第八章　句法分析

No.95, Zhongguancun East Road
Beijing 100080, China

http://www.ia.ac.cn
Tel. No.: +86-10-6255 4263

8.1 概述

任务：句法分析（syntactic parsing）的任务就是识别句子的句法结构（syntactic structure）。

- 例如（前面第四章的例子）：

  句子：他还提出一系列具体措施的政策要点。

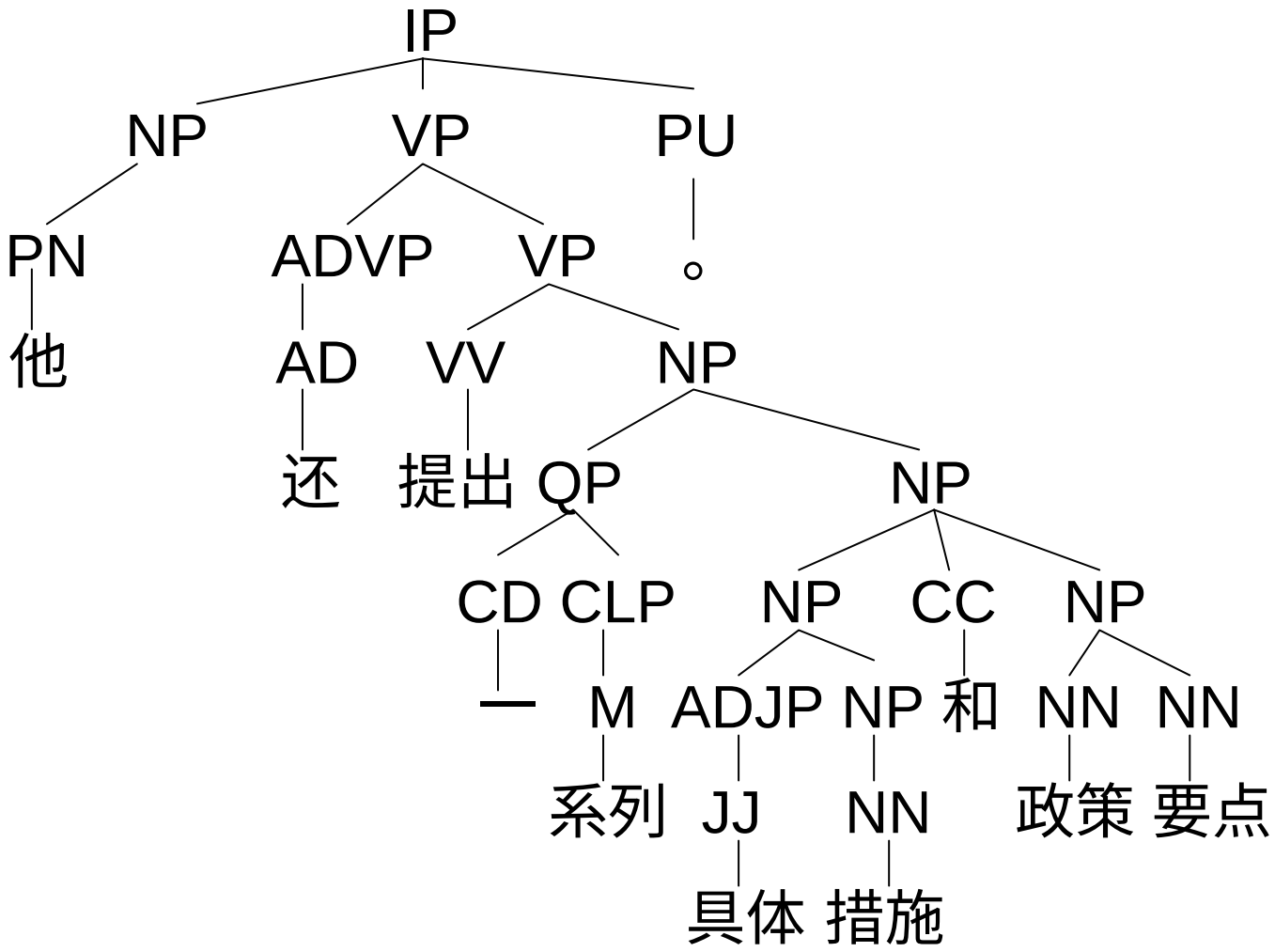
  词性标注：他/PN 还/AD 提出/VV 一/CD 系列/M 具体/JJ 措施/NN 和/CC 政策/NN 要点/NN 。/PU

8.1 概述

结构：(IP (NP-SBJ (PN 他))
(VP (ADVP (AD 还))
(VP (VV 提出))
(NP-OBJ(QP (CD 一)
(CLP (M 系列)))
(NP (NP(ADJP (JJ 具体)
(NP (NN 措施)))
(CC 和)
(NP (NN 政策)
(NN 要点))))))
(PU 。))

8.1 概述

树状表示：


IP
NP
VP
PU
PN
ADVP
VP
。
他
AD
VV
NP
还
提出
QP
NP
CD
CLP
NP
CC
NP
一
M
ADJP
NP
和
NN
NN
系列
JJ
NN
政策
要点
具体
措施

8.1 概述

- 目标：实现高正确率、高鲁棒性 (robustness)、高速度的自动句法分析过程。
- 困难：自然语言中存在大量的复杂的结构歧义（structural ambiguity）。

8.1 概述

- 结构歧义

例如：1) I saw a boy in the park.

[I saw a boy] in the park.

I saw a [boy in the park].

2) I saw a boy in the park with a telescope.

3) I saw a boy swimming on the bridge.

4) 关于鲁迅的文章。

5) 把重要的书籍和手稿带走了。

8.1 概述

英语中的结构歧义随介词短语组合个数的增加而不断加深的，这个组合个数我们称之为开塔兰数(Catalan number，记作C_N)。

如果句子中存在这样 n (n为自然数)个介词短语，C_N 可以由下式获得 [Samuelsson, 2000]：

$$C_N = \begin{pmatrix} 2n \\ n \end{pmatrix} \frac{1}{n+1} = \frac{(2n)!}{(n!)^2 (n+1)}$$

8.1 概述

- 基本方法：

1）基于CFG规则的分析方法：

- 线图分析法 (chart parsing)
- Earley (厄尔利)算法
- CYK
- LR 算法 / Tomita 算法 … …

— Top-down: Depth-first/- Breadth-first

— Bottom-up

8.1 概述

2) 基于概率上下文无关文法的分析方法

— Probabilistic Context-Free Grammar (PCFG)

Rule: $A \rightarrow \alpha,\ p$

8.2 线图分析法

- 三种方法
  - 自底向上（**Bottom-up**）
  - 从上到下（**Top-down**）
  - 从上到下和从下到上结合

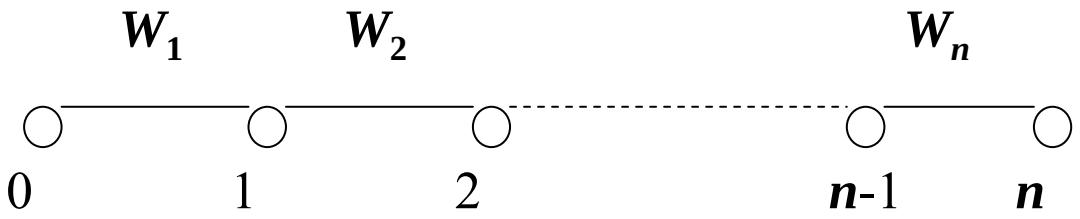
8.2 线图分析法

- 自底向上的 Chart 分析算法

给定一组 CFG 规则：$XP \rightarrow \alpha_1 \ldots \alpha_n \quad (n \geq 1)$

给定一个句子的词性序列：$S = W_1 W_2 \cdots W_n$

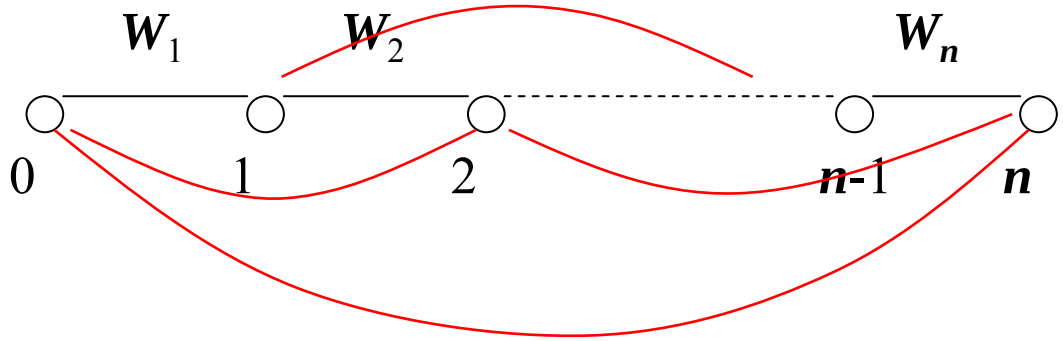
构造一个线图：一组结点和边的集合；



建立一个两维表：记录每一条边的起始位置和终止位置。

8.2 线图分析法

执行操作：查看任意相邻几条边上的词性串是否与某条重写规则的右部相同，如果相同，则增加一条新的边跨越原来相应的边，新增加边上的标记为这条重写规则的头（左部）。重复这个过程，直到没有新的边产生。



