.04
P	$\rightarrow$	*with*	1.0	NP	$\rightarrow$	*stars*	0.18
V	$\rightarrow$	*saw*	1.0	NP	$\rightarrow$	*telescope*	0.1

Parse tree 1:



t1:
S1.0
Noun phrase
NP0.1
VP0.7
Verb phrase
astronomers
V1.0
NP0.4
saw
NP0.18
PP1.0
stars
P1.0
NP0.18
with
ears

Parse tree 2:



t2:
S1.0
NP0.1
VP0.3
Verb phrase
astronomers
VP0.7
PP1.0
Prepositional phrase
V1.0
NP0.18
P1.0
NP0.18
saw
stars
with
ears

The 3 questions of PCFG's

- What is the probability of a sentence w_{1m} to a grammar G: $P(w_{1m} \mid G)$?
- What is the most likely parse for a sentence
  arg $\max_t P(t \mid w_{1m}, G)$?
- How can we choose rule probabilities for the grammar G that maximize the probability of a sentence arg $\max_G$ $\mathrm{P}(w_{1m} \mid G)$?

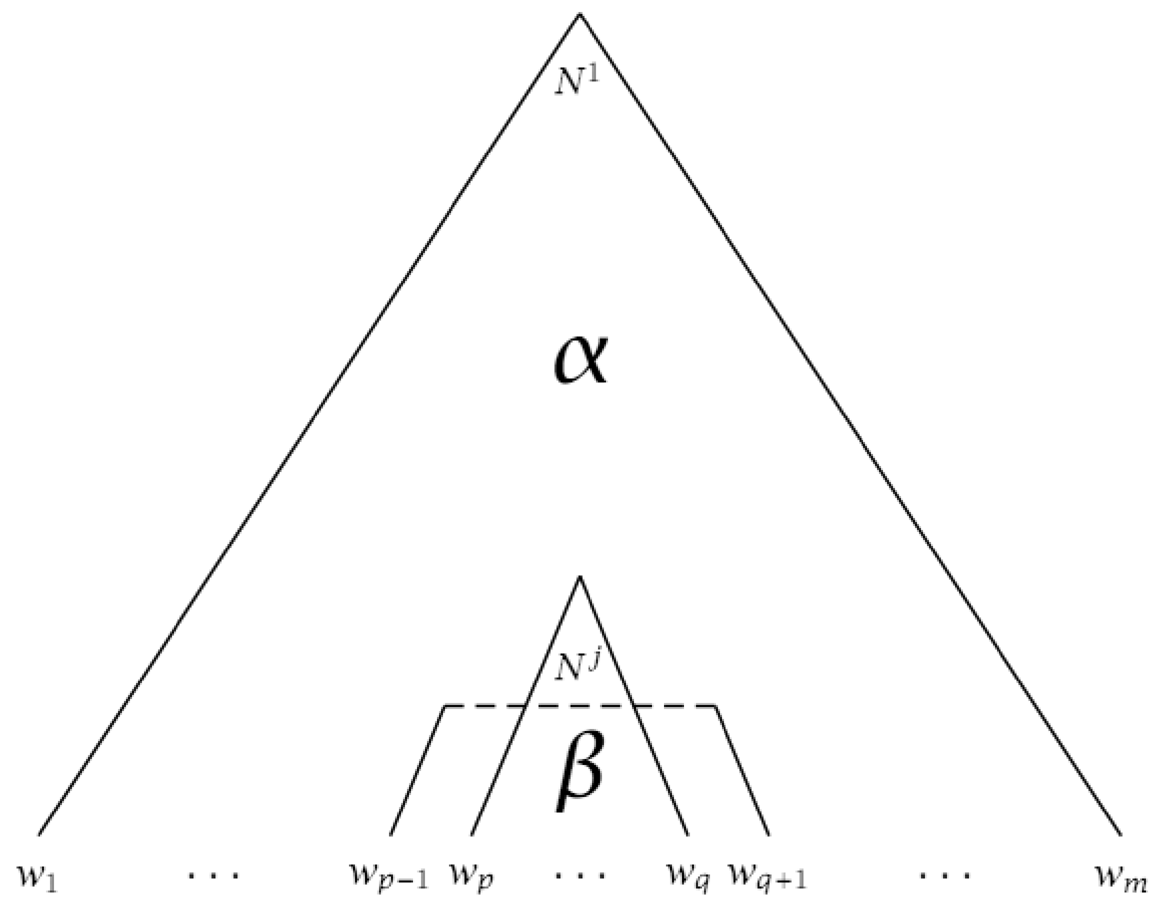


Figure 11.3 Inside and outside probabilities in PCFGs.

