“自然语言处理导论”课程讲义

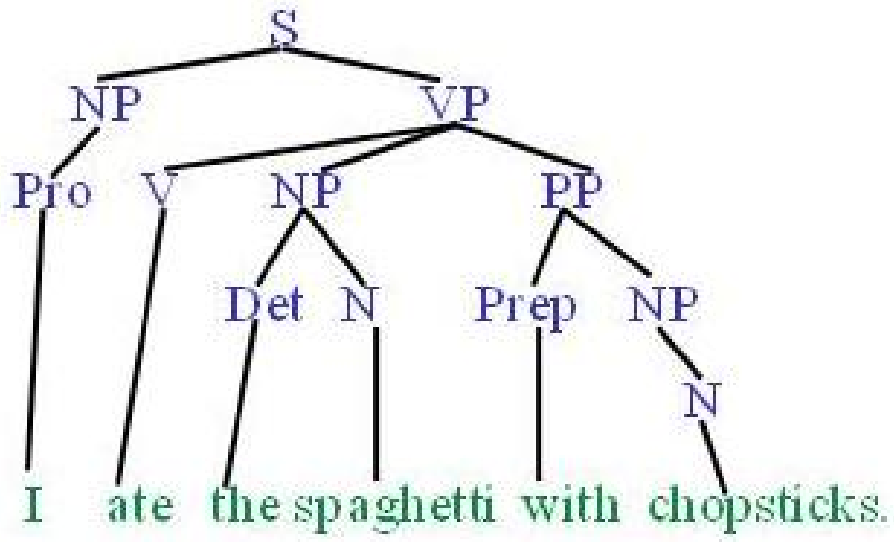
句法分析1：
上下文无关文法(Context Free Grammar)

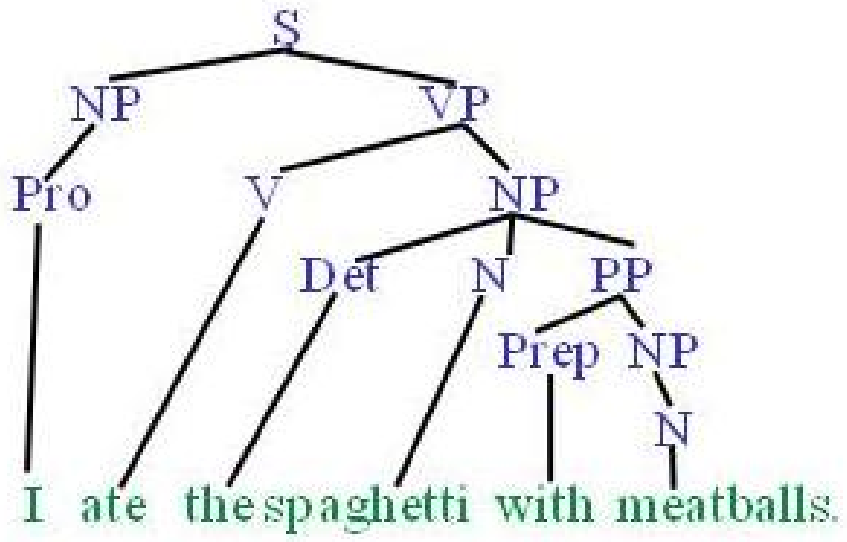
孙栩
信息科学技术学院
xusun@pku.edu.cn
http://klcl.pku.edu.cn/member/sunxu/index.htm

上下文无关文法(Context Free Grammar)

句法分析(Syntactic Parsing)

- 给定一个句子，输出其正确的句法结构树





上下文无关文法 (CFG)

- ***N*** 是非终结符集合
- **Σ** 是终结符集合
- ***R*** 是一个规则集合
  - 比如一个规则是A→β, 其中 A 是非终结符，而β可以是非终结符或终结符
- **S** 是一个特殊非终结符，代表树结构的根节点，称为 ***start symbol***

CFG举例

语法规则

S → NP VP
S → Aux NP VP
S → VP
NP → Pronoun
NP → Proper-Noun
NP → Det Nominal
Nominal → Noun
Nominal → Nominal Noun
Nominal → Nominal PP
VP → Verb
VP → Verb NP
VP → VP PP
PP → Prep NP

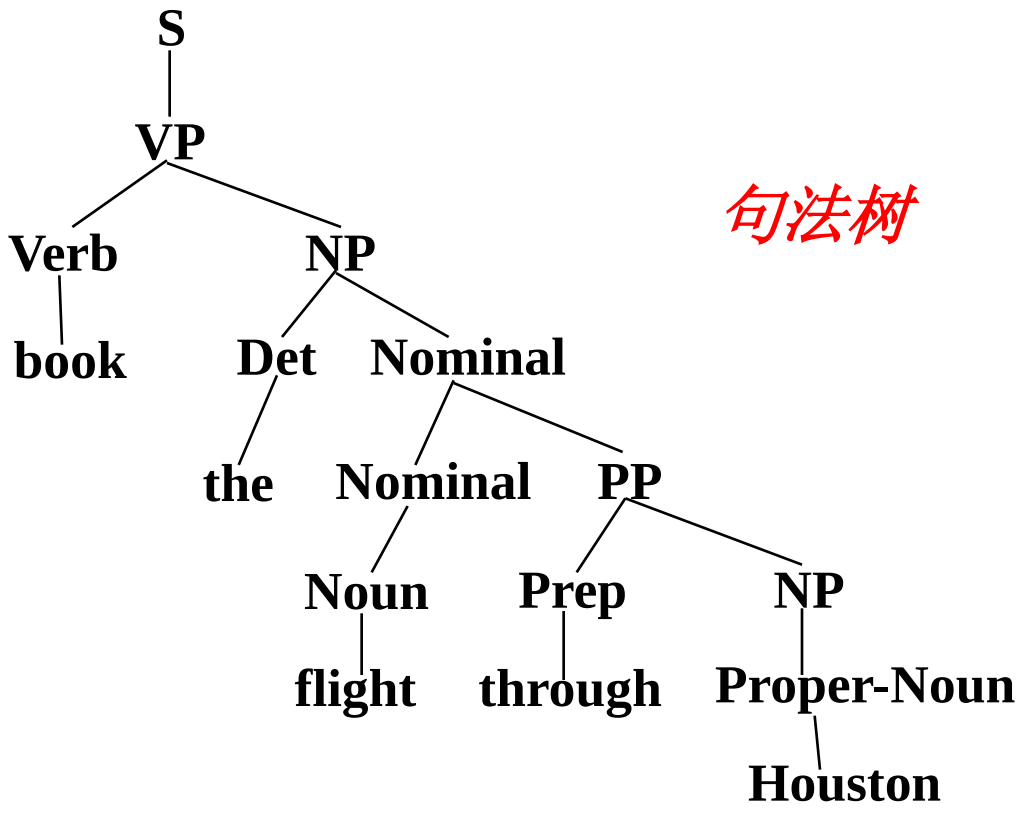
词汇规则

Det → the | a | that | this
Noun → book | flight | meal | money
Verb → book | include | prefer
Pronoun → I | he | she | me
Proper-Noun → Houston | NWA
Aux → does
Prep → from | to | on | near | through

从句法分析看句子是怎么生成的

- 从句法分析的角度来看，一个句子是从根节点开始递归生成句法规则的一个过程，直到最后只有终结符存在。

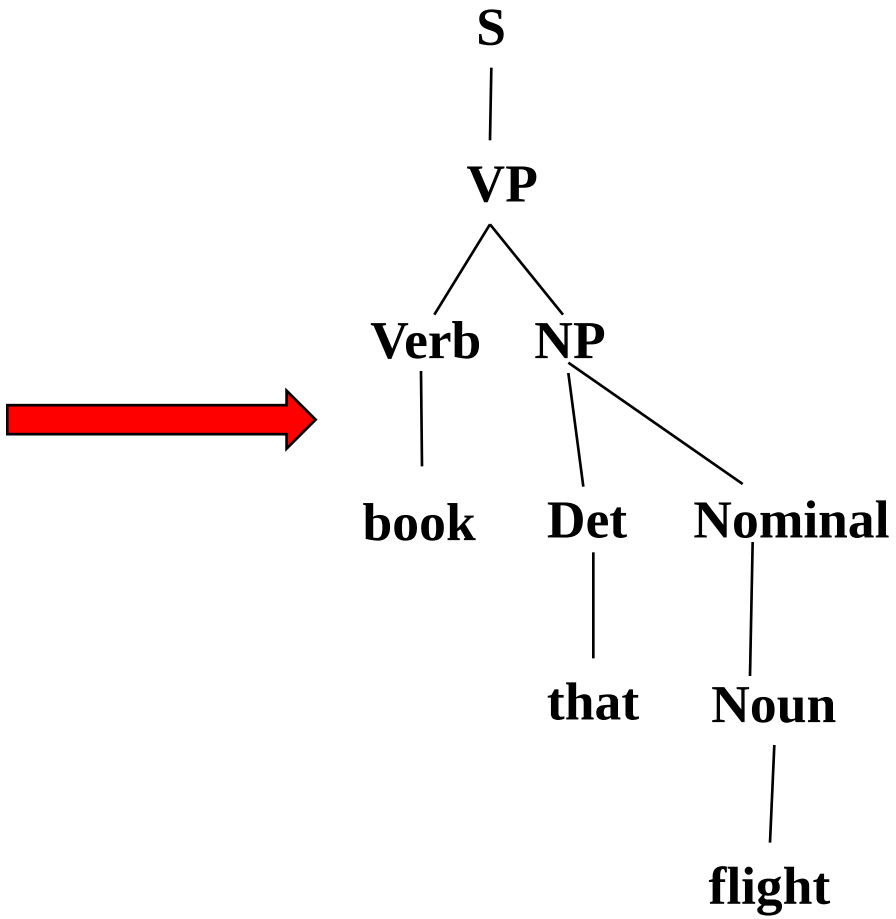
两大类方法：
自上而下句法分析
自下而上句法分析



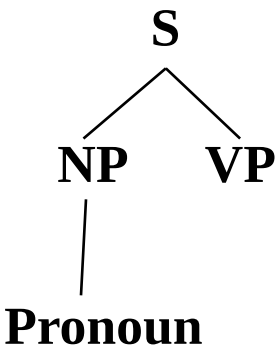
句法树

句法分析举例

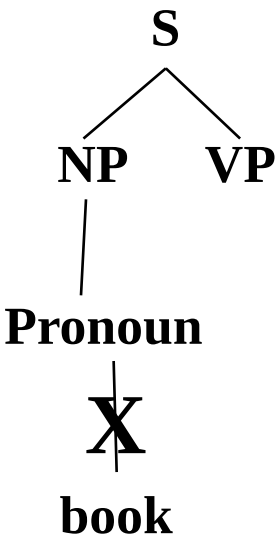
book that flight



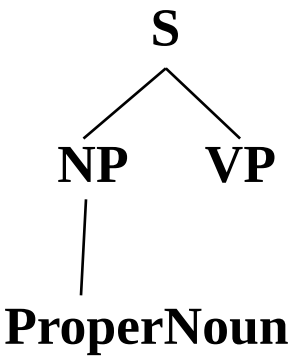
自上而下句法分析(Top Down Parsing)



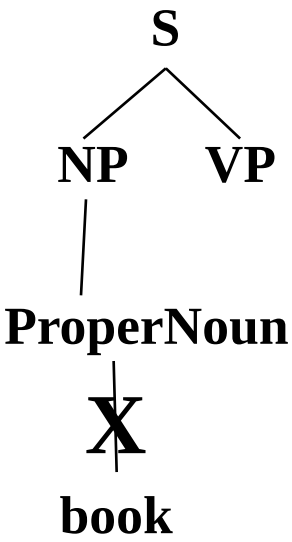
自上而下句法分析(Top Down Parsing)



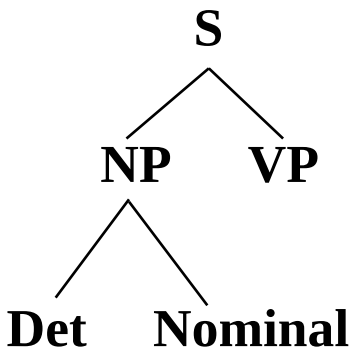
自上而下句法分析(Top Down Parsing)



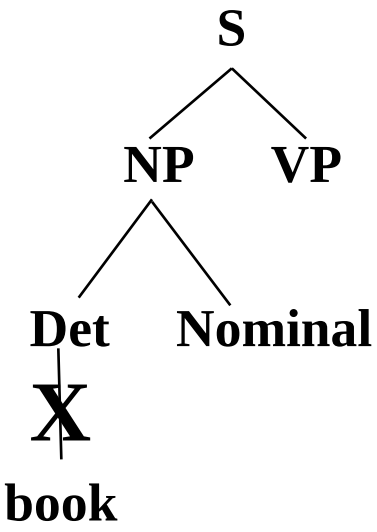
自上而下句法分析(Top Down Parsing)



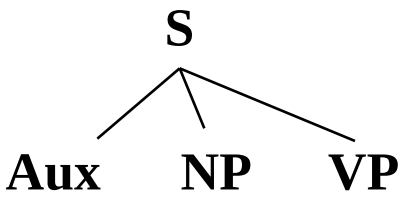
自上而下句法分析(Top Down Parsing)



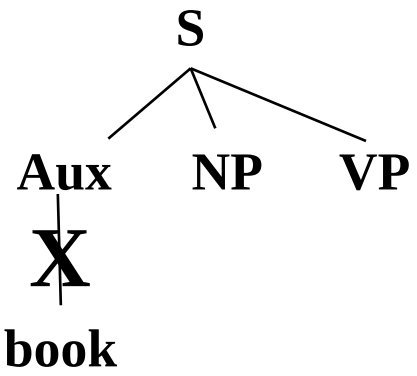
自上而下句法分析(Top Down Parsing)



自上而下句法分析(Top Down Parsing)

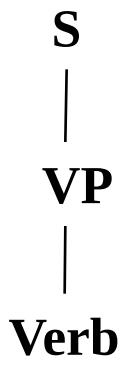


自上而下句法分析(Top Down Parsing)



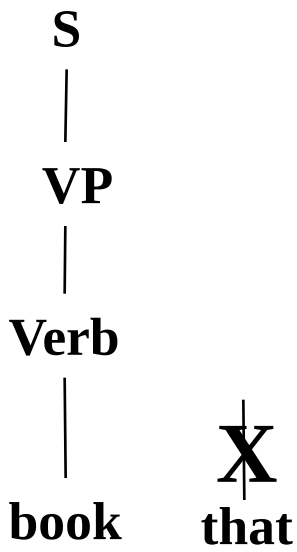
自上而下句法分析(Top Down Parsing)

自上而下句法分析(Top Down Parsing)

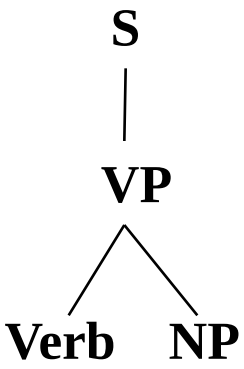


自上而下句法分析(Top Down Parsing)

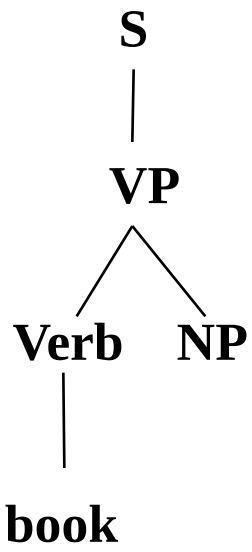
自上而下句法分析(Top Down Parsing)