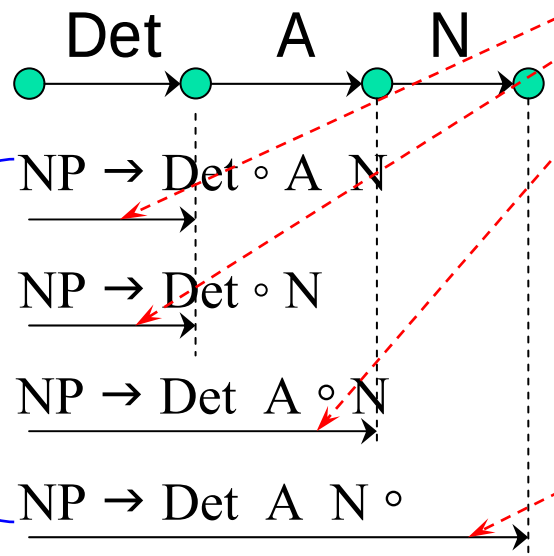
8.2 线图分析法

点规则：设有规则：NP → Det A N

NP → Det N

NP → A N

句子: The good book



引入“点规则”用于表示规则右部被归约(reduce)的程度

活性边(活动弧)：

规则右部未被完全匹配

非活性边 (非活动弧，或完成弧)：

规则右部已被完全匹配

8.2 线图分析法

❑ 数据结构

1）线图（Chart）：保存分析过程中已经建立的成分（包括终结符和非终结符）、位置（包括起点和终点）。通常以 $n \times n$ 的数组表示（n 为句子包含的词数）。

2）代理表（待处理表）（Agenda）：记录刚刚得到的一些重写规则所代表的成分，这些重写规则的右端符号串与输入词性串（或短语标志串）中的一段完全匹配。通常以栈或线性队列表示。

8.2 线图分析法

3）活动边集（ActiveArc）：记录那些右端符号串与输入串的某一段，但还未完全匹配的重写规则。通常以数组或列表存储。

8.2 线图分析法

❑ 算法描述：

从输入串的起始位置到最后位置，循环执行如下步骤：

（1）如果待处理表(Agenda)为空，则找到下一个位置上的词，将该词对应的(所有)词类X 附以 (i, j) 后放到待处理表中，即$X\ (i, j)$。其中，i, j 分别是该词的起始位置和终止位置，$j > i$，$j - i$ 为该词的长度。

（2）从 Agenda 中取出一个元素，其标志成分为 X，位置跨度为(i, j);

（3）对于每条规则 $A \rightarrow X\gamma$，将 $A \rightarrow X \circ \gamma$ 加入活动边集 ActiveArc 的(i, j) 位置中，然后调用 扩展弧子程序。

8.2 线图分析法

□ 扩展弧子程序：

(a) 将 X 插入图表(Chart)的 (i, j) 位置中。

(b) 对于每一个活动边集 (ActiveArc)中位置为(k, i) $(i > k)$ 的点规则，如果该规则具有如下形式：$A \rightarrow \alpha \circ X$, 如果$A=S$, 则把 $S(1, n+1)$ 加入到 Chart 中，并给出一个完整的分析结果；否则，则将 A 加入到待处理表的 (k, j) 位置中。

(c) 对于每个活动边集位置为(k, i) 的点规则：$A \rightarrow \alpha \circ X\beta$，则将 $A \rightarrow \alpha X \circ \beta$ 加入到活动边集 (k, j)位置中。

8.2 线图分析法

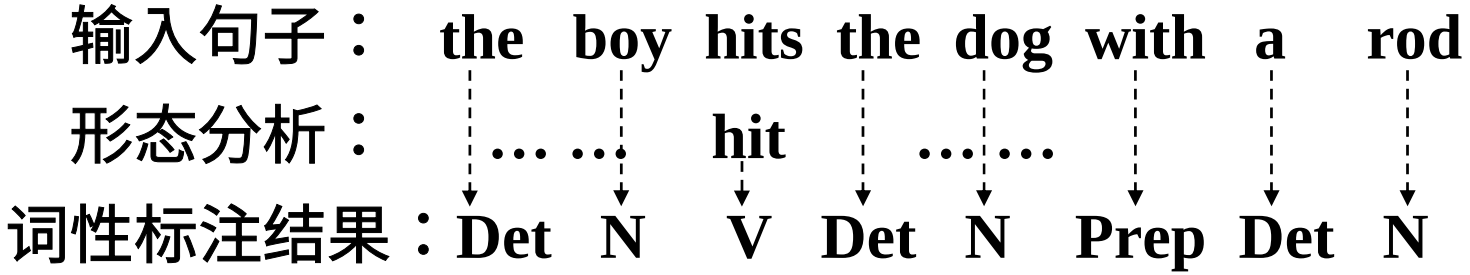
例： G (S): S → NP VP

NP → Det N

VP → V NP

VP → VP PP

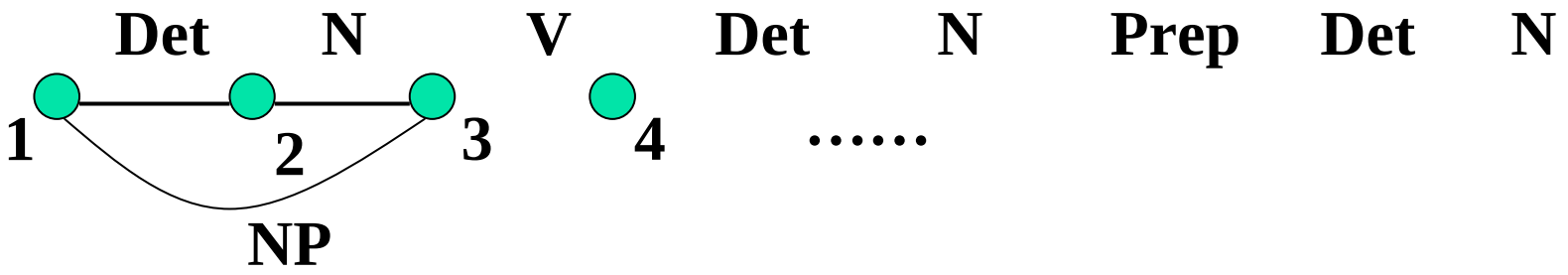
PP → Prep NP



8.2 线图分析法

分析过程：

Agenda	ActiveArc	Chart
Det (1, 2)	NP → Det ◦ N (1,2)	Det (1, 2)
N (2, 3)	NP → Det N◦ (1,3)	N (2, 3)
NP (1, 3)	S → NP ◦ VP (1, 3)	NP (1, 3)