Outside probability $\alpha_j(p,q) = P(w_{1(p-1)}, N^j_{pq}\, w_{(q+1)} \mid G)$

Inside probability $\beta_j\,(p,q) = P(w_{pq} \mid N^j_{pq}, G)$

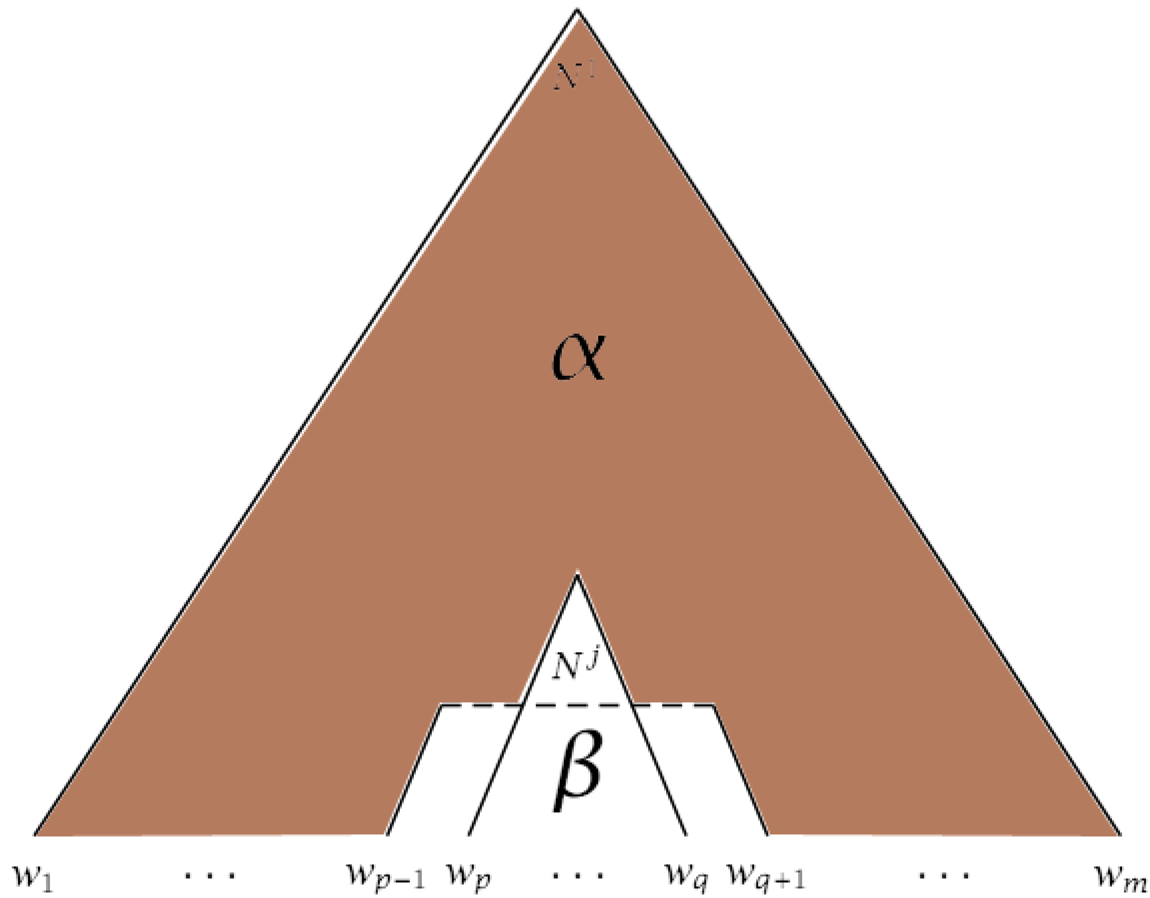


Figure 11.3 Inside and outside probabilities in PCFGs.