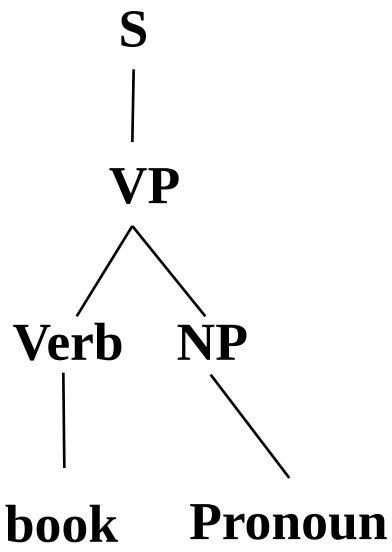
自上而下句法分析(Top Down Parsing)



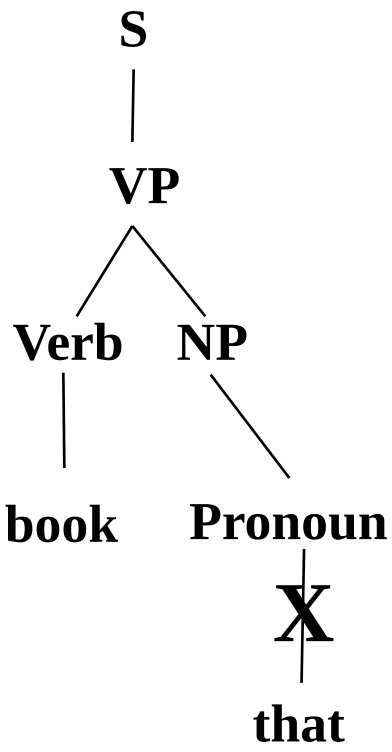
自上而下句法分析(Top Down Parsing)



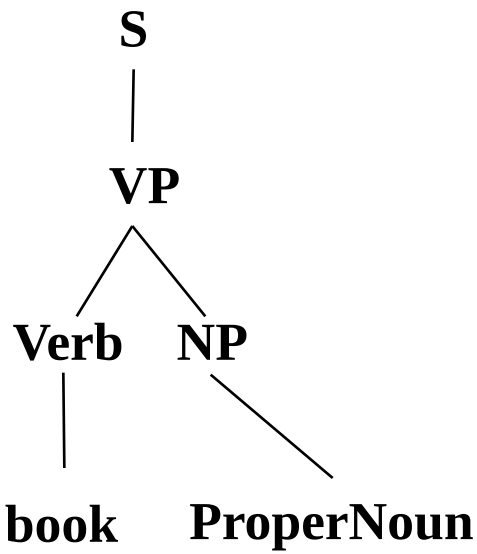
自上而下句法分析(Top Down Parsing)



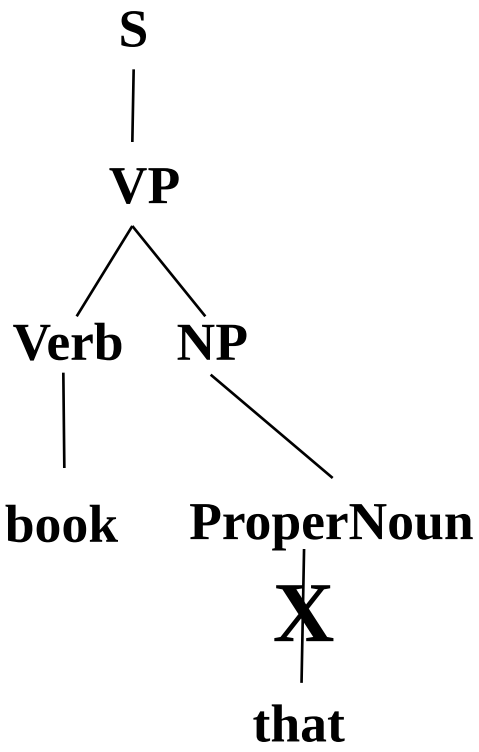
自上而下句法分析(Top Down Parsing)



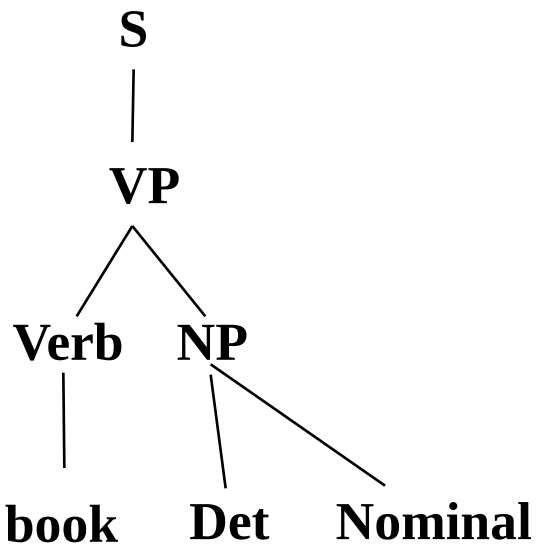
自上而下句法分析(Top Down Parsing)



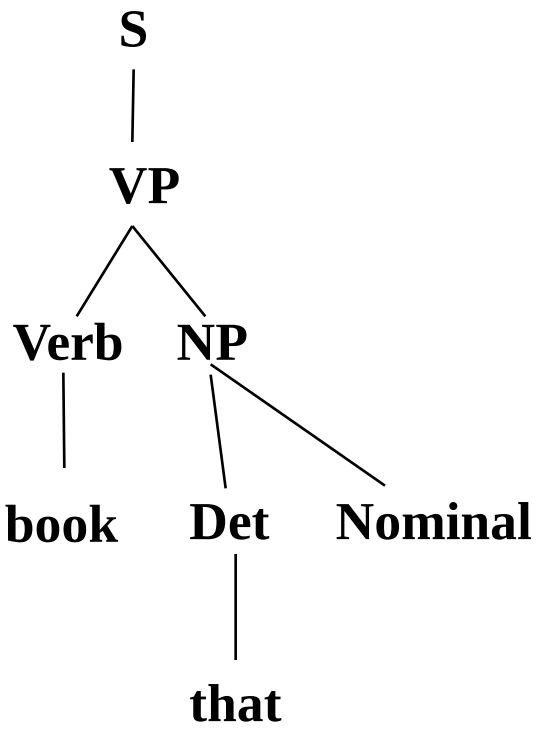
自上而下句法分析(Top Down Parsing)



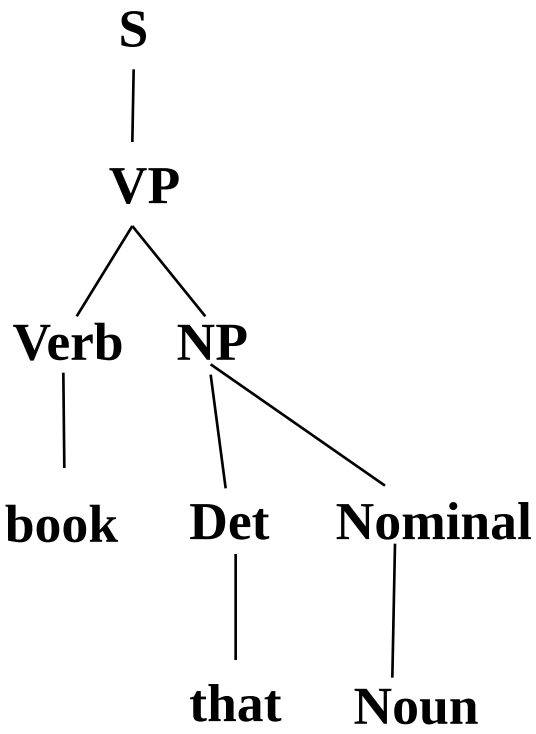
自上而下句法分析(Top Down Parsing)



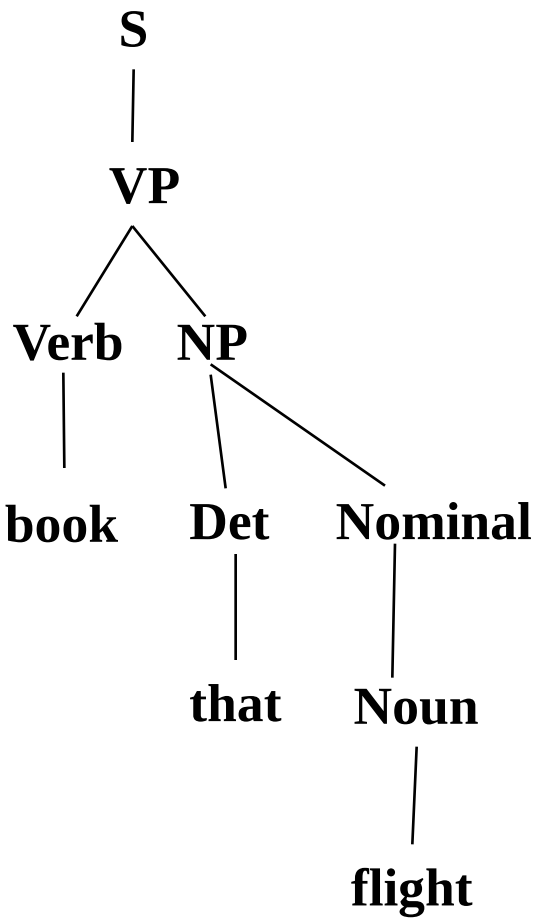
自上而下句法分析(Top Down Parsing)



自上而下句法分析(Top Down Parsing)



自上而下句法分析(Top Down Parsing)

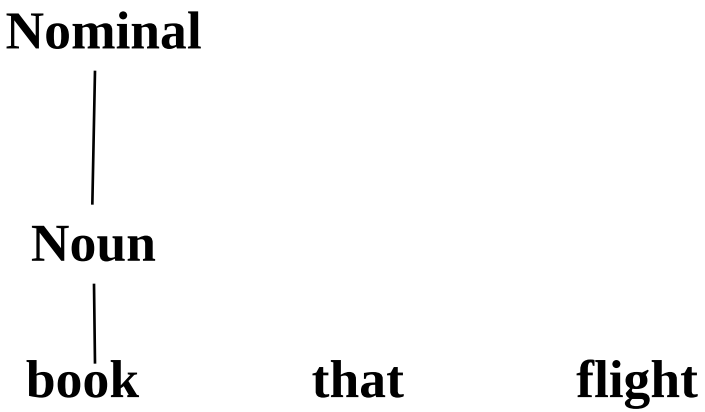


自下而上句法分析(Bottom Up Parsing)

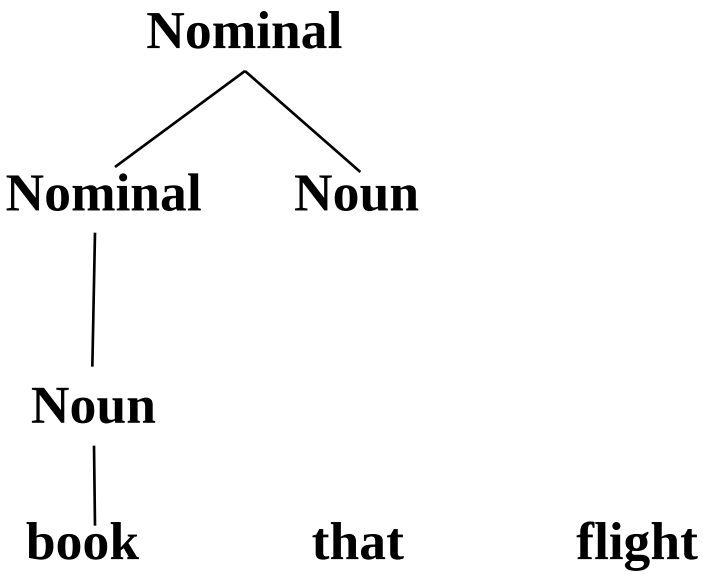
book **that** **flight**

自下而上句法分析(Bottom Up Parsing)

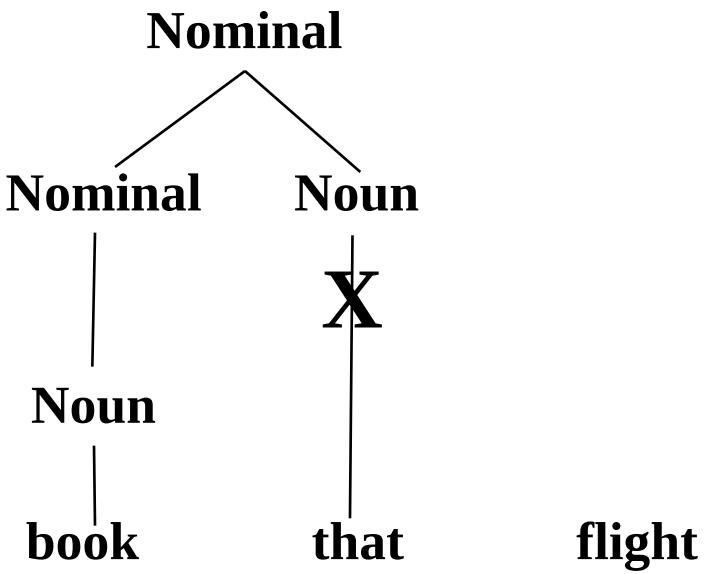
自下而上句法分析(Bottom Up Parsing)



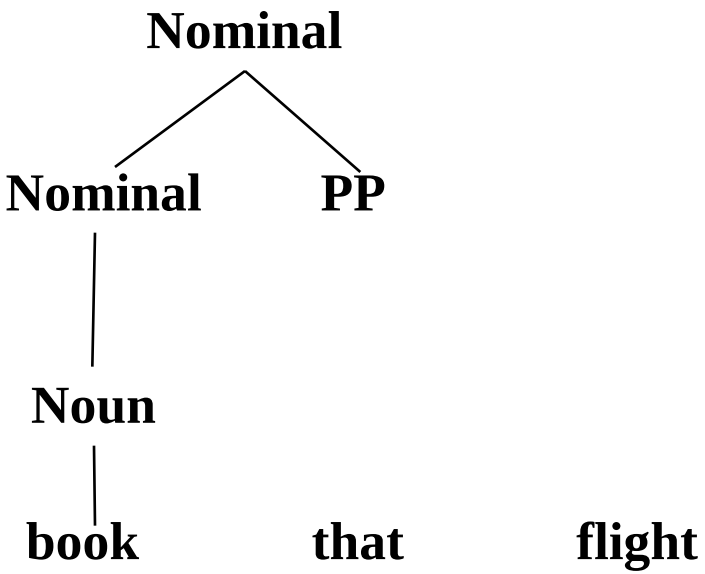
自下而上句法分析(Bottom Up Parsing)



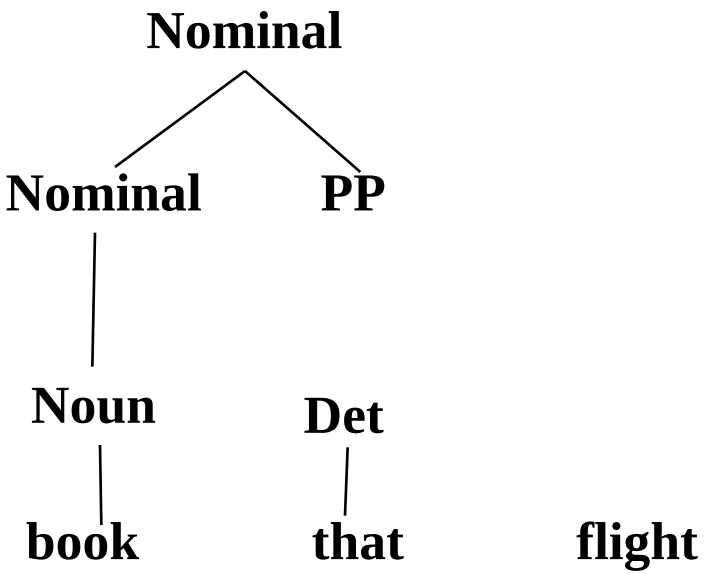
自下而上句法分析(Bottom Up Parsing)