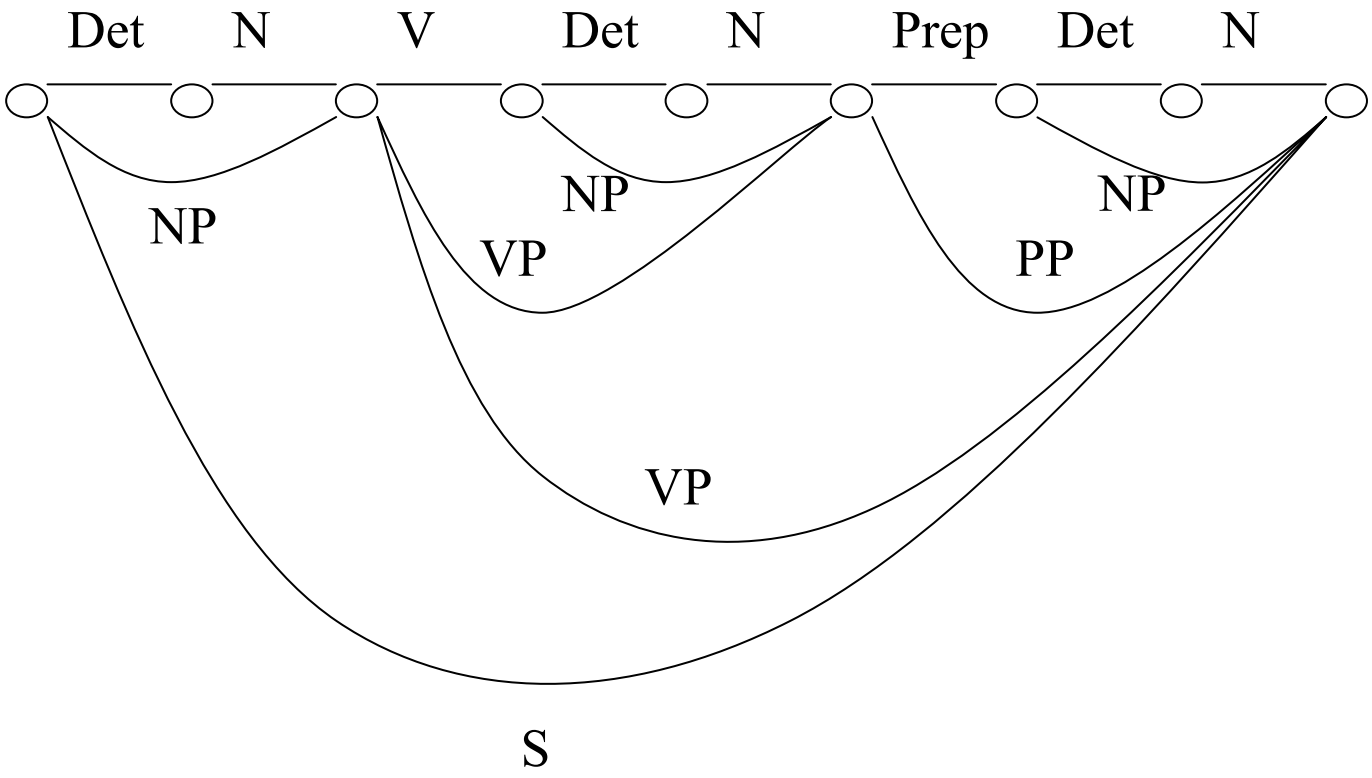
8.2 线图分析法

分析过程：

Agenda	ActiveArc	Chart
Det (1, 2)	NP → Det ◦ N (1,2)	Det (1, 2)
N (2, 3)	NP → Det N◦ (1,3)	N (2, 3)
NP (1, 3)	S → NP ◦ VP (1, 3)	NP (1, 3)
V (3, 4)	VP → V ◦ NP (3, 4)	V (3, 4)
Det (4,5)	NP → Det ◦ N (4,5)	Det (4,5)
N (5,6)	NP → Det N ◦ (4,6)	N (5,6)
NP(4,6)	S → NP ◦ VP (4, 6)	NP (4, 6)
VP (3,6)	VP → V NP ◦ (3, 6)	VP (3,6)
	S → NP VP ◦ (1, 6)	
… …	VP → VP ◦ PP (3,6)	… …

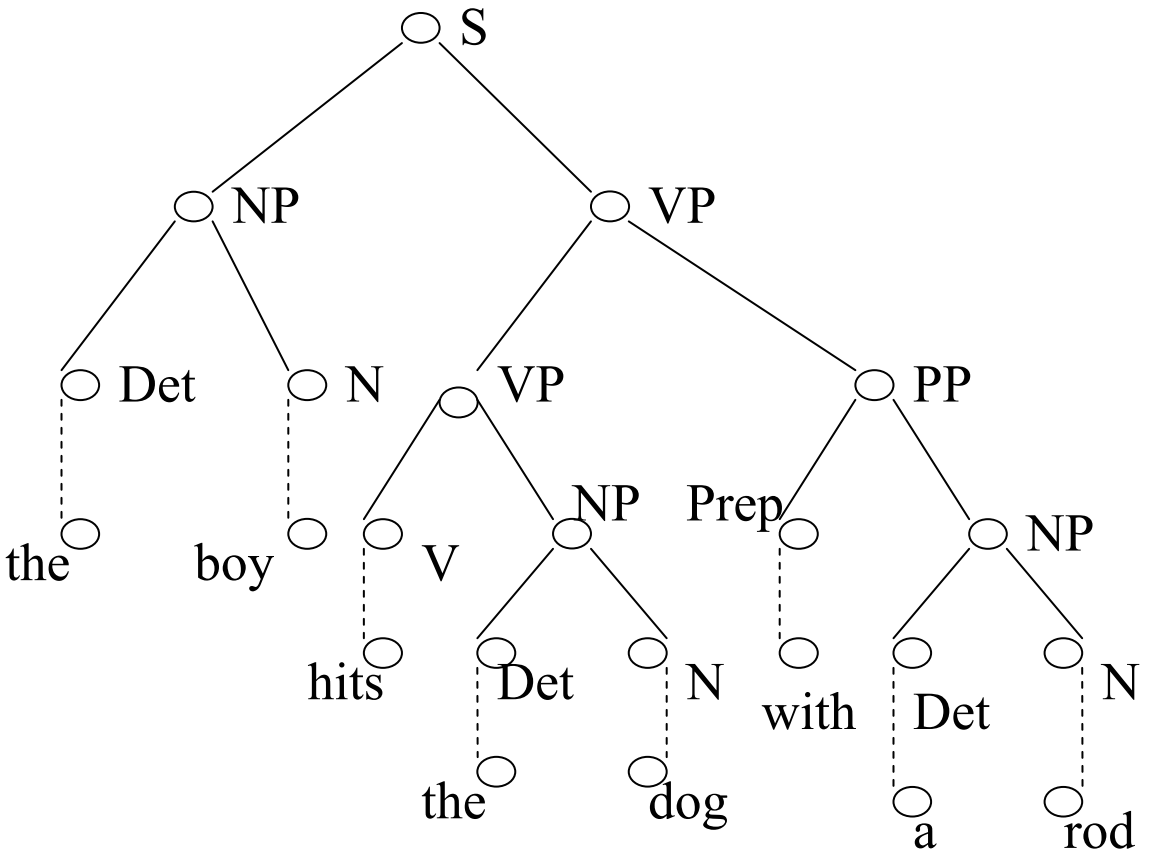
8.2 线图分析法

最后分析结果：



8.2 线图分析法

分析结果的直观图：



8.2 线图分析法

Chart parsing 评价

- 优点：
  - 算法简单，容易实现，开发周期短
- 弱点：
  - 算法效率低，时间复杂度为 $K^* n^3$ (n为句子长度， K 为常量)。
  - 需要高质量的规则，分析结果与规则质量密切相关；
  - 难以区分歧义结构。

8.3 CYK分析算法

❑ Coke-Younger-Kasami (CYK) 算法

- 对 Chomsky 文法进行范式化：

  $A \rightarrow a$ 或 $A \rightarrow BC$

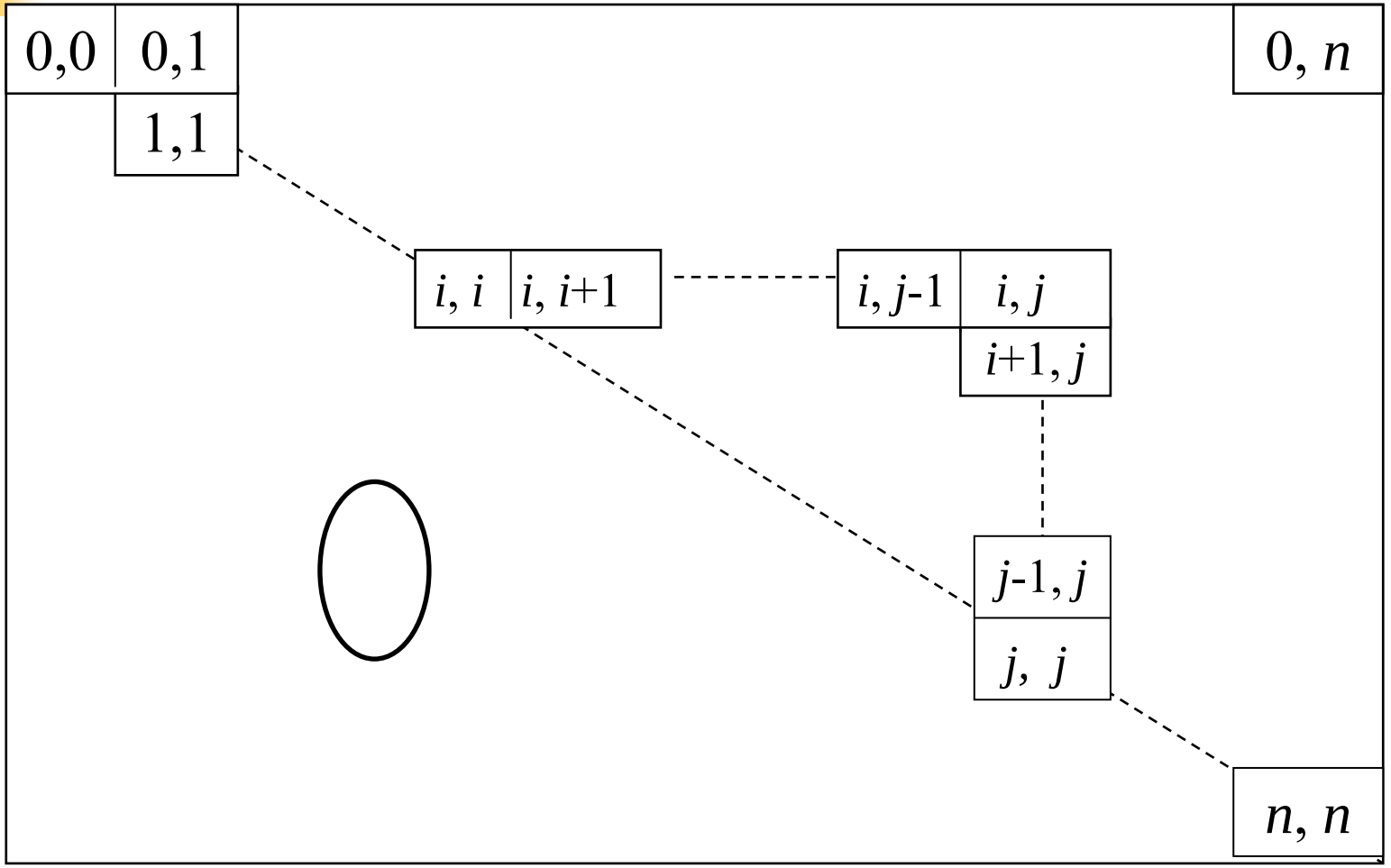
  $A, B, C \in V_N$, $a \in V_T$, $G=(V_N, V_T, P, S)$