Outside probability $\alpha_j(p,q) = P(w_{1(p-1)}, N^j_{pq}\ w_{(q+1)} \mid G)$

Inside probability $\beta_j\ (p,q) = P(w_{pq} \mid N^j_{pq}, G)$

The probability of a String

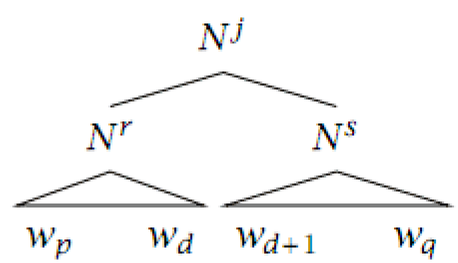
- Inside algorithm:

$$
\begin{aligned}
P(w_{1m}|G) &= P(N^1 \stackrel{*}{\Longrightarrow} w_{1m}|G) \\
&= P(w_{1m}|N^1_{1m}, G) \quad = \beta_1(1,m)
\end{aligned}
$$

Base case: We want to find $\beta_j(k,k)$ (the probability of a rule $N^j \rightarrow w_k$):

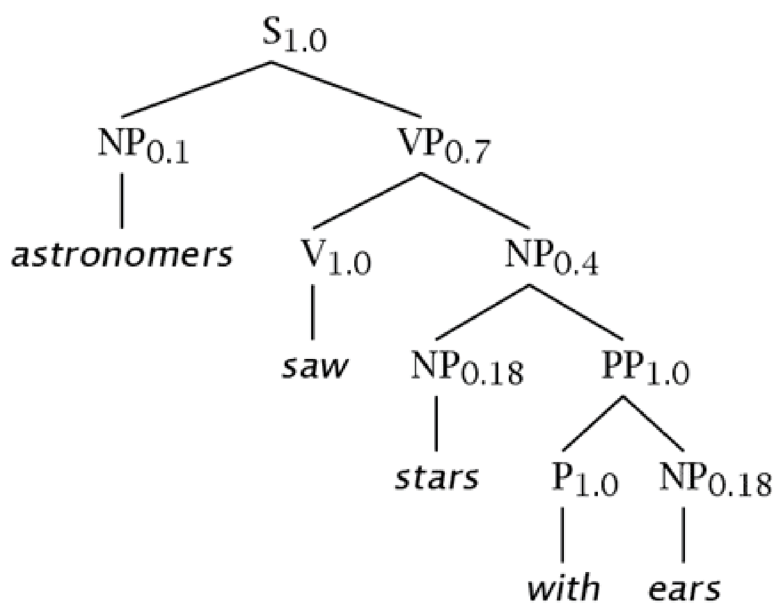
$$
\begin{aligned}
\beta_j(k,k) &= P(w_k|N^j_{kk}, G) \\
&= P(N^j \rightarrow w_k|G)
\end{aligned}
$$

Induction:



The probability of a String

	1	2	3	4	5
1	$\beta_{NP} = 0.1$		$\beta_S = 0.0126$		$\beta_S = 0.0015876$
2		$\beta_{NP} = 0.04$ $\beta_V = 1.0$	$\beta_{VP} = 0.126$		$\beta_{VP} = 0.015876$
3			$\beta_{NP} = 0.18$		$\beta_{NP} = 0.01296$
4				$\beta_P = 1.0$	$\beta_{PP} = 0.18$
5					$\beta_{NP} = 0.18$
	astronomers	*saw*	*stars*	*with*	*ears*