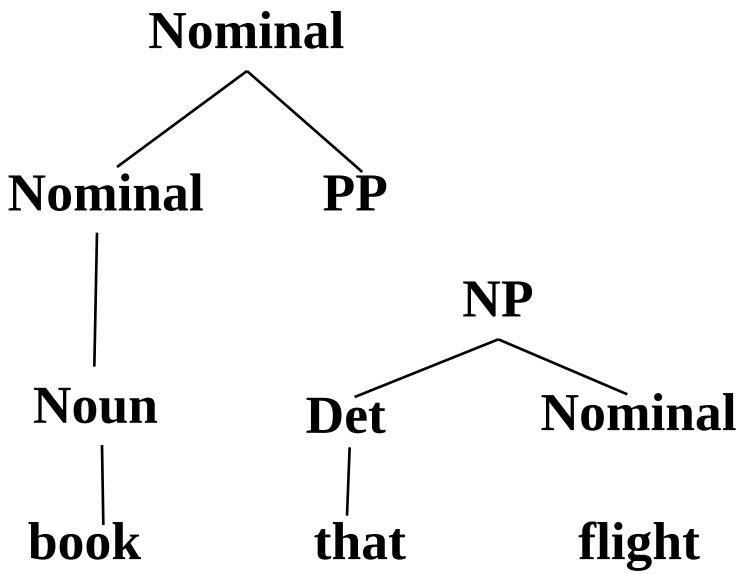
自下而上句法分析(Bottom Up Parsing)



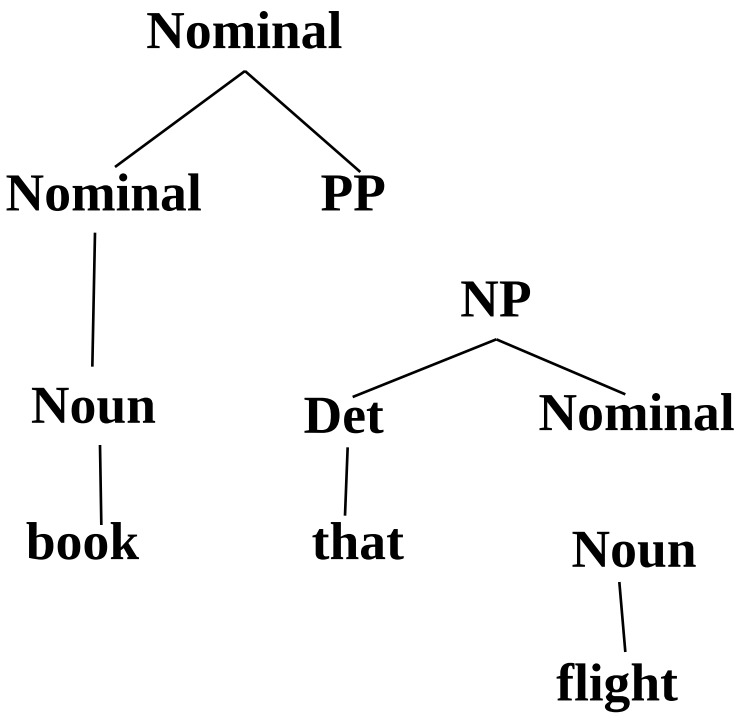
自下而上句法分析(Bottom Up Parsing)



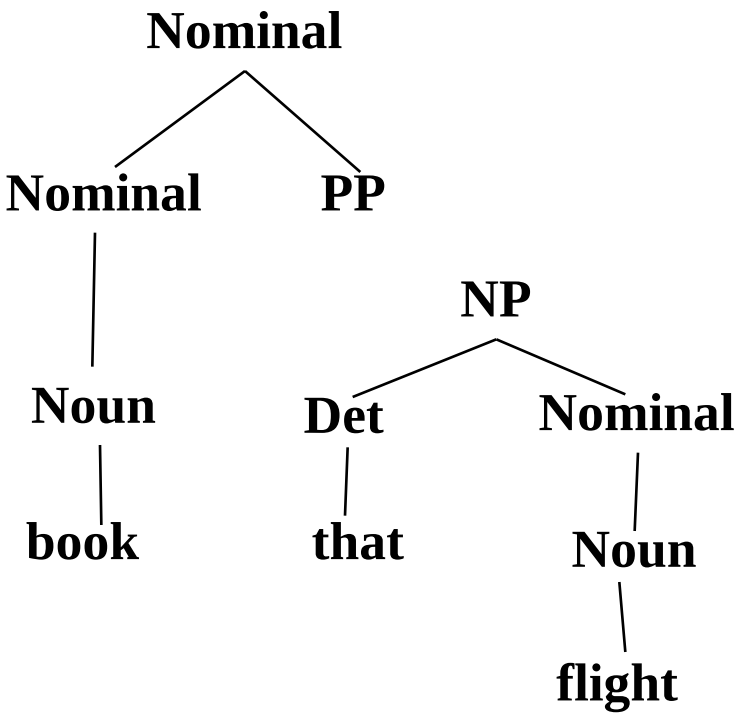
自下而上句法分析(Bottom Up Parsing)



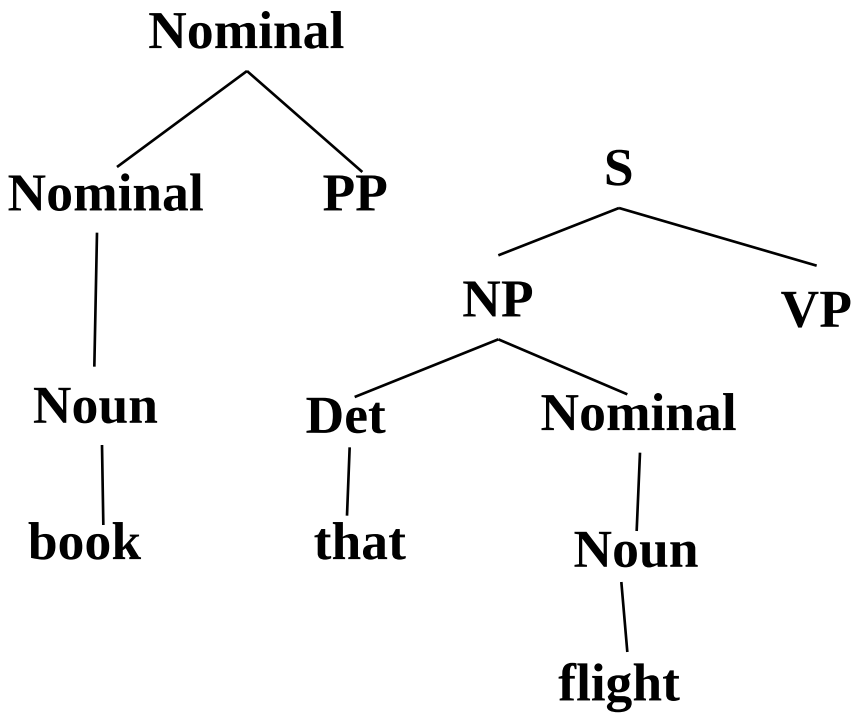
自下而上句法分析(Bottom Up Parsing)



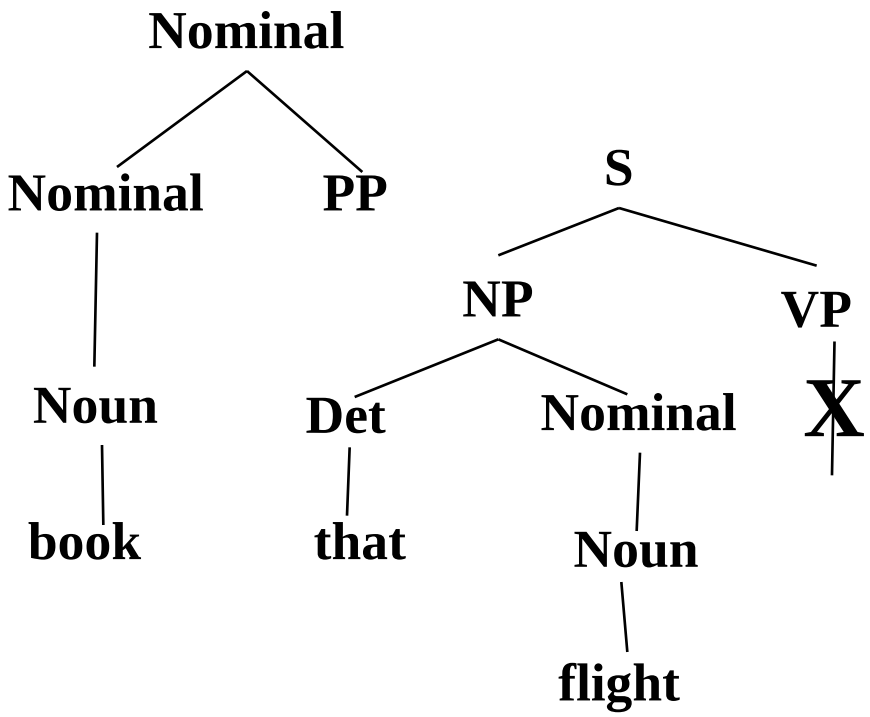
自下而上句法分析(Bottom Up Parsing)



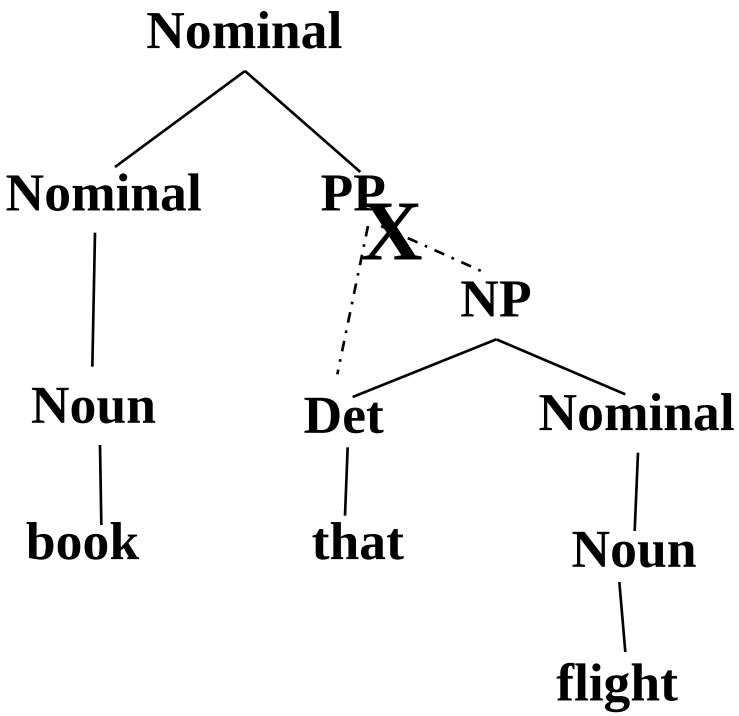
自下而上句法分析(Bottom Up Parsing)



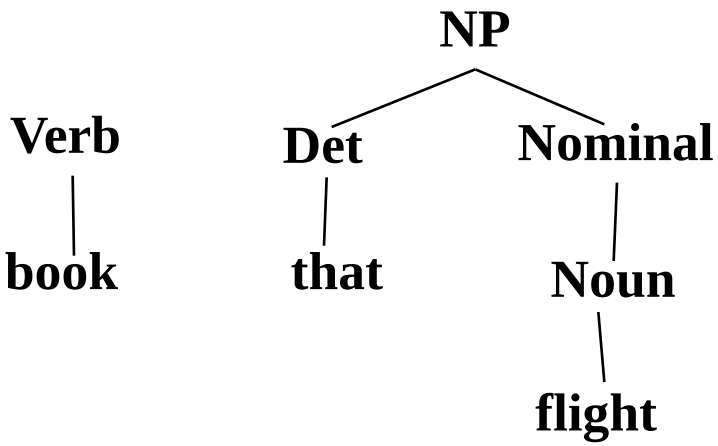
自下而上句法分析(Bottom Up Parsing)



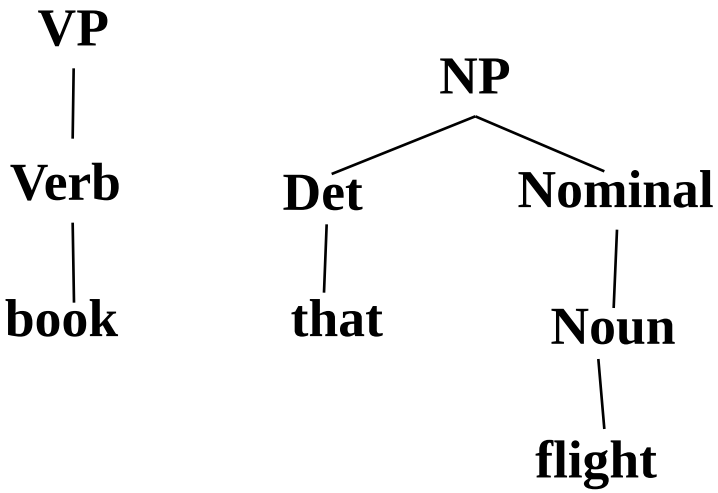
自下而上句法分析(Bottom Up Parsing)



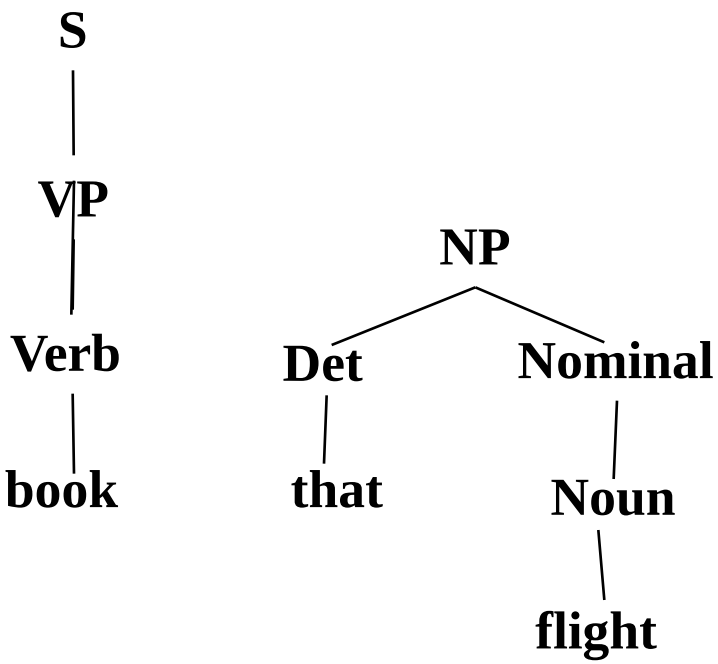
自下而上句法分析(Bottom Up Parsing)



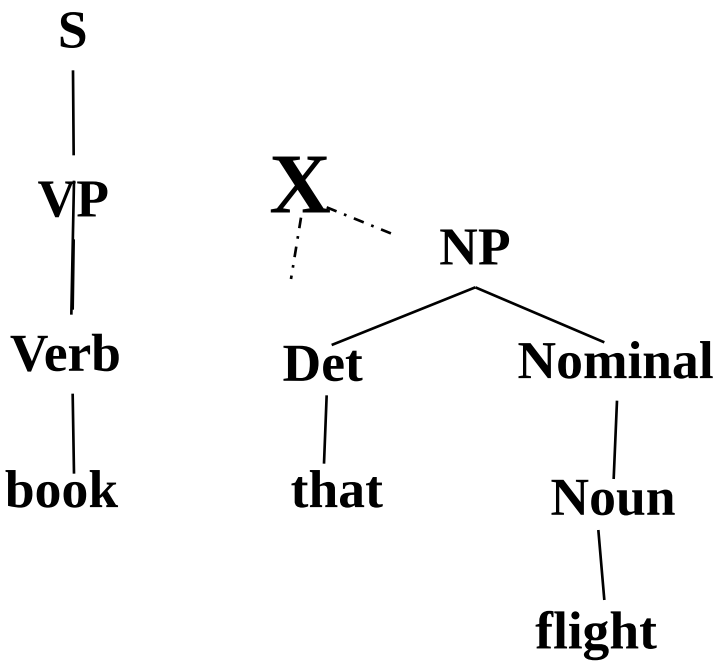
自下而上句法分析(Bottom Up Parsing)