- 自下而上的分析方法
- 构造 $(n+1)\times(n+1)$ 识别矩阵，n为输入句子长度。假设输入句子 $x=a_1a_2\ldots a_n$，$n=|x|$。

8.3 CYK分析算法

- 识别矩阵的构成
  - 方阵对角线以下全部为0；
  - 主对角线以上的元素由文法G的非终结符构成；
  - 主对角线上的元素由输入句子的终结符构成。

8.3 CYK分析算法



8.3 CYK分析算法

❑ 识别矩阵构造步骤

(1) 首先构造主对角线，令 $t_{0,0}=0$，然后，从$t_{1,1}$ 到 $t_{n,n}$ 在主对角线的位置上依次放入输入句子x 的单词。

(2) 构造主对角线以上紧靠主对角线的元素$t_{i,i+1}$, 其中，$i = 0, 1, 2, \ldots, n\text{-}1$。

对于输入句子 $x = a_1 a_2 \ldots a_n$，从 a_1 开始分析。

8.3 CYK分析算法

如果在文法G 的产生式集中有一条规则：

$$A \rightarrow a_1$$

则 $t_{0,1}$=A。

依此类推，如果有 $A \rightarrow a_{i+1}$，则 $t_{i,\ i+1}$ =A。

即，对于主对角线上的每一个终结符 a_i，所有可能推导出它的非终结符写在它的右边主对角线上方的位置上。

8.3 CYK分析算法

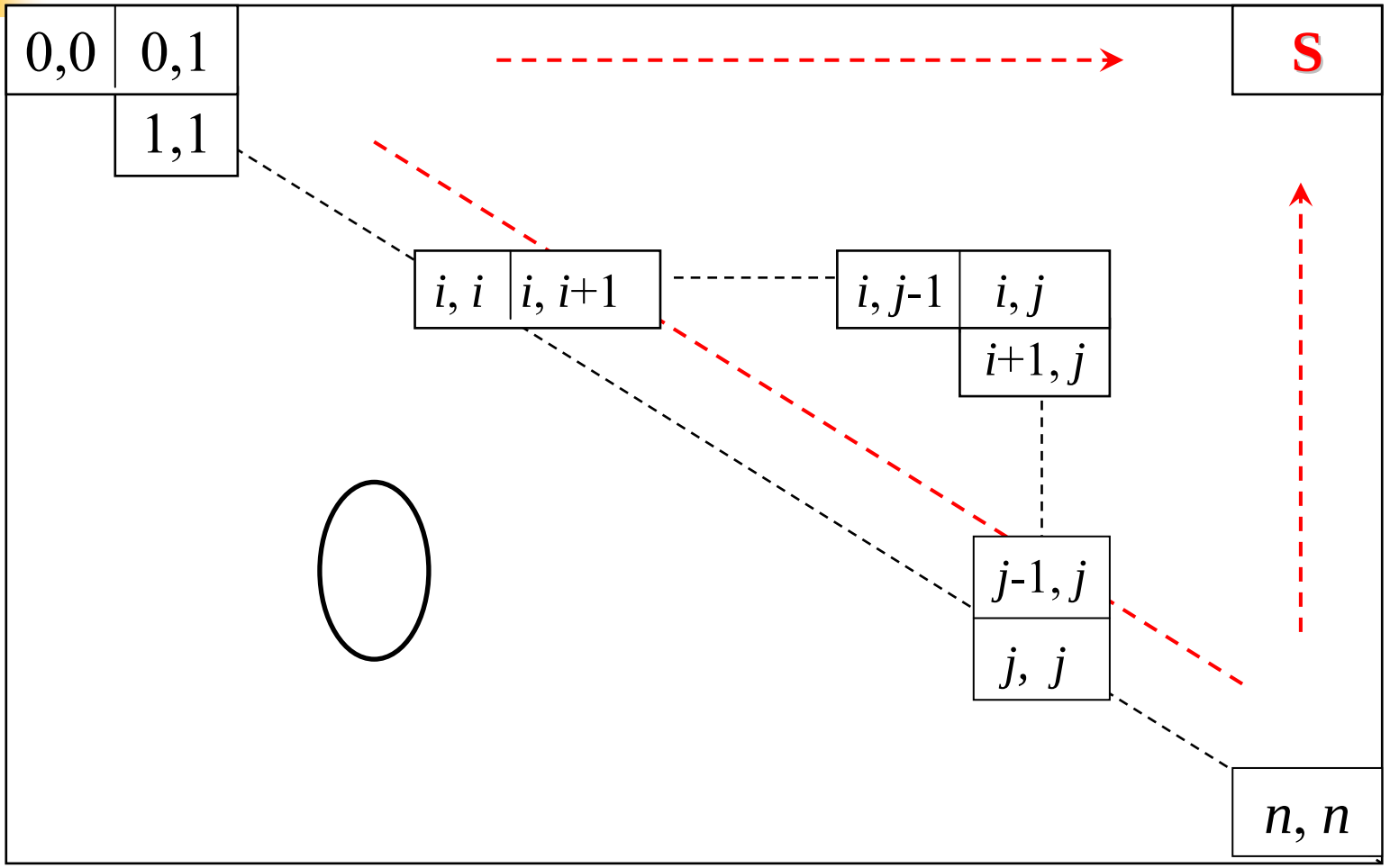
(3) 按平行于主对角线的方向，一层一层地向上填写矩阵的各个元素 $t_{i,j}$，其中，$i = 0,1, \ldots, n\text{-}d$,

$j = d+i$，$d=2, 3, \ldots, n$。

如果存在一个正整数K, $i+1 \le K \le j-1$，在文法G的规则集中有产生式 $A \to BC$, 并且，$B \in t_{i,k}$, $C \in t_{k,j}$， 那么，将A写到矩阵 $t_{i,j}$ 位置上。

判断句子 x 由文法G所产生的充要条件是 $t_{0,n}=S$。

8.3 CYK分析算法



8.3 CYK分析算法

❑ CYK 算法评价

- ◆ 优点
  - ➢ 简单易行，执行效率高
- ◆ 弱点
  - ➢ 必须对文法进行范式化处理
  - ➢ 无法区分歧义

8.4 概率上下文无关文法

❑ PCFG (Probabilistic CFG) 规则

(PCFG 也写作：Stochastic CFG, SCFG)