Using outside probabilities

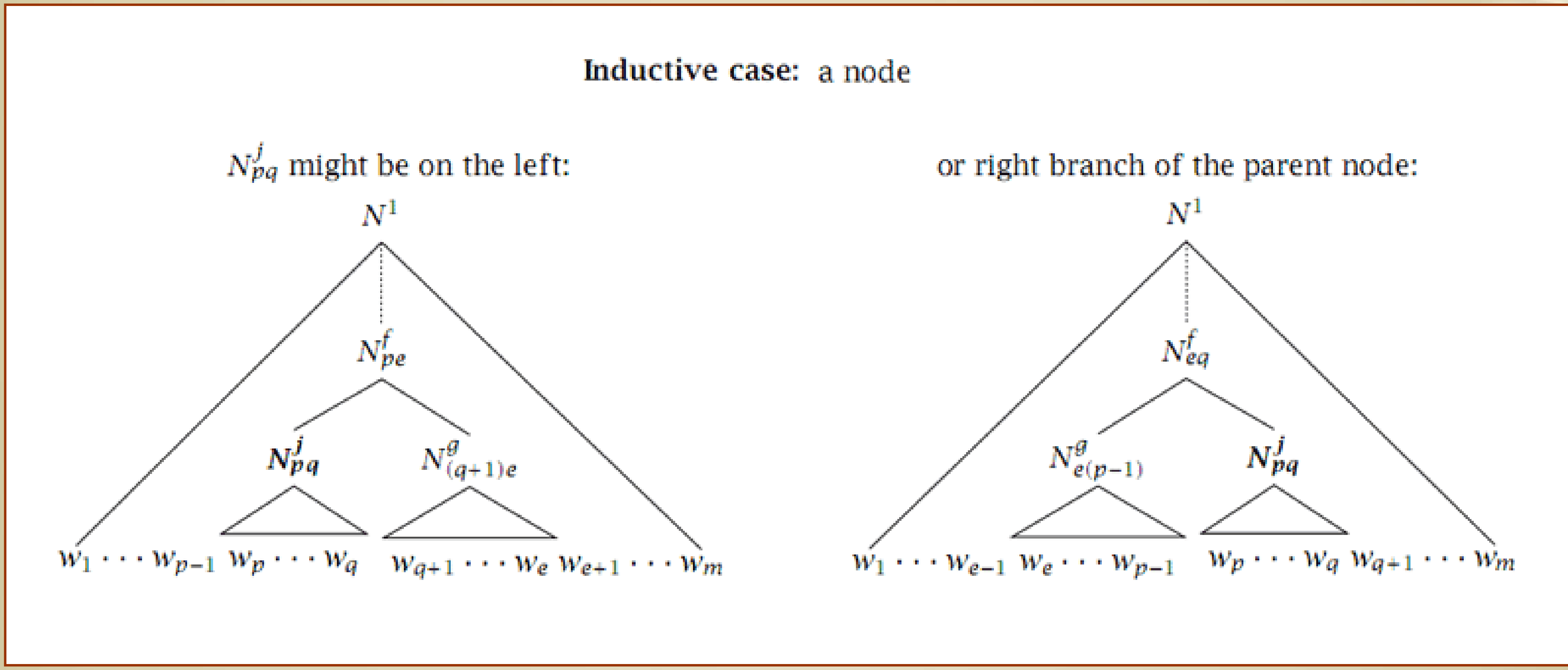
For any k, $1 \le k \le m$,

$$\begin{aligned}
P(w_{1m}|G) &= \sum_j P(w_{1(k-1)}, w_k, w_{(k+1)m}, N^j_{kk}|G) \\
&= \sum_j P(w_{1(k-1)}, N^j_{kk}, w_{(k+1)m}|G) \\
&\qquad \times P(w_k|w_{1(k-1)}, N^j_{kk}, w_{(k+1)n}, G) \\
&= \sum_j \alpha_j(k,k) P(N^j \rightarrow w_k)
\end{aligned}$$

Base Case: The base case is the probability of the root of the tree being nonterminal N^i with nothing outside it:

$$\begin{aligned}
\alpha_1(1,m) &= 1 \\
\alpha_j(1,m) &= 0 \quad \text{for } j \neq 1
\end{aligned}$$