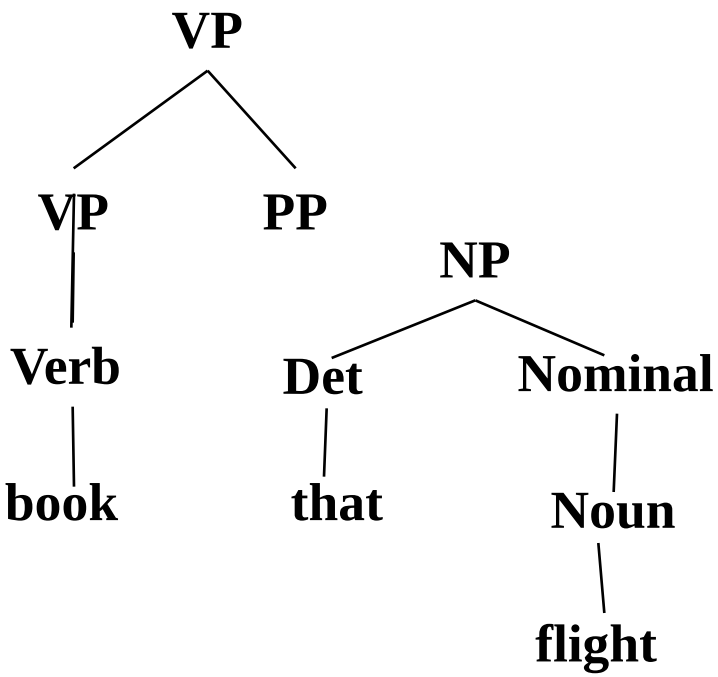
自下而上句法分析(Bottom Up Parsing)



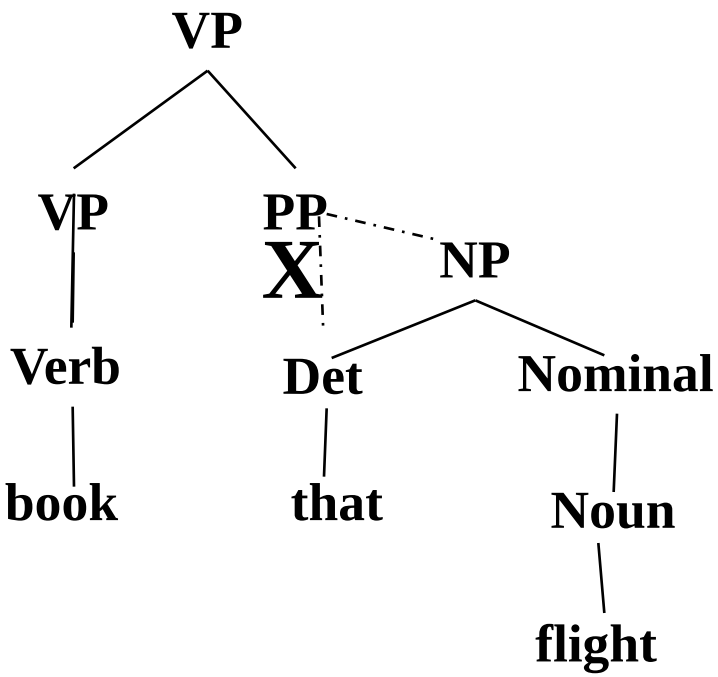
自下而上句法分析(Bottom Up Parsing)



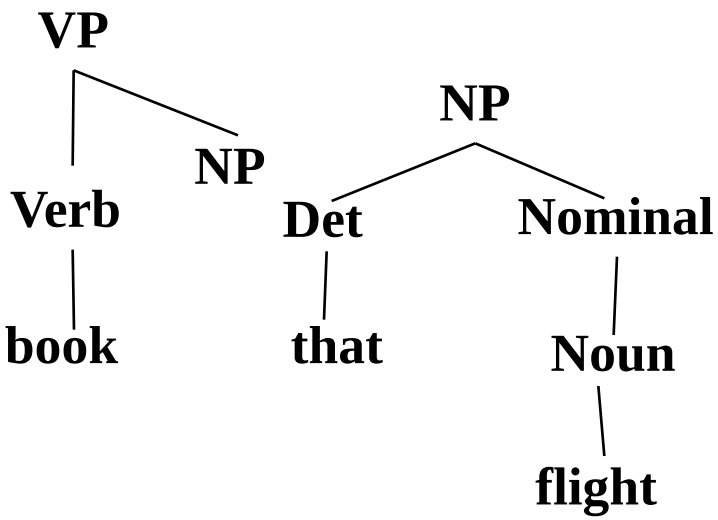
自下而上句法分析(Bottom Up Parsing)



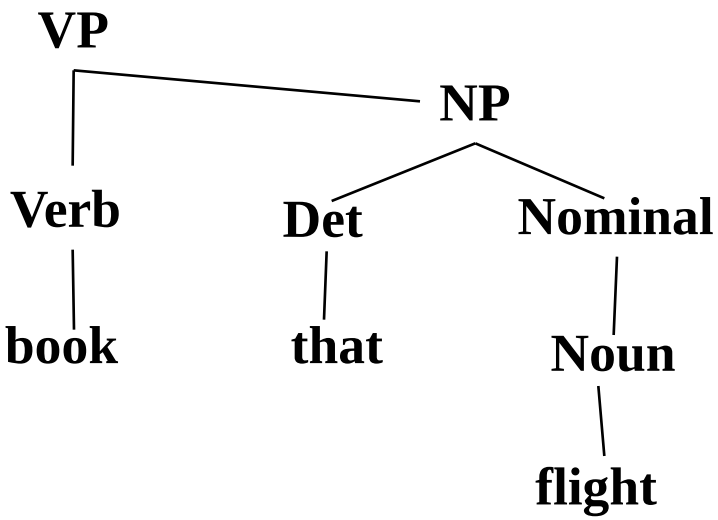
自下而上句法分析(Bottom Up Parsing)



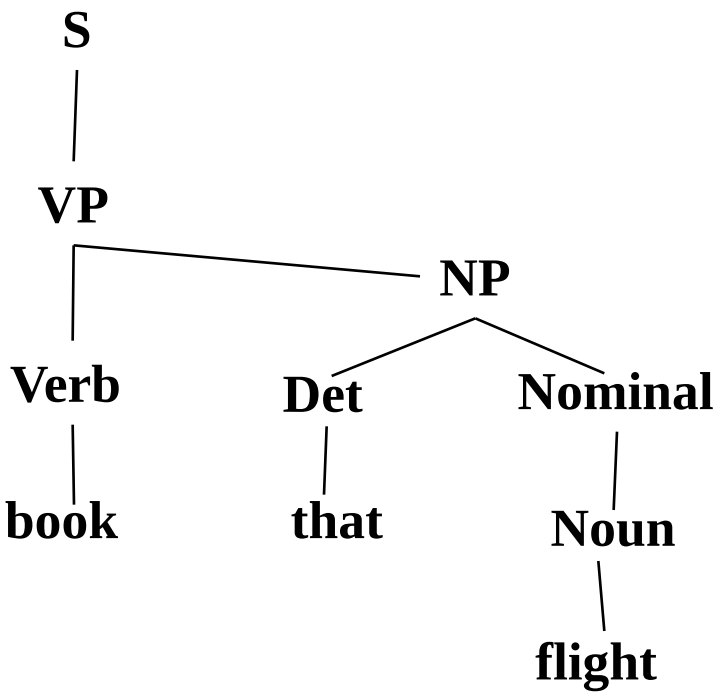
自下而上句法分析(Bottom Up Parsing)



自下而上句法分析(Bottom Up Parsing)



自下而上句法分析(Bottom Up Parsing)



自上而下 vs. 自下而上

- 自上而下方法的结果总是一个完整的句法树，但是有可能无法匹配实际的句子
- 自下而上方法总是能够匹配实际的句子，但是有可能不是一个完整的句法树
- 采用动态规划的方法，这两种方法都可以实现$O(n^3)$的句法分析复杂度，n是句子长度

动态规划的句法分析方法

- **CKY (Cocke-Kasami-Younger) 算法**是基于自底向上的动态规划句法分析方法
- **Earley算法**是基于自顶向下的句法分析方法
- **Chart算法**在图表(chart)里面保存了完整的短语信息，可以把自顶向下和自底向上的方法结合起来

CKY方法

- 乔姆斯基范式
  - 首先，CKY方法需要把不规则语法转换成**乔姆斯基范式 (Chomsky normal form, CNF)**
  - 在乔姆斯基范式里，一个生成规则必须生成2个非终结符，或者1个终结符

乔姆斯基范式转换

原始语法

S → NP VP
S → Aux NP VP

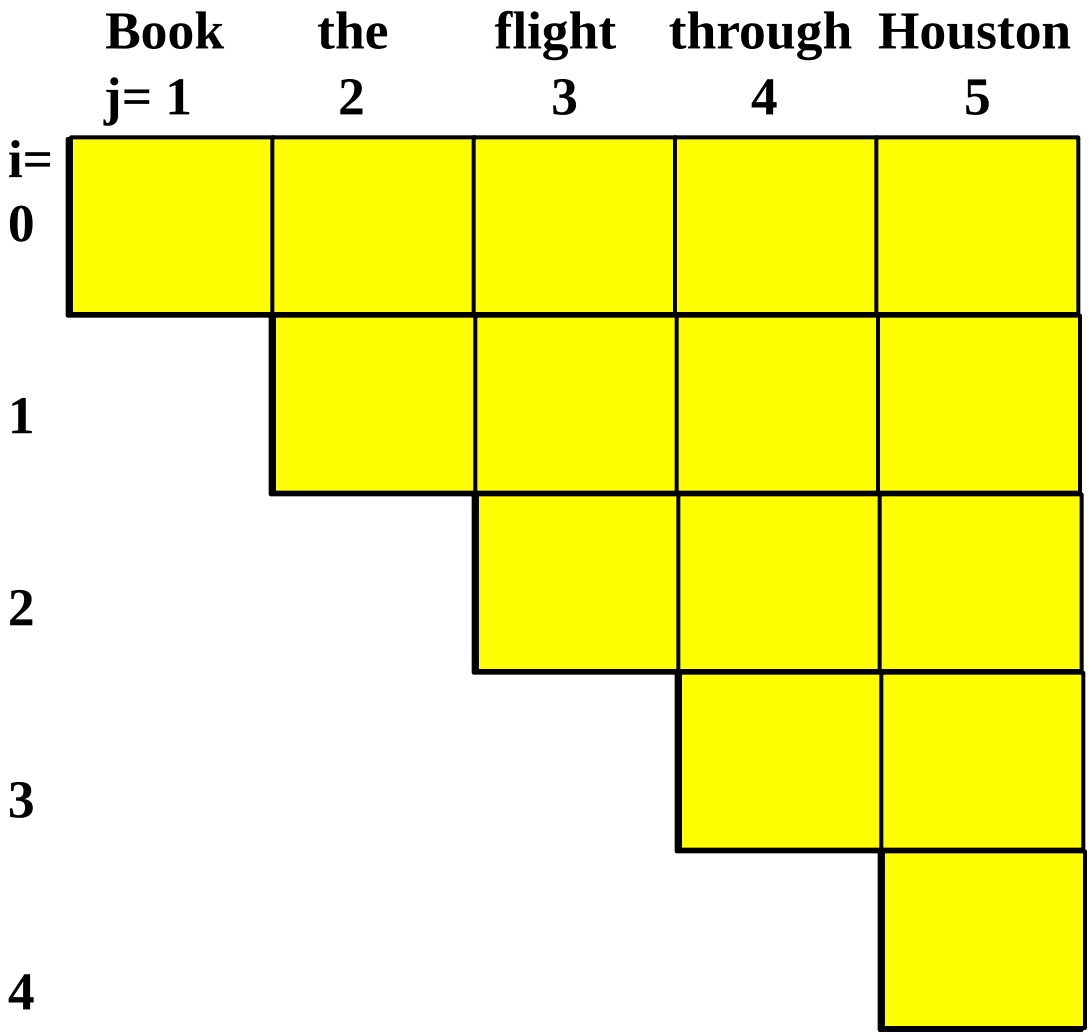
S → VP

NP → Pronoun
NP → Proper-Noun
NP → Det Nominal
Nominal → Noun
Nominal → Nominal Noun
Nominal → Nominal PP
VP → Verb
VP → Verb NP
VP → VP PP
PP → Prep NP

乔姆斯基范式

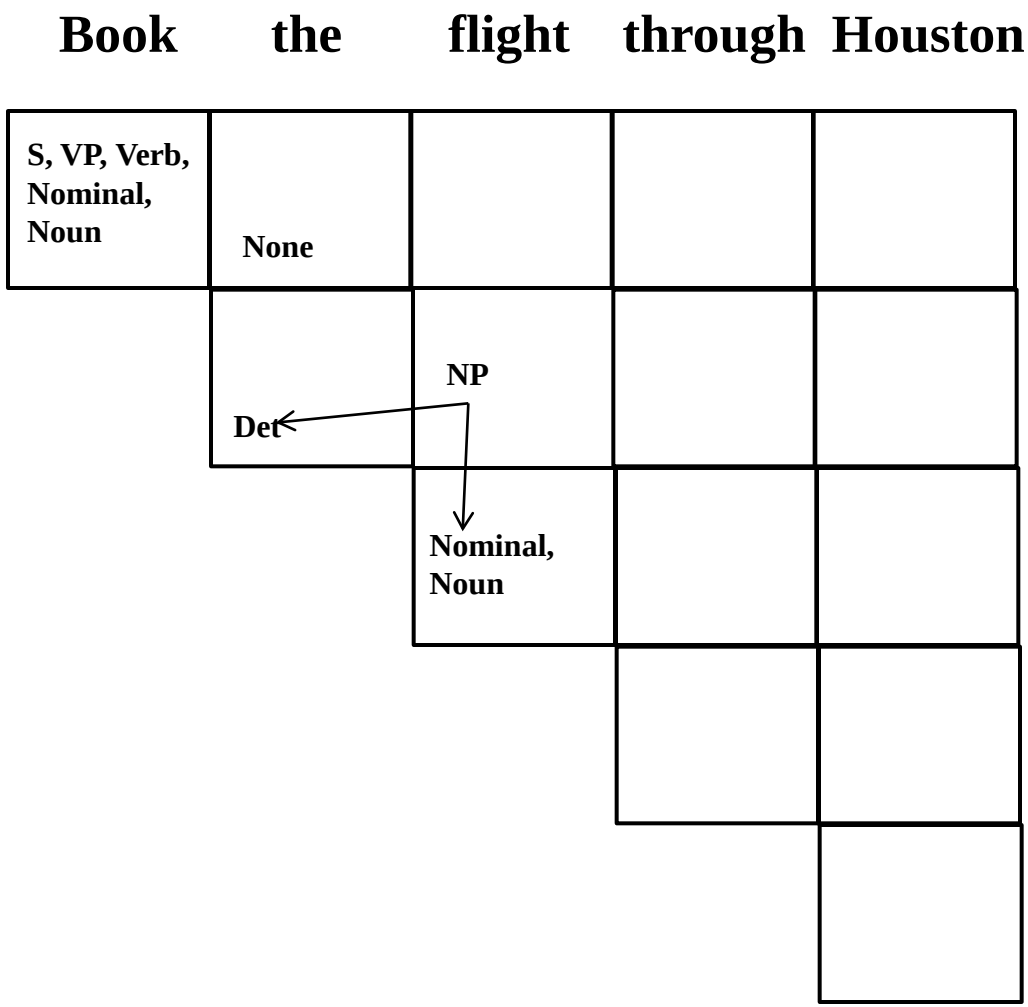
S → NP VP
S → X1 VP
X1 → Aux NP
S → book | include | prefer
S → Verb NP
S → VP PP
NP → I | he | she | me
NP → Houston | NWA
NP → Det Nominal
Nominal → book | flight | meal | money
Nominal → Nominal Noun
Nominal → Nominal PP
VP → book | include | prefer
VP → Verb NP
VP → VP PP
PP → Prep NP