形式： $A \rightarrow \alpha, \quad P$

约束： $\sum_{\alpha} P(A \rightarrow \alpha) = 1$

例如：NP → NN NN, 0.60

NP → NN CC NN, 0.40

CD → QP, 0.99

CD → LST, 0.01

8.4 概率上下文无关文法

◆例-1：S → NP VP, 1.00　　NP → NP PP, 0.40

NP → astronomers, 0.10

NP → ears, 0.18　　NP → saw, 0.04

NP → stars, 0.18　　NP → telescopes, 0.1

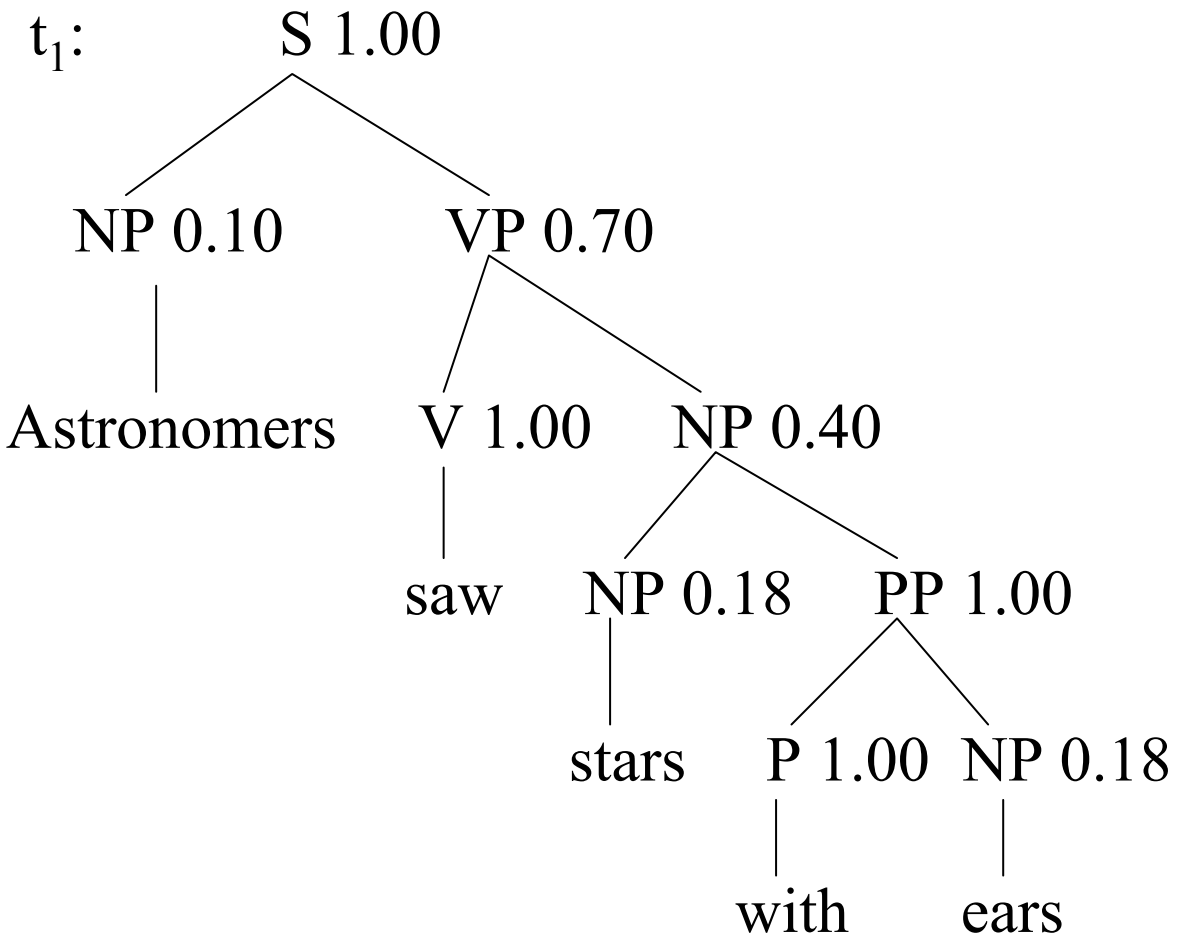
PP → P NP, 1.00　　P → with, 1.00

VP → V NP, 0.70　　VP → VP PP, 0.30

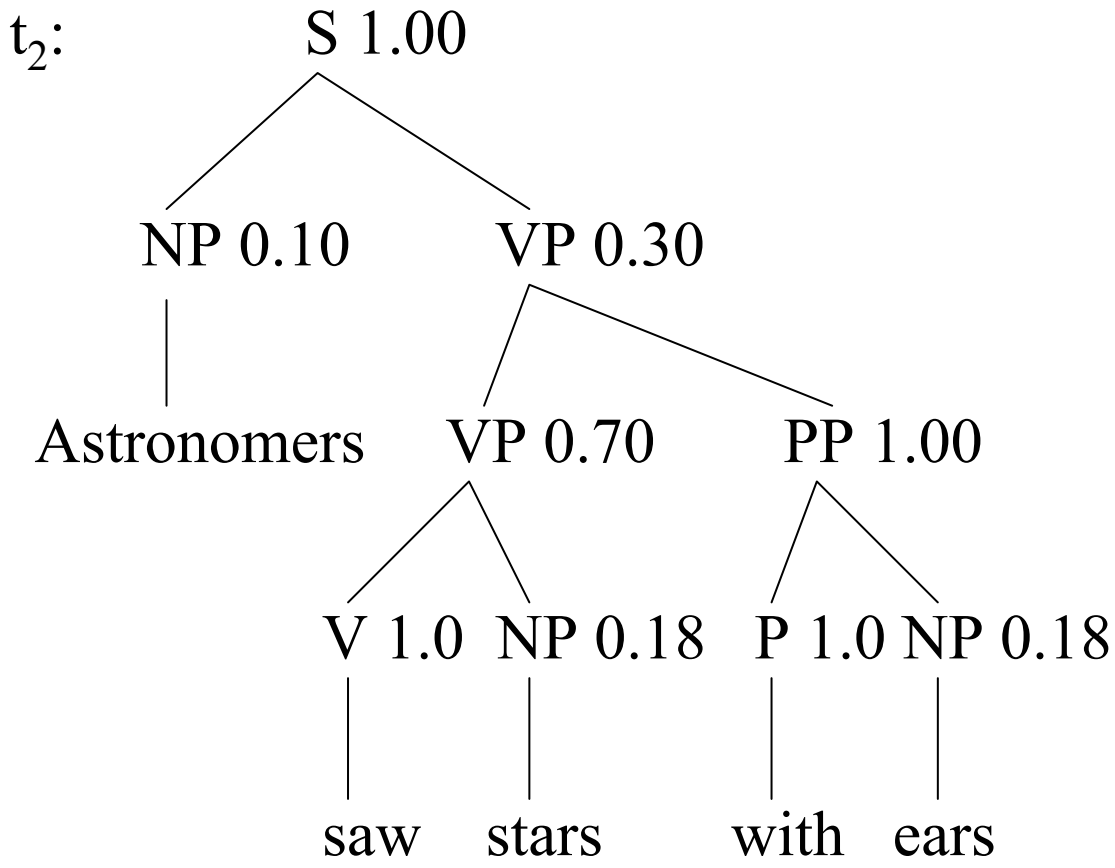
V → saw, 1.00

给定句子 S：***Astronomers saw stars with ears.***

8.4 概率上下文无关文法



8.4 概率上下文无关文法



8.4 概率上下文无关文法

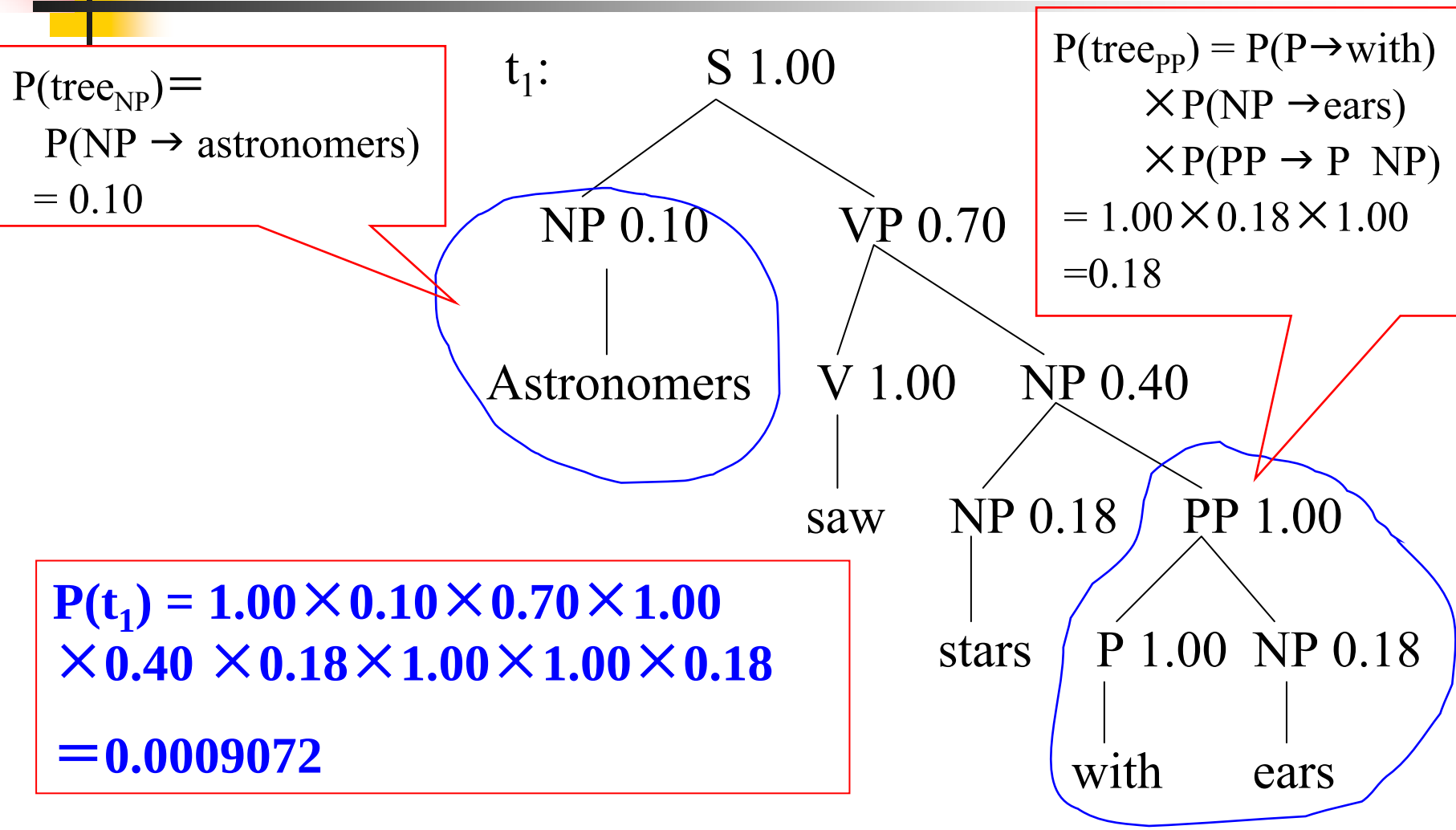
- 计算分析树概率的基本假设
  - 位置不变性：子树的概率与其管辖的词在整个句子中所处的位置无关，即对于任意的 k,