Using outside probabilities



Using outside probabilities

$$
\begin{aligned}
\alpha_j(p,q) &= \Big[\sum_{f,g\neq j}\sum_{e=q+1}^{m} P(w_{1(p-1)}, w_{(q+1)m}, N^f_{pe}, N^j_{pq}, N^g_{(q+1)e})\Big] \\
&\quad + \Big[\sum_{f,g}\sum_{e=1}^{p-1} P(w_{1(p-1)}, w_{(q+1)m}, N^f_{eq}, N^g_{e(p-1)}, N^j_{pq})\Big] \\
&= \Big[\sum_{f,g\neq j}\sum_{e=q+1}^{m} P(w_{1(p-1)}, w_{(e+1)m}, N^f_{pe}) P(N^j_{pq}, N^g_{(q+1)e} | N^f_{pe}) \\
&\quad \times P(w_{(q+1)e} | N^g_{(q+1)e})\Big] + \Big[\sum_{f,g}\sum_{e=1}^{p-1} P(w_{1(e-1)}, w_{(q+1)m}, N^f_{eq}) \\
&\quad \times P(N^g_{e(p-1)}, N^j_{pq} | N^f_{eq}) P(w_{e(p-1)} | N^g_{e(p-1)})\Big] \\
&= \Big[\sum_{f,g\neq j}\sum_{e=q+1}^{m} \alpha_f(p,e) P(N^f \to N^j\, N^g)\, \beta_g(q+1,e)\Big] \\
&\quad + \Big[\sum_{f,g}\sum_{e=1}^{p-1} \alpha_f(e,q) P(N^f \to N^g\, N^j)\, \beta_g(e,p-1)\Big]
\end{aligned}
$$

$$
\begin{aligned}
\alpha_j(p,q)\beta_j(p,q) &= P(w_{1(p-1)}, N^j_{pq}, w_{(q+1)m} | G) P(w_{pq} | N^j_{pq}, G) \\
&= P(w_{1m}, N^j_{pq} | G)
\end{aligned}
$$

$$
P(w_{1m}, N_{pq} | G) = \sum_j \alpha_j(p,q)\beta_j(p,q)
$$

Finding the most likely parse for a sentence

$\delta_i(p,q)$ = the highest inside probability parse of a subtree N^i_{pq}

1. Initialization

$$\delta_i(p,p) = P(N^i \to w_p)$$

2. Induction

$$\delta_i(p,q) = \max_{\substack{1 \le j,k \le n \\ p \le r < q}} P(N^i \to N^j \ N^k)\delta_j(p,r)\delta_k(r+1,q)$$

Store backtrace

$$\psi_i(p,q) = \arg\max_{(j,k,r)} P(N^i \to N^j \ N^k)\delta_j(p,r)\delta_k(r+1,q)$$

3. Termination and path readout (by backtracking).

$$P(\hat{t}) = \delta_1(1,m)$$

Training a PCFG with the Inside-Outside algorithm

$$\hat{P}(N^j \rightarrow \zeta) = \frac{C(N^j \rightarrow \zeta)}{\sum_{\gamma} C(N^j \rightarrow \gamma)}$$

$$\begin{aligned}\alpha_j(p,q)\beta_j(p,q) &= P(N^1 \stackrel{*}{\Rightarrow} w_{1m}, N^j \stackrel{*}{\Rightarrow} w_{pq} | G) \\ &= P(N^1 \stackrel{*}{\Rightarrow} w_{1m} | G) P(N^j \stackrel{*}{\Rightarrow} w_{pq} | N^1 \stackrel{*}{\Rightarrow} w_{1m}, G)\end{aligned}$$

$$\pi = P(N^1 \stackrel{*}{\Rightarrow} w_{1m})$$

$$P(N^j \stackrel{*}{\Rightarrow} w_{pq} | N^1 \stackrel{*}{\Rightarrow} w_{1m}, G) = \frac{\alpha_j(p,q)\beta_j(p,q)}{\pi}$$

$$E(N^j \text{ is used in the derivation}) = \sum_{p=1}^{m} \sum_{q=p}^{m} \frac{\alpha_j(p,q)\beta_j(p,q)}{\pi}$$

$\forall r, s, p < q$:

$$P(N^j \rightarrow N^r\ N^s \stackrel{*}{\Rightarrow} w_{pq} | N^1 \stackrel{*}{\Rightarrow} w_{1m}, G)$$

$$= \frac{\sum_{d=p}^{q-1} \alpha_j(p,q) P(N^j \rightarrow N^r\ N^s) \beta_r(p,d) \beta_s(d+1,q)}{\pi}$$