CKY句法分析过程

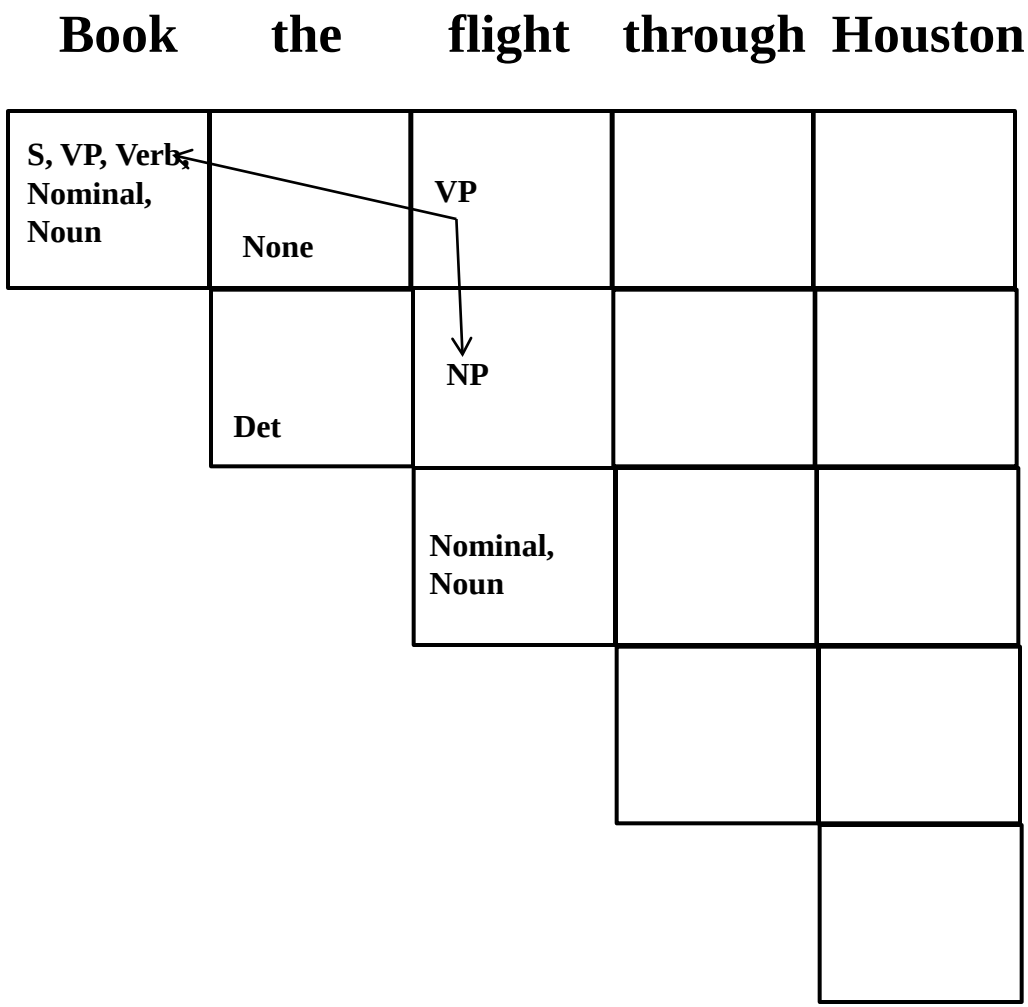


Cell[*i*,*j*]包含了所有的从i +1到 j 的非终结符信息

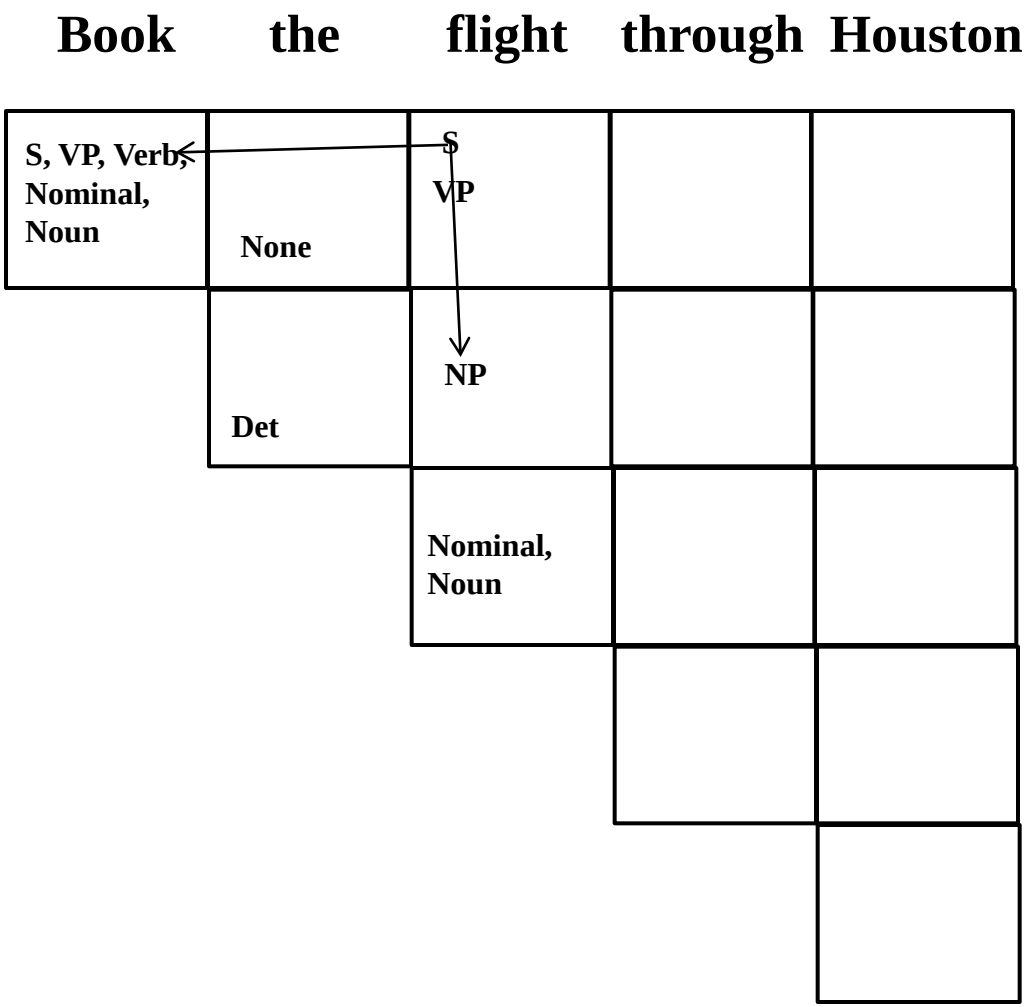
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
NP
Det
Nominal, Noun

CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
VP
NP
Det
Nominal, Noun

CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
NP
Det
Nominal, Noun

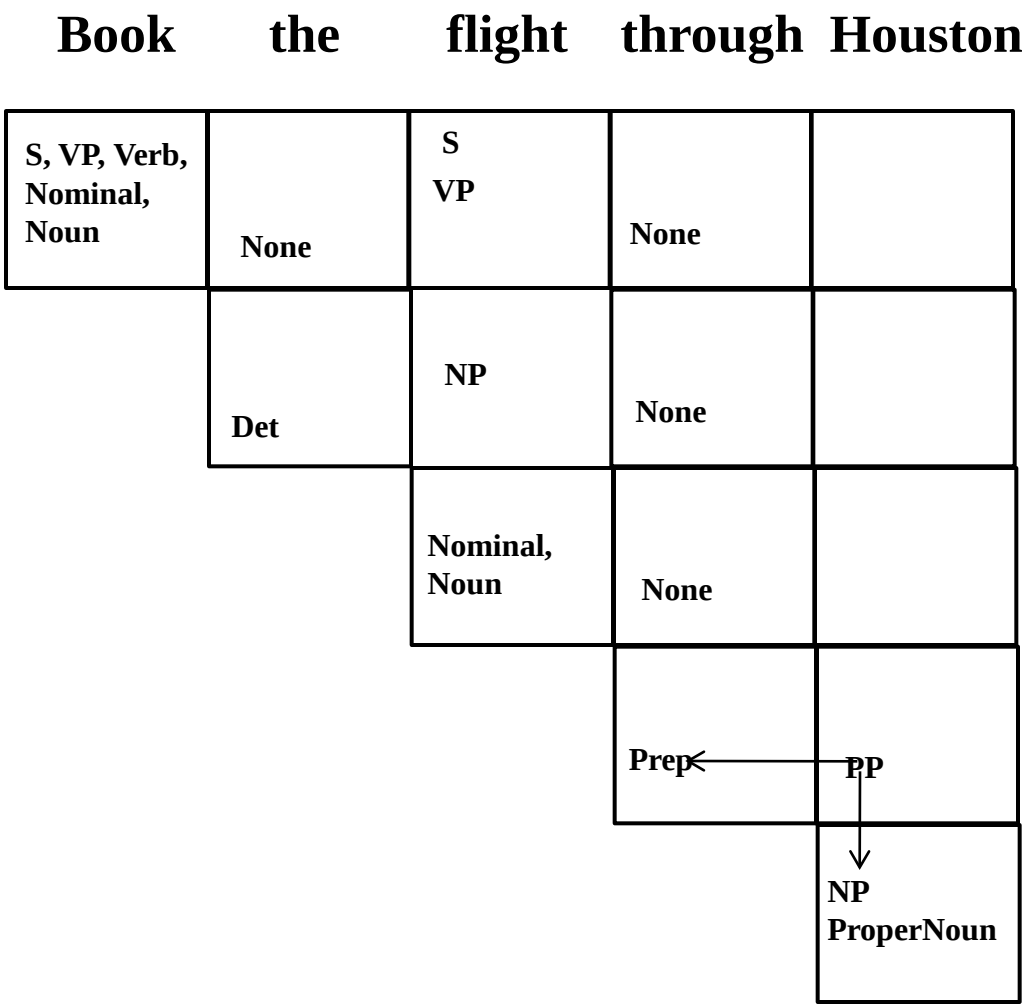
CKY句法分析过程

Book	the	flight	through	Houston
S, VP, Verb, Nominal, Noun	None	S VP		
	Det	NP		
		Nominal, Noun		

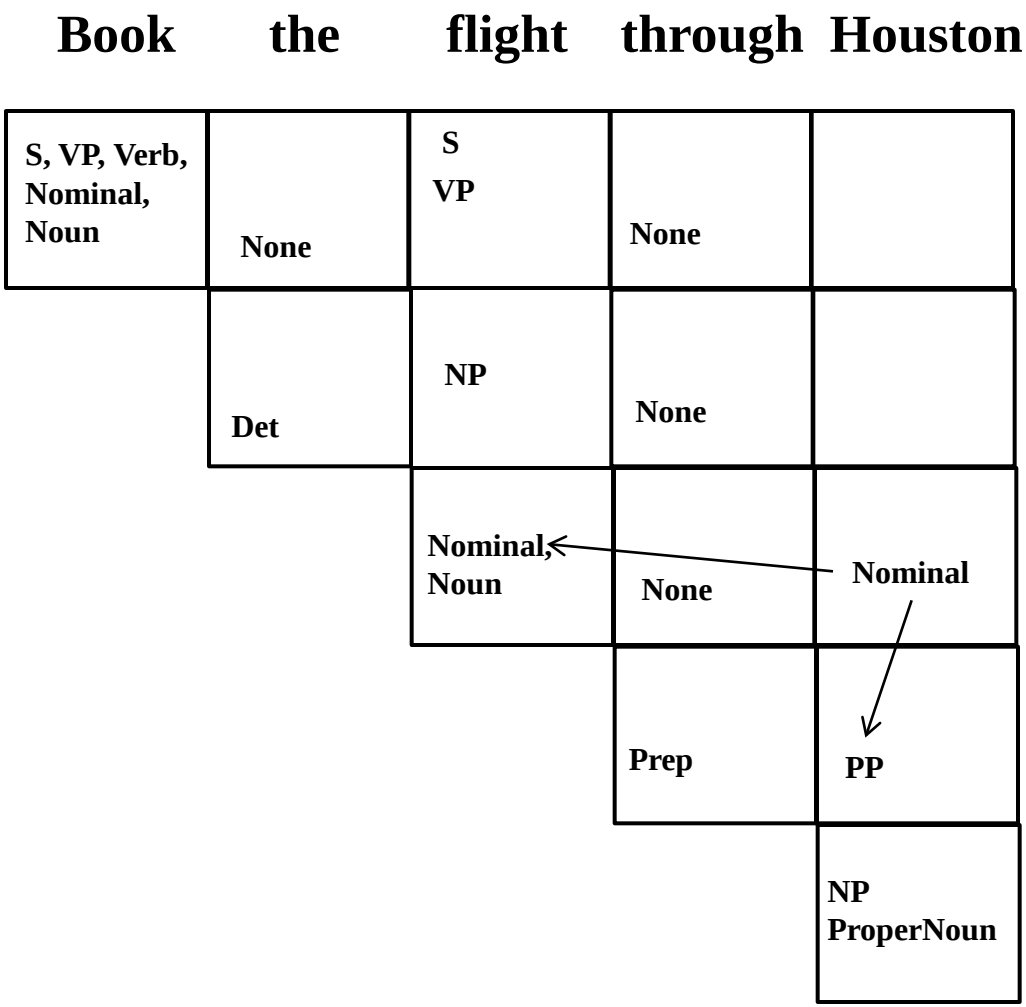
CKY句法分析过程

Book	the	flight	through	Houston
S, VP, Verb, Nominal, Noun	None	S VP	None	
	Det	NP	None	
		Nominal, Noun	None	
			Prep	

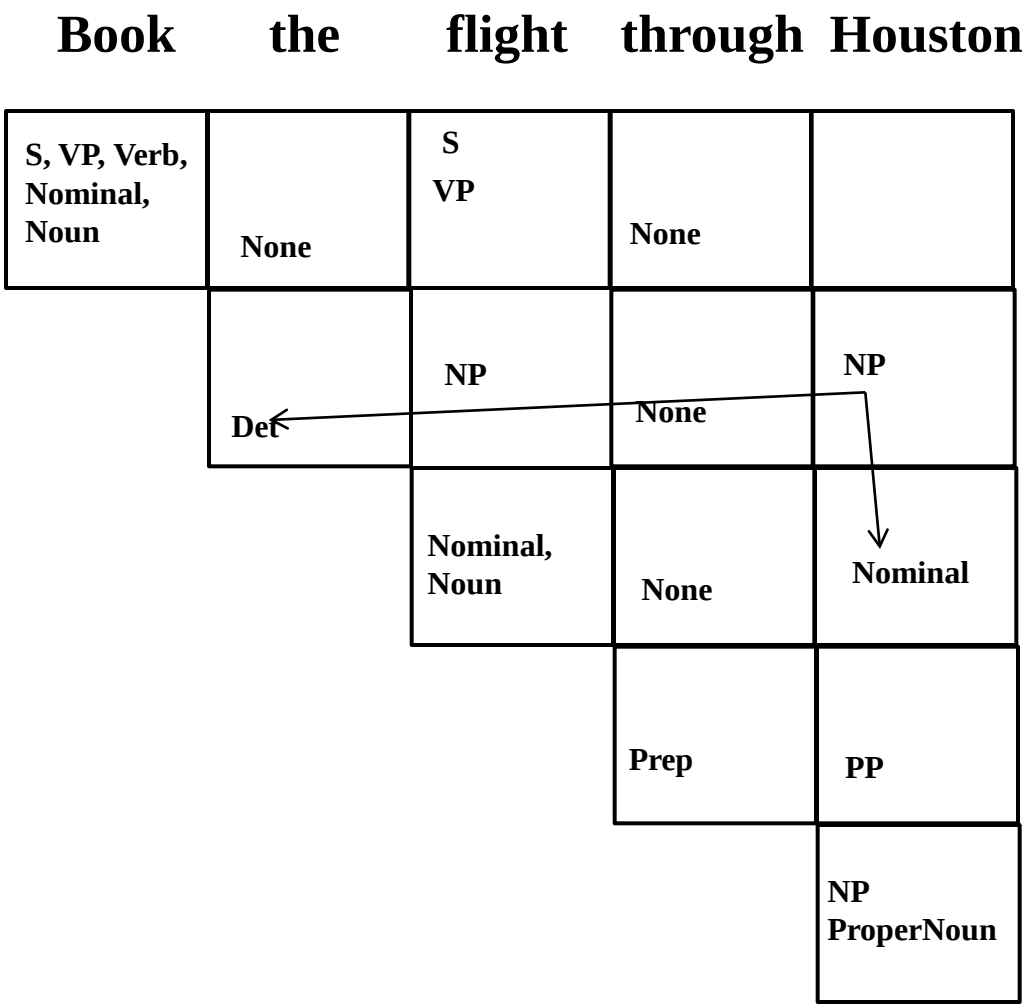
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
Det
NP
None
Nominal, Noun
None
Prep
PP
NP
ProperNoun