    $P(A_{k(k+C)} \rightarrow w)$ 一样。
  - 上下文无关性：子树的概率与子树管辖范围以外的词无关，即 $P(A_{kl} \rightarrow w | \textit{任何超出k~l范围的上下文}) = P(A_{kl} \rightarrow w)$

8.4 概率上下文无关文法

祖先无关性：子树的概率与推导出该子树的祖先结点无关，即 $P(A_{kl} \rightarrow w \mid$ *任何除 A 以外的祖先结点*$) = P(A_{kl} \rightarrow w)$

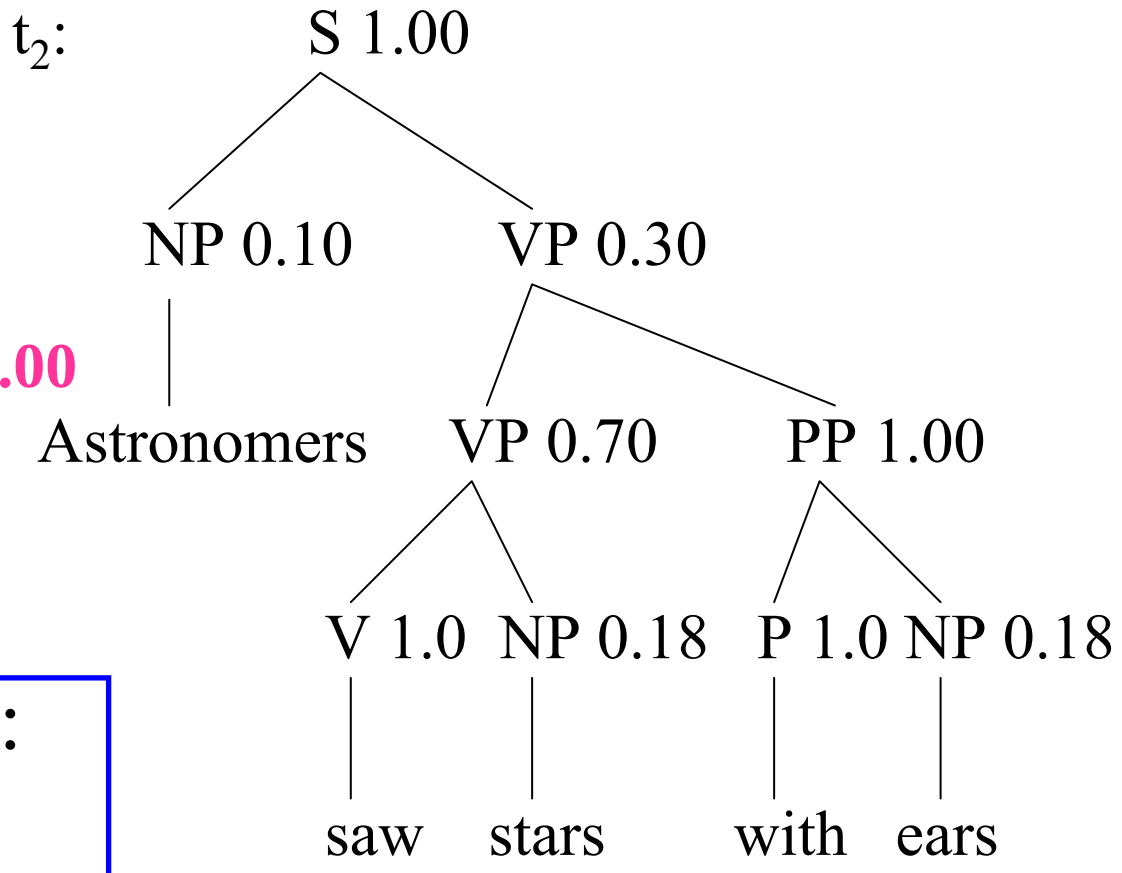
8.4 概率上下文无关文法



$P(t_1) = 1.00 \times 0.10 \times 0.70 \times 1.00 \times 0.40 \times 0.18 \times 1.00 \times 1.00 \times 0.18$

$= 0.0009072$

8.4 概率上下文无关文法



$P(t_2)=1.00\times0.10\times0.30\times0.70\times1.00\times0.18\times1.00\times1.00\times0.18$

$=0.0006804$

给定的句子 s：

$P(t_1) > P(t_2)$

8.4 概率上下文无关文法

❑ 运用PCFG的三个基本问题

(1) 给定句子 $W = w_1 w_2 \cdots w_n$ 和 PCFG G，如何快速计算 $P(W|G)$ ？

(2) 给定句子 $W = w_1 w_2 \cdots w_n$ 和 PCFG G，如何快速地选择最佳句法结构树？

(3) 给定句子 $W = w_1 w_2 \cdots w_n$ 和 PCFG G，如何调节 G 的参数，使得 $P(W|G)$ 最大？

8.4 概率上下文无关文法

假设文法 G(S) 的规则只有两种形式：

$$A \rightarrow a, \quad a \in V_T$$

$$A \rightarrow BC, \quad B, C \in V_N$$

可以通过范式化处理，使 CFG 规则满足上述形式。

8.4 概率上下文无关文法

◆ 向内算法－解决第一个问题：计算句子的概率

基本思想：通过动态规划计算由非终结符 A 推导出的字串 $w_i w_{i+1} \cdots w_j$ 的概率 $\alpha_{ij}(A)$。语句 $W = w_1 w_2 \cdots w_n$ 的概率即为文法 $G(S)$中 S 推导出的字串的概率 $\alpha_{1n}(S)$ 。

8.4 概率上下文无关文法

- 定义：

向内变量 $\alpha_{ij}(A)$ 是由非终结符 A 推导出的语句 W 中子字串 $w_i w_{i+1} \cdots w_j$ 的概率：

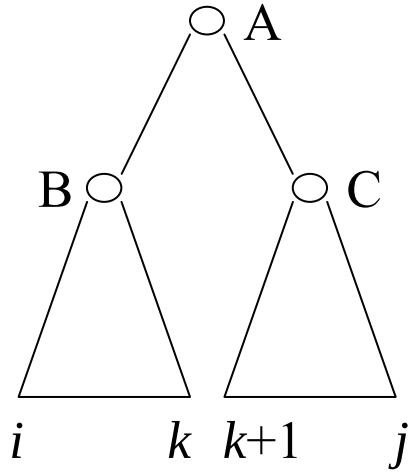
$$\alpha_{ij}(A) = P(A \overset{*}{\Rightarrow} w_i w_{i+1} \cdots w_j)$$

8.4 概率上下文无关文法

计算 $\alpha_{ij}(A)$ 的递推公式：

$$\alpha_{ii}(A) = P(A \to w_i)$$

$$\alpha_{ij}(A) = \sum_{B,C \in V_N} \sum_{i \le k \le j} P(A \to BC)\alpha_{ik}(B)\alpha_{(k+1)j}(C)$$