Training a PCFG with the Inside-Outside algorithm

$$E(N^j \to N^r\, N^s, N^j \text{ used}) = \frac{\sum_{p=1}^{m-1}\sum_{q=p+1}^{m}\sum_{d=p}^{q-1}\alpha_j(p,q)P(N^j \to N^r\, N^s)\beta_r(p,d)\beta_s(d+1,q)}{\pi}$$

Now for the maximization step, we want:

$$P(N^j \to N^r\, N^s) = \frac{E(N^j \to N^r\, N^s, N^j \text{ used})}{E(N^j \text{ used})}$$

So, the reestimation formula is:

$$\hat{P}(N^j \to N^r\, N^s) = \frac{\sum_{p=1}^{m-1}\sum_{q=p+1}^{m}\sum_{d=p}^{q-1}\alpha_j(p,q)P(N^j \to N^r\, N^s)\beta_r(p,d)\beta_s(d+1,q)}{\sum_{p=1}^{m}\sum_{q=p}^{m}\alpha_j(p,q)\beta_j(p,q)}$$

Similarly for preterminals,

$$P(N^j \to w^k | N^1 \stackrel{*}{\Longrightarrow} w_{1m}, G) = \frac{\sum_{h=1}^{m}\alpha_j(h,h)P(N^j \to w_h, w_h = w^k)}{\pi} = \frac{\sum_{h=1}^{m}\alpha_j(h,h)P(w_h = w^k)\beta_j(h,h)}{\pi}$$

$$\hat{P}(N^j \to w^k) = \frac{\sum_{h=1}^{m}\alpha_j(h,h)P(w_h = w^k)\beta_j(h,h)}{\sum_{p=1}^{m}\sum_{q=p}^{m}\alpha_j(p,q)\beta_j(p,q)}$$

Training a PCFG with the Inside-Outside algorithm

$$f_i(p,q,j,r,s) = \frac{\sum_{d=p}^{q-1} \alpha_j(p,q) P(N^j \to N^r N^s) \beta_r(p,d) \beta_s(d+1,q)}{P(N^1 \stackrel{*}{\Longrightarrow} W_i | G)}$$

$$g_i(h,j,k) = \frac{\alpha_j(h,h) P(w_h = w^k) \beta_j(h,h)}{P(N^1 \stackrel{*}{\Longrightarrow} W_i | G)}$$

$$h_i(p,q,j) = \frac{\alpha_j(p,q) \beta_j(p,q)}{P(N^1 \stackrel{*}{\Longrightarrow} W_i | G)}$$

$$\hat{P}(N^j \to N^r\, N^s) = \frac{\sum_{i=1}^{\omega} \sum_{p=1}^{m_i - 1} \sum_{q=p+1}^{m_i} f_i(p,q,j,r,s)}{\sum_{i=1}^{\omega} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p,q,j)}$$ ← Estimation

and

$$\hat{P}(N^j \to w^k) = \frac{\sum_{i=1}^{\omega} \sum_{h=1}^{m_i} g_i(h,j,k)}{\sum_{i=1}^{\omega} \sum_{p=1}^{m_i} \sum_{q=p}^{m_i} h_i(p,q,j)}$$ ← Reestimation

$$P(W | G_{i+1}) \geq P(W | G_i).$$

Problems with the Inside-Outside algorithm

- Training PCFGs is way slow: For each sentence, each iteration of training is $O(m^3n^3)$, being m the length of the sentence and n the number of non-terminals in the grammar.
- Algorithm is very sensitive to initialization parameters.

Some features of PCFGs

- Better than grammars based on HMMs, as the diversity and size of a corpus of texts expands.
- PCFG's don't take lexical context into account.
- For that reason, it does not give a plausibility of different parses.
- Robustness: Real text has grammatical errors. Just give implausible sentences a low probability
- PCFG's have certain biases: a smaller tree has more probability than a larger tree.

Acknowledgments

- Some material of this presentation was taken from Lecture 2 on Statistical Parsing by Christopher Manning, as well some graphics and examples from Chapter 11 of the book *Foundations of Statistical Natural Language Processing* by Christopher D. Manning and Hinrich Schütze