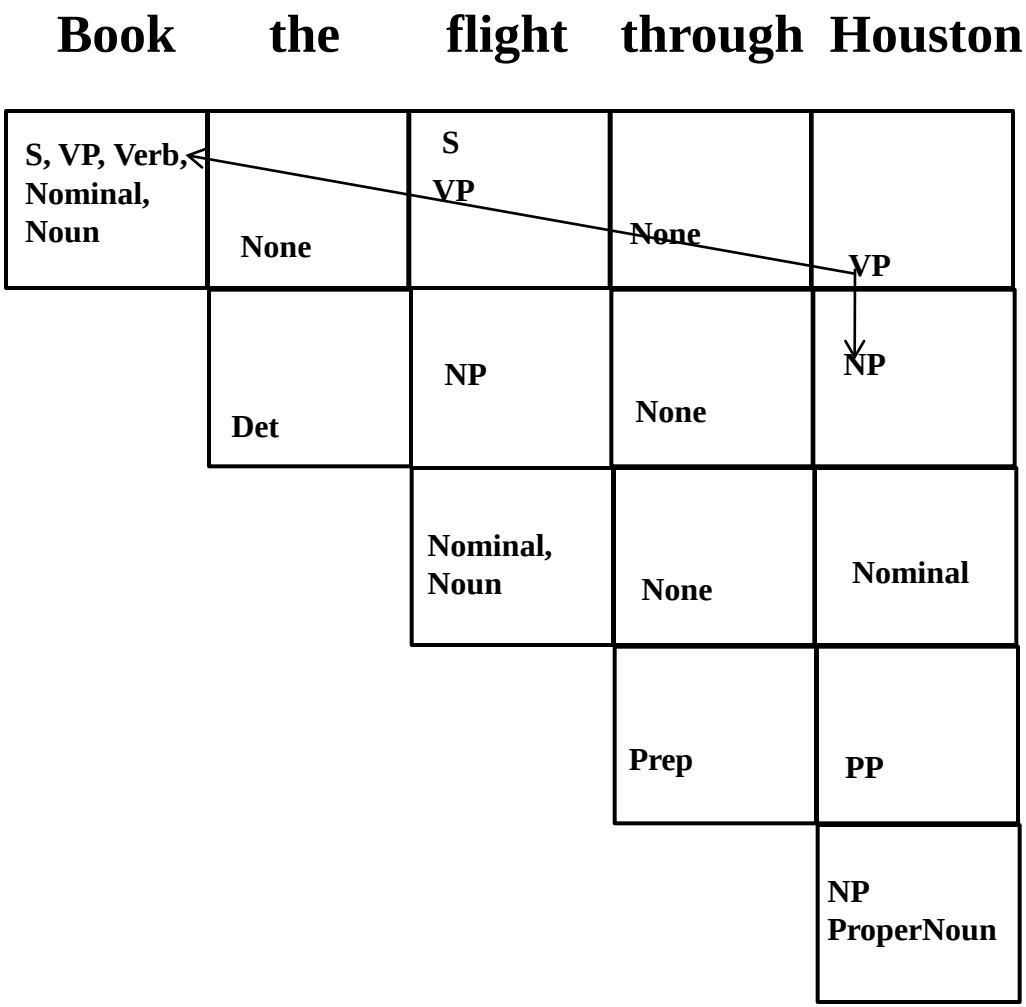
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
NP
Det
None
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun

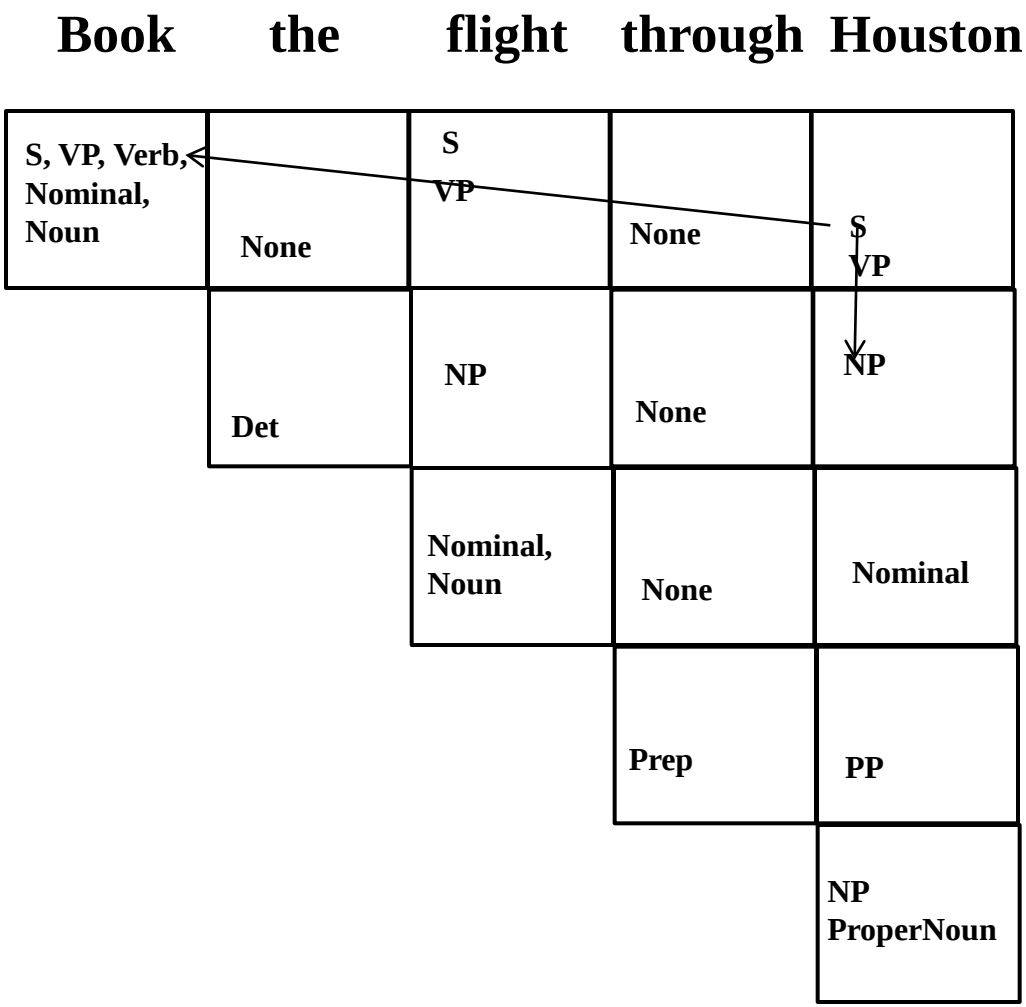
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
NP
Det
None
NP
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun

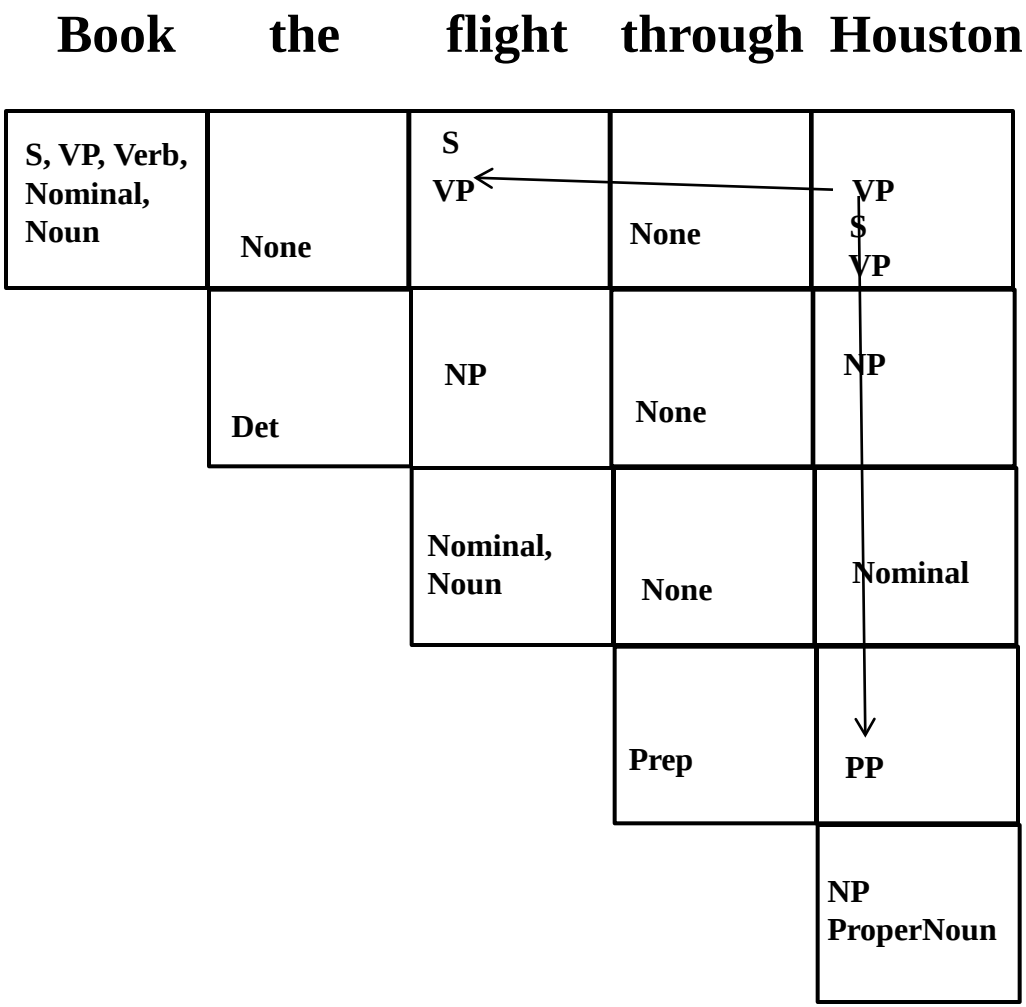
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
VP
NP
Det
NP
None
NP
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun

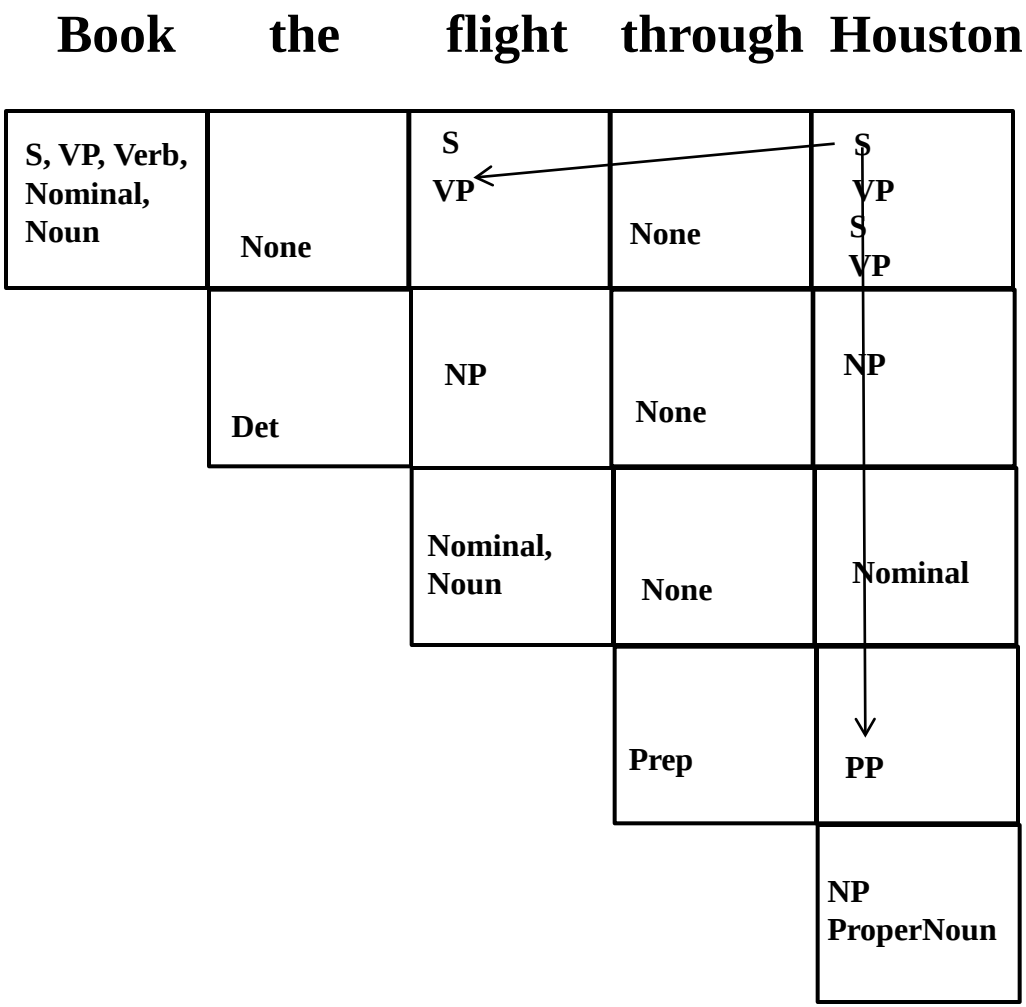
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
S
VP
NP
Det
NP
None
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun

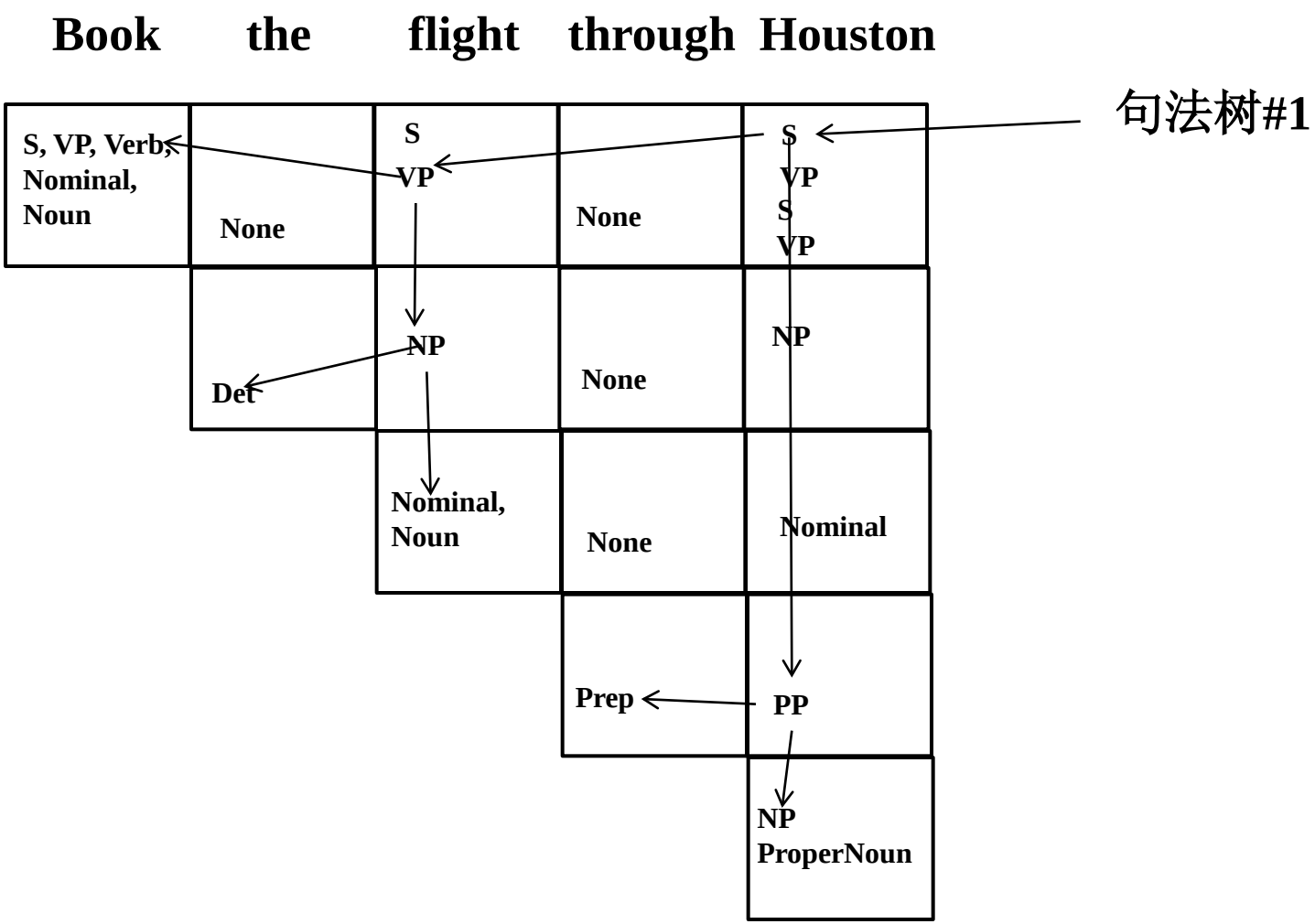
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
VP
S
VP
Det
NP
None
NP
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun

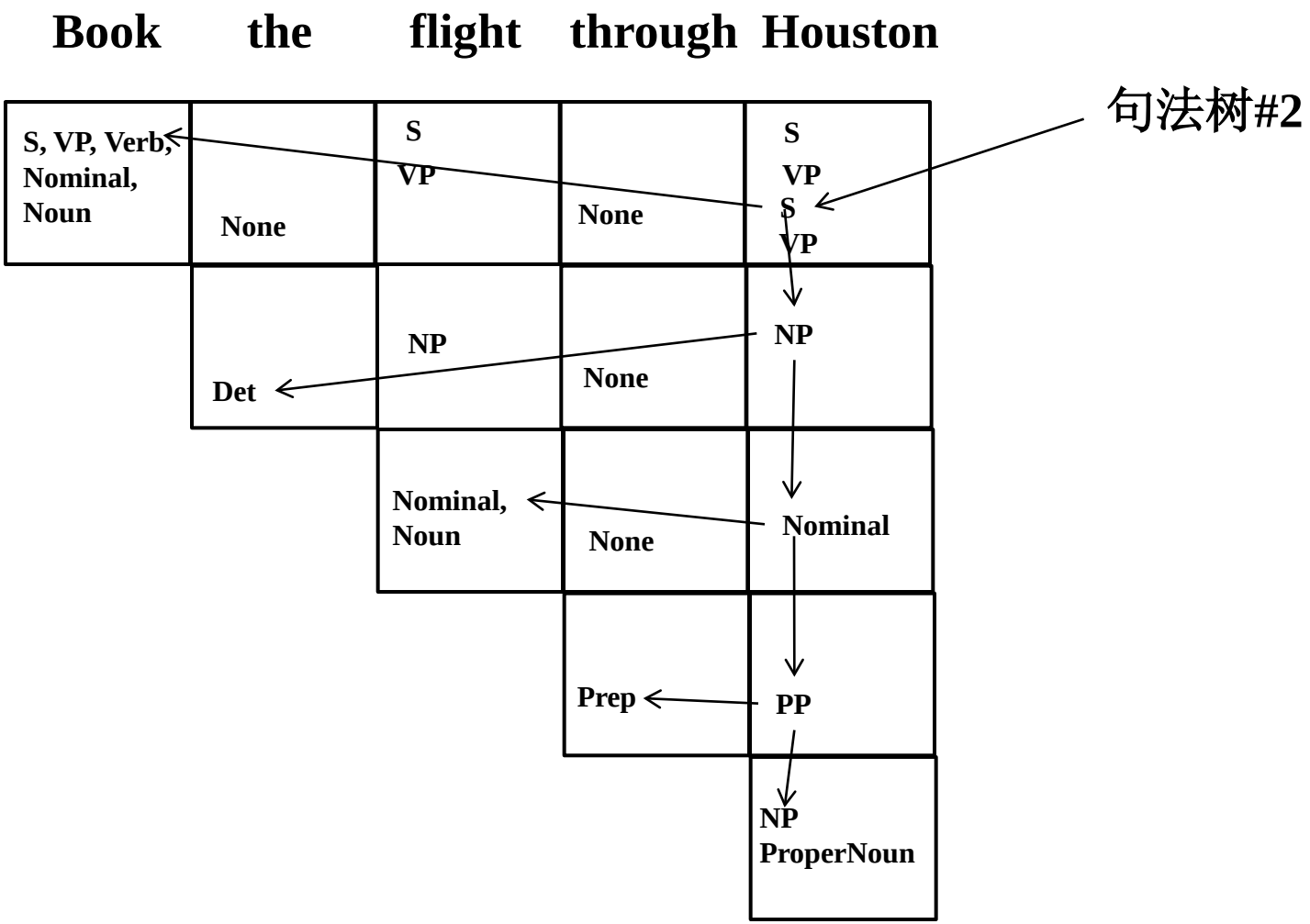
CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
S
VP
S
VP
Det
NP
None
NP
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun

CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
S
VP
S
VP
Det
NP
None
NP
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun
句法树#1

CKY句法分析过程


Book
the
flight
through
Houston
S, VP, Verb, Nominal, Noun
None
S
VP
None
S
VP
S
VP
NP
Det
NP
None
NP
Nominal, Noun
None
Nominal
Prep
PP
NP
ProperNoun
句法树#2

CKY (recognition)的复杂度

- 总共有 $(n(n+1)/2) = O(n^2)$ 个单元格
- 总体的时间复杂度是 $O(n^3)$
- 分析得到的句法树（parse tree）是基于乔姆斯基范式的
- 有必要的话，可以用一个后处理过程把乔姆斯基范式转换回原来的语法规则结构

总结

- 自动句法分析有助于计算机理解句子的意思
  - John ate the spaghetti with meatballs with chopsticks.
    - How did John eat the spaghetti?
    - What did John eat?

- 动态规划算法可以在3次方时间内计算一棵句法树，或指数时间内计算所有的句法树

- 问题：只是单纯的输出符合规则的句法树，无法确定哪个句法树的概率最大
  - 解决方法：基于概率的句法分析

基于概率的上下文无关文法
(Probabilistic Context Free Grammar)

基于概率的句法分析

- **使用概率模型对每个句法树赋予一个概率信息**
  - 通过概率信息消解句法分析中的歧义现象
  - 在标注好的树库的基础上，实现有监督学习
  - 也可以实现无监督学习，但是目前的无监督学习效果比较有限

- **基于概率的上下文无关文法 (Probabilistic Context Free Grammar, PCFG)**
  - 基于概率的上下文无关文法(PCFG)是上下文无关文法(CFG)的概率版本
  - 每个生成规则都带有概率信息

PCFG举例

语法规则	概率		
S → NP VP	0.8		
S → Aux NP VP	0.1	+	1.0
S → VP	0.1		
NP → Pronoun	0.2		
NP → Proper-Noun	0.2	+	1.0
NP → Det Nominal	0.6		
Nominal → Noun	0.3		
Nominal → Nominal Noun	0.2	+	1.0
Nominal → Nominal PP	0.5		
VP → Verb	0.2		
VP → Verb NP	0.5	+	1.0
VP → VP PP	0.3		
PP → Prep NP	1.0		

词汇规则

Det → the | a | that | this
0.6 0.2 0.1 0.1

Noun → book | flight | meal | money
0.1 0.5 0.2 0.2

Verb → book | include | prefer
0.5 0.2 0.3

Pronoun → I | he | she | me
0.5 0.1 0.1 0.3

Proper-Noun → Houston | NWA
0.8 0.2

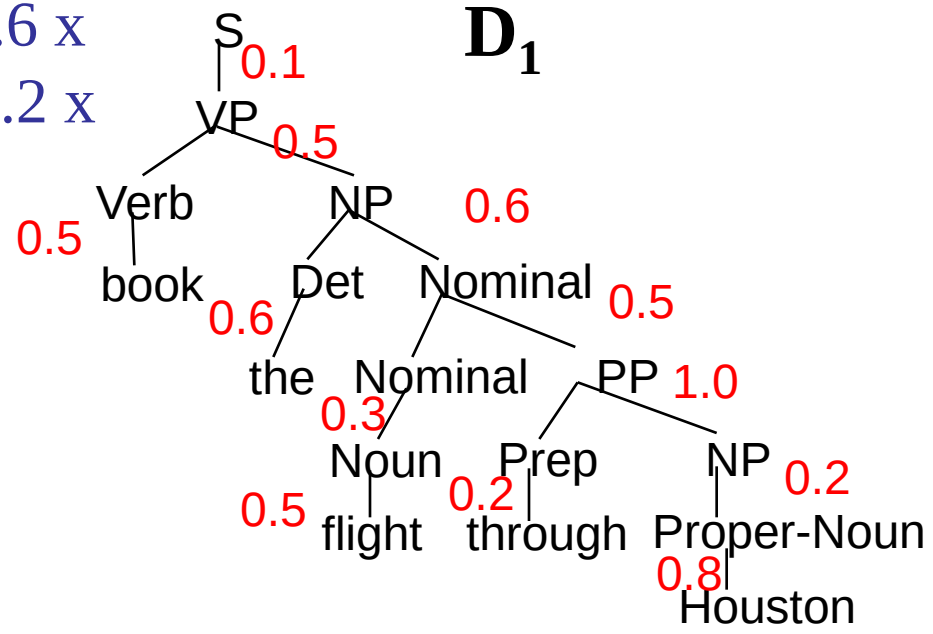
Aux → does
1.0

Prep → from | to | on | near | through
0.25 0.25 0.1 0.2 0.2

计算句法树、句子的概率

- 独立性假设
- 概率相乘

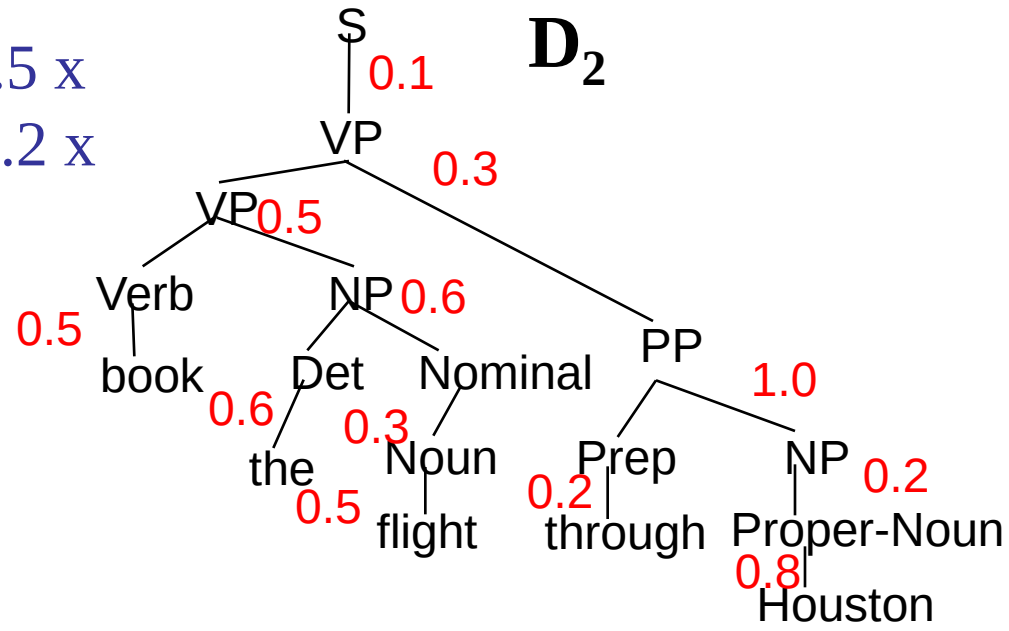
$$P(D_1) = 0.1 \times 0.5 \times 0.5 \times 0.6 \times 0.6 \times 0.5 \times 0.3 \times 1.0 \times 0.2 \times 0.2 \times 0.5 \times 0.8 = 0.0000216$$



计算句法树、句子的概率

- 挑选概率最大的句法树作为句法分析的结果

$$P(D_2) = 0.1 \times 0.3 \times 0.5 \times 0.6 \times 0.5 \times 0.6 \times 0.3 \times 1.0 \times 0.5 \times 0.2 \times 0.2 \times 0.8 = 0.00001296$$



77

计算句法树、句子的概率

- 也可以计算句子本身的概率，句子本身的概率是其所有句法树概率之和

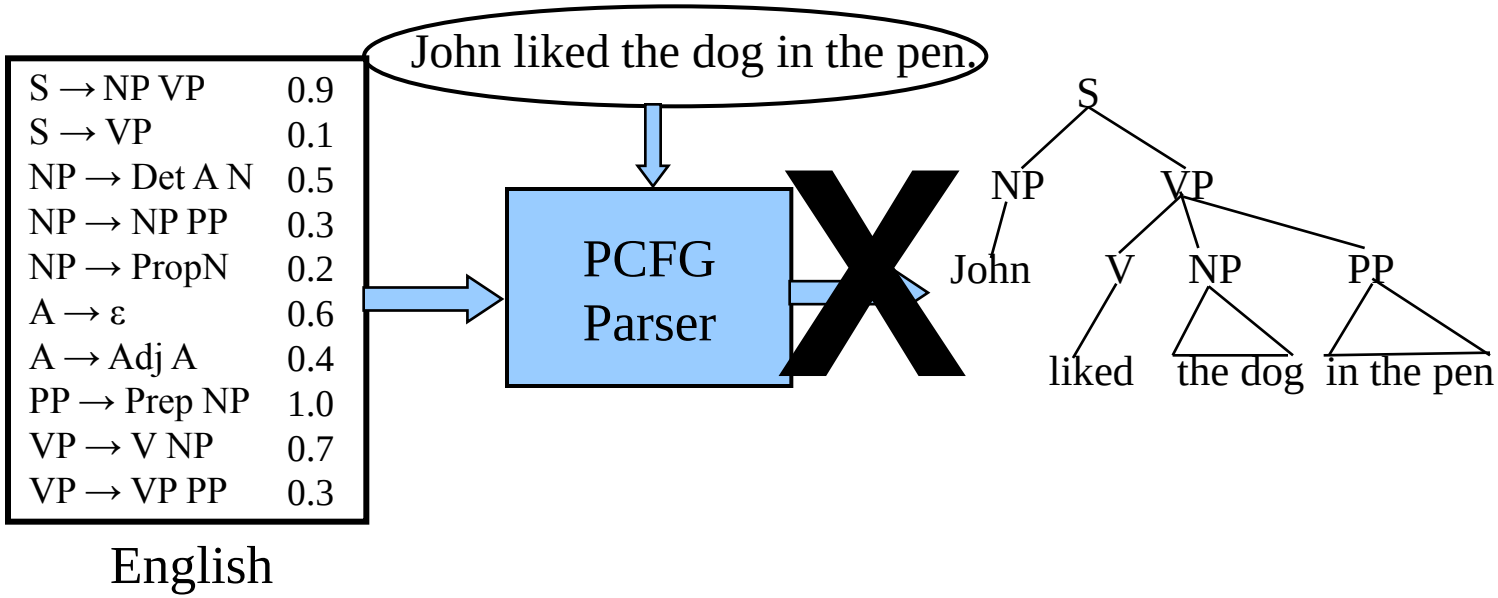
P(“book the flight through Houston”) =
$P(D_1) + P(D_2) = 0.0000216 + 0.00001296$
$= 0.00003456$