$A \to BC$

$B \to w_i w_{i+1} \cdots w_k$

$C \to w_{k+1} \cdots w_j$

8.4 概率上下文无关文法

- 向内算法描述

输入：文法 G(S)，语句 $W = w_1, w_2, \cdots, w_n$

输出：$P(S \overset{*}{\Rightarrow} w_1 w_2 \cdots w_n)$

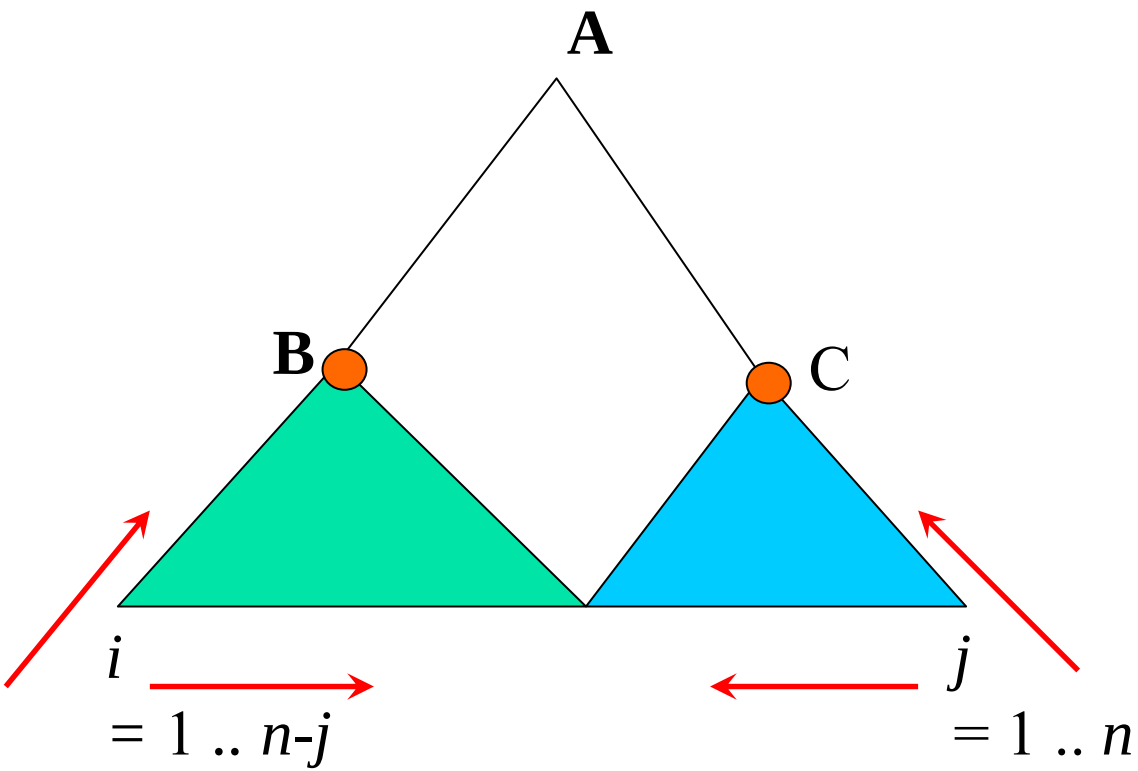
(1) 初始化： $\alpha_{ii}(A) = P(A \to w_i)$ $A \in V_N$， $1 \le i \le j \le n$

(2) 归纳计算： j=1.. n, i=1..n- j, 重复下列计算：

$$\alpha_{i(i+j)}(A) = \sum_{B,C \in V_N} \sum_{i \le k \le i+j} P(A \to BC)\alpha_{ik}(B)\alpha_{(k+1)(i+j)}(C)$$

(3) 终结：$P(S \overset{*}{\Rightarrow} w_1 w_2 \cdots w_n) = \alpha_{1n}(S)$

8.4 概率上下文无关文法



向内归纳的过程

8.4 概率上下文无关文法

◆ Viterbi 算法－解决第二个问题：

最佳分析结果搜索

➢ 定义：

Viterbi 变量 $\gamma_{ij}(A)$ 是由非终结符 A 推导出语句 W 中子字串 $w_i w_{i+1} \cdots w_j$ 的最大概率。

变量 $\psi_{i,j}$ 用于记忆字串 $W = w_1 w_2 \cdots w_n$ 的 Viterbi 语法分析结果。

8.4 概率上下文无关文法

- **Viterbi 搜索算法描述**

输入：文法 G(S)，语句 $W = w_1 w_2 \cdots w_n$

输出： $\gamma_{1n}(S)$

(1) 初始化： $\gamma_{ii}(A) = P(A \to w_i)$ $A \in V_N$， $1 \le i \le j \le n$

(2) 归纳计算： j=1..n, i=1..n- j, 重复下列计算：

$$\gamma_{i(i+j)}(A) = \max_{B,C \in V_N; i \le k \le i+j} P(A \to BC)\gamma_{ik}(B)\gamma_{(k+1)(i+j)}(C)$$

$$\psi_{i(i+j)}(A) = \max_{B,C \in V_N; i \le k \le i+j} P(A \to BC)\gamma_{ik}(B)\gamma_{(k+1)(i+j)}(C)$$

(3) 终结： $P(S \overset{*}{\Rightarrow} w_1 w_2 \cdots w_n) = \gamma_{1n}(S)$

8.4 概率上下文无关文法

◆ 向内向外算法－解决第三个问题：参数估计

➢ 参数估计的基本思路

已知训练语料中语法结构，记录每个语法规则的使用次数，用最大似然估计计算 PCFG 的参数，即：

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

利用大量标注语料，使用 EM (Expectation Maximization) 方法估计参数。

8.4 概率上下文无关文法

- 定义：

向外变量 $\beta_{ij}(A)$ 是由文法初始符号 S 推导出语句 $W = w_1 w_2 \cdots w_n$ 的过程中，到达扩展符号串 $w_1 \cdots w_{i-1} A w_{j+1} \cdots w_n$ 的概率：

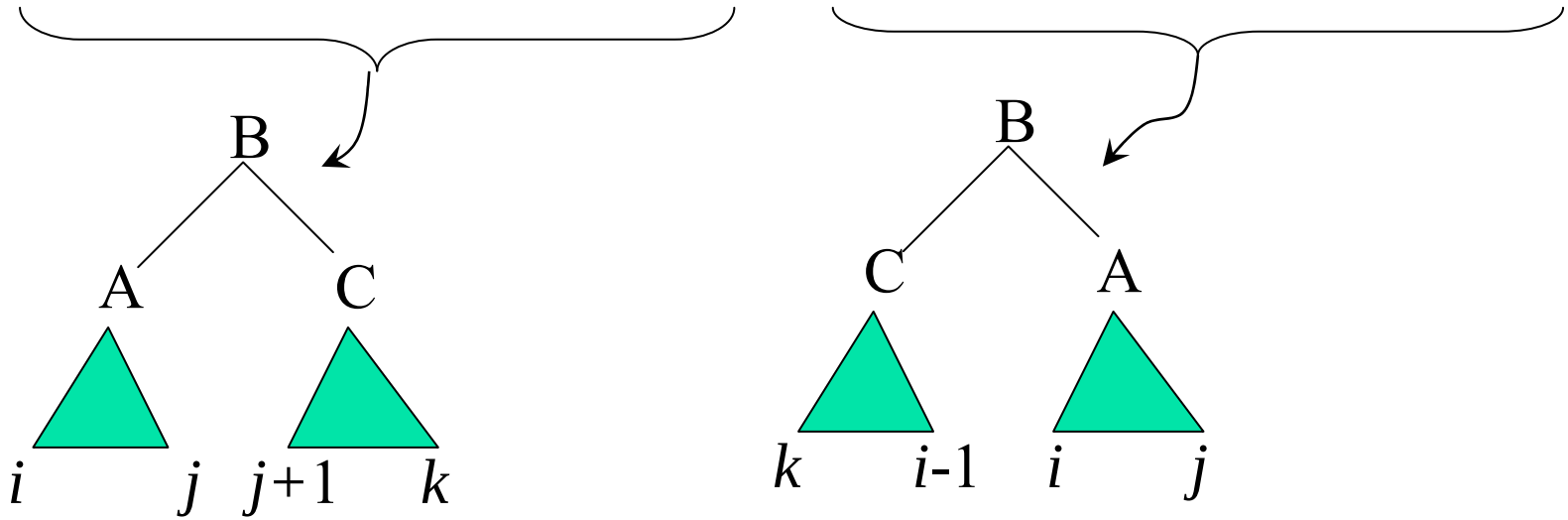
$$\beta_{ij}(A) = P(S \overset{*}{\Rightarrow} w_1 \cdots w_{i-1} A w_{j+1} \cdots w_n)$$

8.4 概率上下文无关文法

$\beta_{ij}(A)$可由动态规划算法计算求得，其递推公式：

$$\beta_{1n}(A) = \delta(A, S)$$ （初始化）

$$\beta_{ij}(A) = \sum_{B,C}\sum_{k>j} P(B \to AC)\alpha_{j+1,k}(C)\beta_{ik}(B) + \sum_{B,C}\sum_{k<i} P(B \to CA)\alpha_{k,i-1}(C)\beta_{kj}(B)$$



8.4 概率上下文无关文法

➢ 几种可能：

（1）当 $i=1, j=n$ 时，即 $w_i w_{i+1} \cdots w_j$ 是整个语句 W 时，由于语法中不可能有规则 $S \rightarrow A$，因此，S 推导出 W 的过程中，如果 $A \neq S$，那么，A 推导出 W 的概率为 0 （$\beta_{1n}(A)$）。

如果 $A=S$，$\beta_{1n}(A)$ 为 S 直接推导出 W 的概率，即为1。

8.4 概率上下文无关文法

（2）当 $i \neq 1$ 或者 $j \neq n$ 时，如果在 S 推导出 W 的过程中出现了字符串 $w_1 \cdots w_k A w_{j+1} \cdots w_n$，则该推导过程必定使用了规则$B \rightarrow A\ C$ 或者 $B \rightarrow C\ A$。假定运用了规则 $B \rightarrow A\ C$ 推导出 $w_i \cdots w_j w_{j+1} \cdots w_k$，则该推导可以分解为以下三种情况：

8.4 概率上下文无关文法

(*i*) 由 S 推导出 $w_1 \cdots w_{i-1} B w_{k+1} \cdots w_n$，其概率为 $\beta_{ik}(B)$

(*ii*) 运用产生式 $B \rightarrow AC$ 扩展非终结符B，其概率为 $P(B \rightarrow AC)$

(*iii*)由非终结符 C 推导出 $w_{j+1} \cdots w_k$，其概率为 $\alpha_{j+1,k}(C)$。

由于B, C, 和k 的任意性，在计算$\beta_{ik}(B)$ 时，必须考虑所有可能的B, C 和k，同时还必须考虑相似的由产生式 $B \rightarrow CA$ 导出 A 的情况。

8.4 概率上下文无关文法

➢ 向外算法：

输入：PCFG G=(S, N, T, P)，语句$W = w_1 w_2 \cdots w_n$

输出：$\beta_{ij}(A)$, $A \in N$, $1 \le i \le j \le n$

(1) 初始化：$\beta_{1n}(A) = \delta(A, S)$, $A \in N$

(2) 归纳计算：j 从n-1 到0，i 从1 到 n-j，重复计算：

$$\beta_{i(i+j)}(A) = \sum_{B,C} \sum_{i+j<k\le n} P(B \to AC)\alpha_{(i+j+1)k}(C)\beta_{ik}(B)$$

$$+ \sum_{B,C} \sum_{1\le k<i} P(B \to CA)\alpha_{k(i-1)}(C)\beta_{k(i+j)}(B)$$

8.4 概率上下文无关文法

- 利用向内、向外算法估计 PCFG 的参数

初始时随机地给这些参数赋值，得到语法 G_0，依据 G_0 和训练语料，得到语法规则使用次数的期望值，以期望次数运用于最大似然估计，得到语法参数的新的估计，由此得到新的语法 G_1，由 G_1 再次得到语法规则的使用次数的期望值，然后又可以重新估计语法参数。循环这个过程，语法参数将收敛于最大似然估计值。

8.4 概率上下文无关文法

给定 CFG G 和训练数据 $W = w_1 w_2 \cdots w_n$，语法规则 $A \to BC$ 的使用次数的期望值为：

$$C(A \to BC) = \sum_{1 \le i \le k \le j \le n} P(A_{ij}, B_{ik}, C_{(k+1)j} \mid w_1 \cdots w_n, G)$$

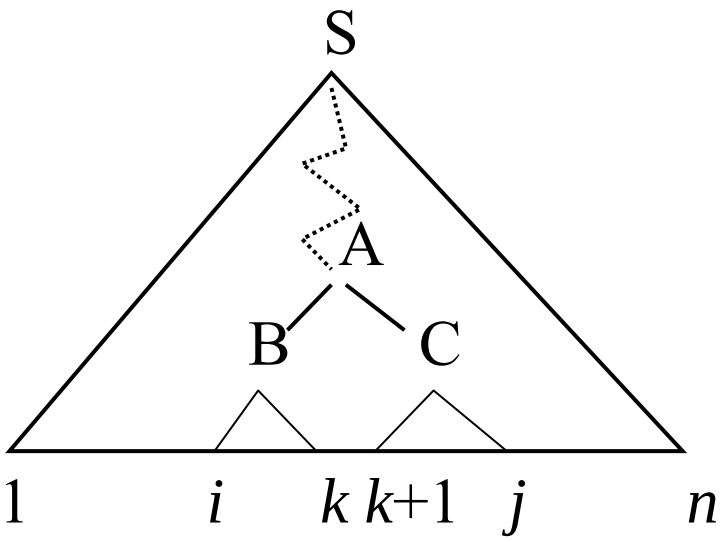
$$= \frac{1}{P(w_1 \cdots w_n \mid G)} \sum_{1 \le i \le k \le j \le n} P(A_{ij}, B_{ik}, C_{(k+1)j}, w_1 \cdots w_n \mid G)$$

$$= \frac{1}{P(w_1 \cdots w_n \mid G)} \sum_{1 \le i \le k \le j \le n} \beta_{ij}(A) P(A \to BC) \alpha_{ik}(B) \alpha_{(k+1)j}(C)$$

… (8-1)

8.4 概率上下文无关文法

解释：给定了语句 $w_1 \cdots w_n$，PCFG G 中产生式 $A \rightarrow BC$ 被用于产生 $w_1 \cdots w_n$ 的使用次数的期望值为：在所有可能的$1 \le i \le k \le j \le n$ 的情况下，$w_1 \cdots w_n$的语法分析结构中 $w_i \cdots w_k$ 由 B 导出，$w_{k+1} \cdots w_j$由 C 导出，$w_i \cdots w_j$ 由 A 导出的概率总和。



8.4 概率上下文无关文法

类似地，语法规则 $A \to a$ 的使用次数的期望值为：

$$C(A \to a) = \sum_{1 \le i \le n} P(A_{ii} \mid w_1 \cdots w_n, G)$$

$$= \frac{1}{P(w_1 \cdots w_n \mid G)} \sum_{1 \le i \le n} P(A_{ii}, w_1 \cdots w_n \mid G)$$

$$= \frac{1}{P(w_1 \cdots w_n \mid G)} \sum_{1 \le i \le n} \beta_{ii}(A) P(A \to a) \delta(a, w_i) \quad \ldots (8\text{-}2)$$

8.4 概率上下文无关文法

G 的参数可由下面的公式重新估计：

$$\hat{P}(A \to \mu) = \frac{C(A \to \mu)}{\sum_{\mu} C(A \to \mu)} \qquad \ldots (8\text{-}3)$$

μ 要么为终结符号，要么为两个非终结符号串，即 $A \to \mu$ 为PCFG 要求的两种规则形式。

8.4 概率上下文无关文法

向内、向外算法：

Step-1: 初始化：随机地给 $P(A\to\mu)$ 赋值，使得 $\sum_{\mu}P(A\to\mu)=1$，由此得到语法 G_0。令 $i=0$；

Step-2: **EM步骤**：

E-步骤：由 G_i 根据公式(8-1)和(8-2)，计算期望值 $C(A\to BC)$和 $C(A\to a)$；

M-步骤：用 E-步骤所得的期望值，根据公式(8-3) 重新估计 $P(A\to\mu)$，得到 G_{i+1}。

8.4 概率上下文无关文法

Step-3: 循环计算：$i=i+1$，重复EM步骤，直至

$P(A\rightarrow\mu)$ 收敛。

8.4 概率上下文无关文法

❒ PCFG 的评价

- ◆ 优点：
  - ➢可利用概率减少分析过程的搜索空间；
  - ➢可利用概率对概率较小的子树剪枝，加快分析效率；
  - ➢可以定量地比较两个语法的性能。
- ◆ 弱点：
  - ➢无法统计词与词、词类与词类、短语与短语的同现信息。

本章小结

- 句法分析的任务，面临的困难
- 句法分析方法
  - 基于规则的方法：
    - Chart Parsing
    - CYK 方法
  - 基于概率上下文无关文法 PCFG
    - 快速计算分析树的概率（向内算法）
    - 快速计算最大概率分析树（Viterbi 算法）
    - 参数估计（向内向外算法）

习题

1. 编写程序实现自顶向下（top-down）的 Chart 分析器，体会自顶向下和自底向上（bottom-up）分析算法的不同。
2. 自学Left Corner 分析算法和 Tomita GLR 句法分析算法。
3. 如有条件，利用树库语料抽取 PCFG 规则，结合 Chart 分析算法实现一个基于 PCFG 的句法分析器。

Thanks

谢谢!