三个主要的PCFG 任务

- **观测概率（Observation likelihood）**
  - 用于对句子排序等
- **最大概率句法树（Most likely derivation）**
  - 找出最大概率句法树
- **最大似然训练（Maximum likelihood training）**
  - 基于训练数据训练一个句法分析器

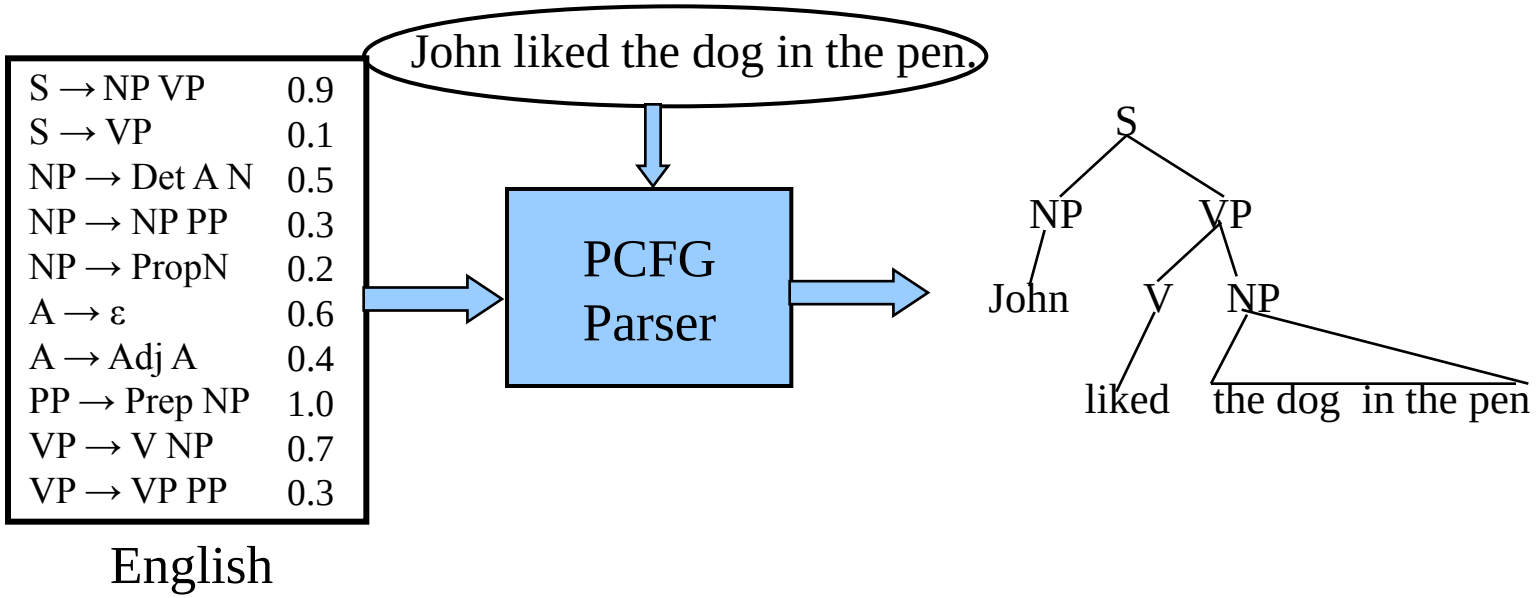
PCFG: 最大概率的句法树

- 可以使用维特比（**Viterbi**）算法确定最大可能的句法树



PCFG: 最大概率的句法树

- 可以使用维特比（**Viterbi**）算法确定最大可能的句法树



基于概率的CKY算法（Probabilistic CKY）

- 在原来的**CKY**算法的基础上加入概率信息

- 当转换为乔姆斯基范式的时候，需要重新设置概率信息从而保证原有的概率分布

基于概率的乔姆斯基范式转换

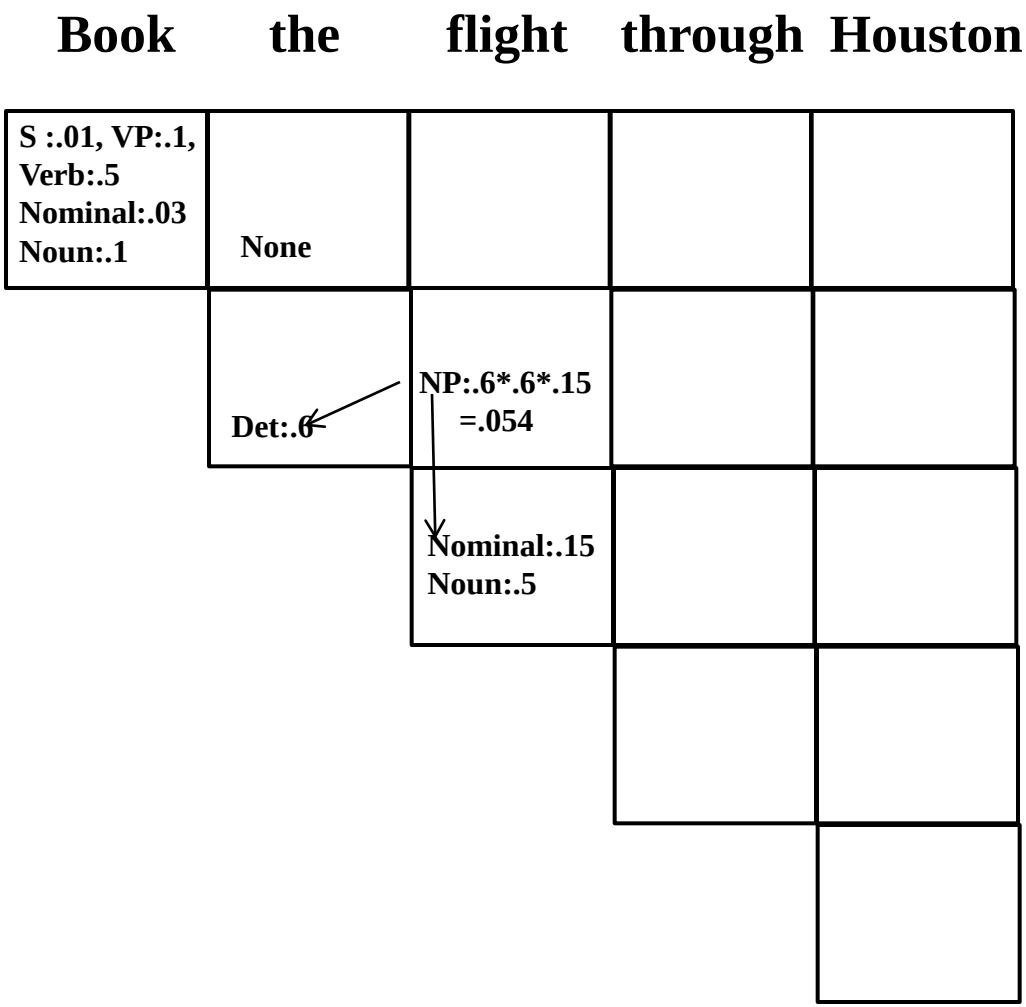
原始语法规则

S → NP VP	0.8
S → Aux NP VP	0.1
S → VP	0.1
NP → Pronoun	0.2
NP → Proper-Noun	0.2
NP → Det Nominal	0.6
Nominal → Noun	0.3
Nominal → Nominal Noun	0.2
Nominal → Nominal PP	0.5
VP → Verb	0.2
VP → Verb NP	0.5
VP → VP PP	0.3
PP → Prep NP	1.0

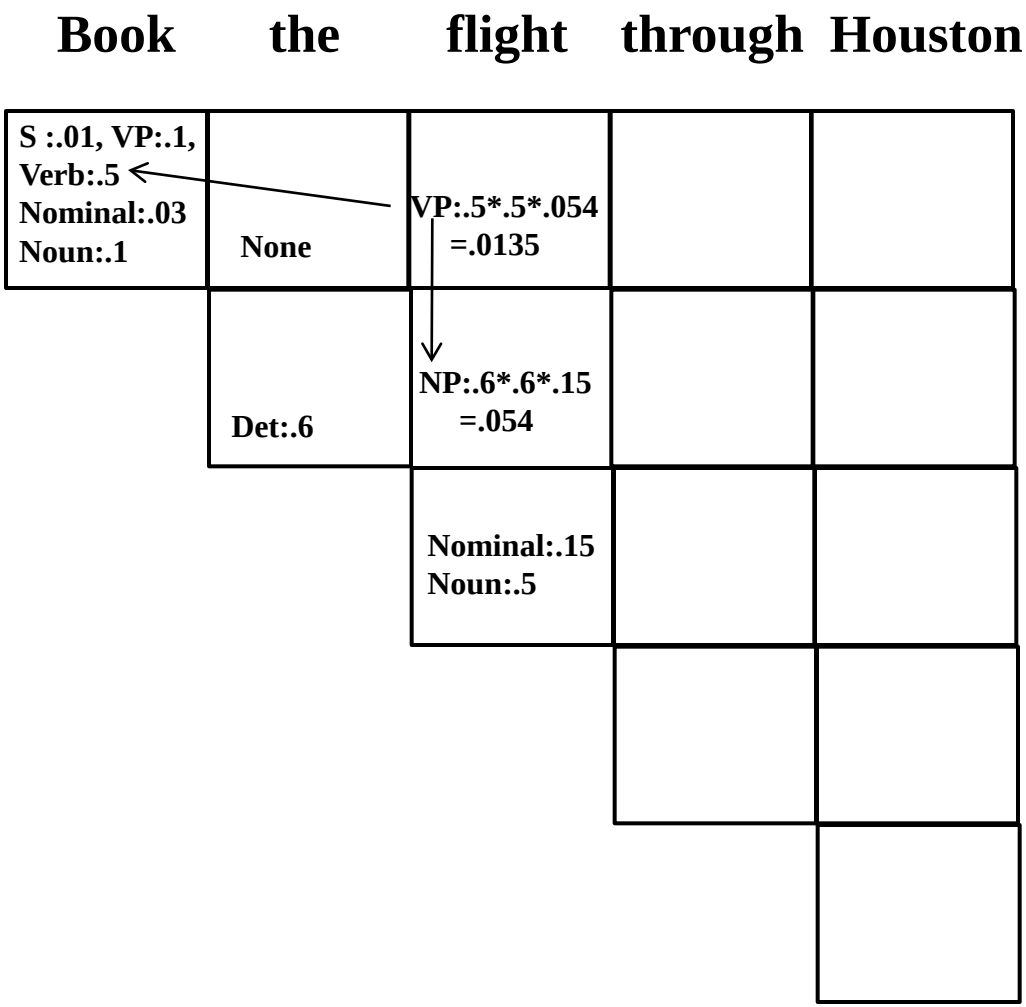
乔姆斯基范式

S → NP VP	0.8
S → X1 VP	0.1
X1 → Aux NP	1.0
S → book | include | prefer 0.01 0.004 0.006	
S → Verb NP	0.05
S → VP PP	0.03
NP → I | he | she | me 0.1 0.02 0.02 0.06	
NP → Houston | NWA 0.16 .04	
NP → Det Nominal	0.6
Nominal → book | flight | meal | money 0.03 0.15 0.06 0.06	
Nominal → Nominal Noun	0.2
Nominal → Nominal PP	0.5
VP → book | include | prefer 0.1 0.04 0.06	
VP → Verb NP	0.5
VP → VP PP	0.3
PP → Prep NP	1.0

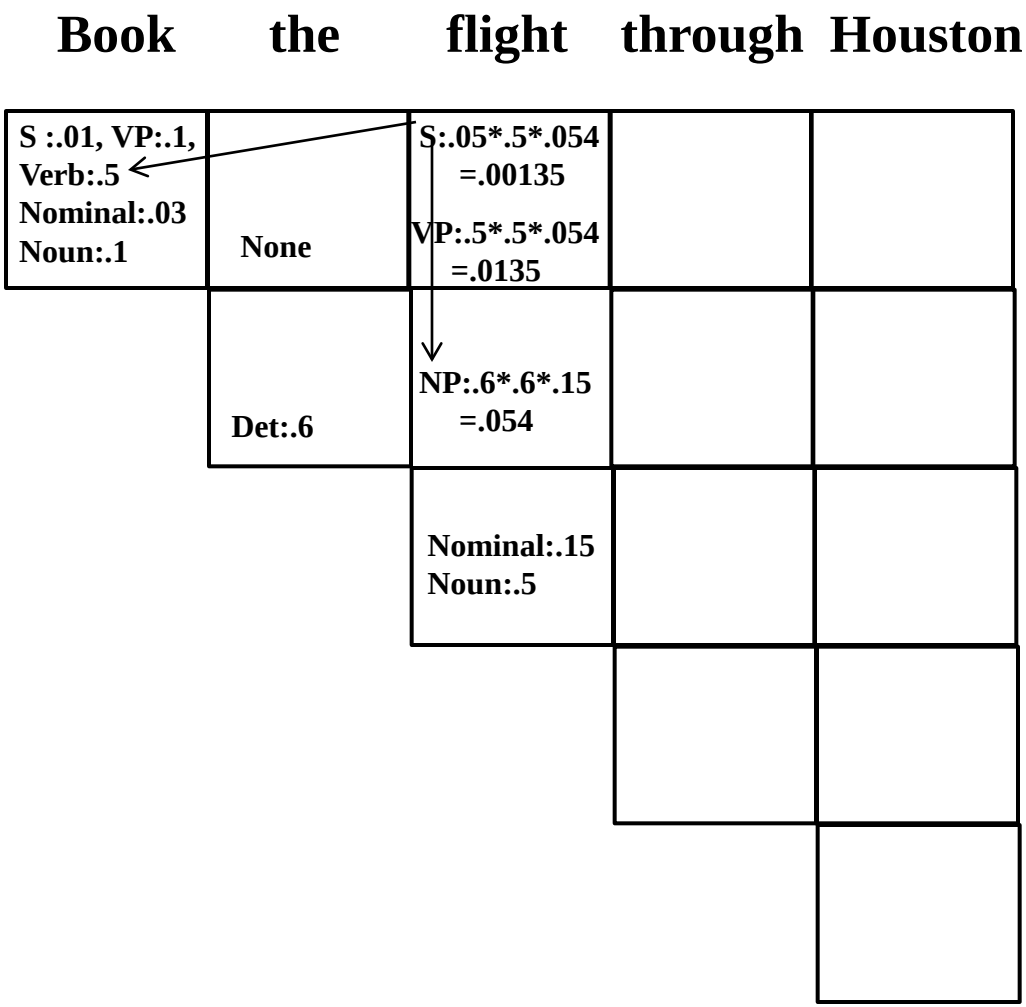
基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1,
Verb:.5
Nominal:.03
Noun:.1
None
Det:.6
NP:.6*.6*.15
=.054
Nominal:.15
Noun:.5

基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1,
Verb:.5
Nominal:.03
Noun:.1
None
VP:.5*.5*.054
=.0135
NP:.6*.6*.15
=.054
Det:.6
Nominal:.15
Noun:.5

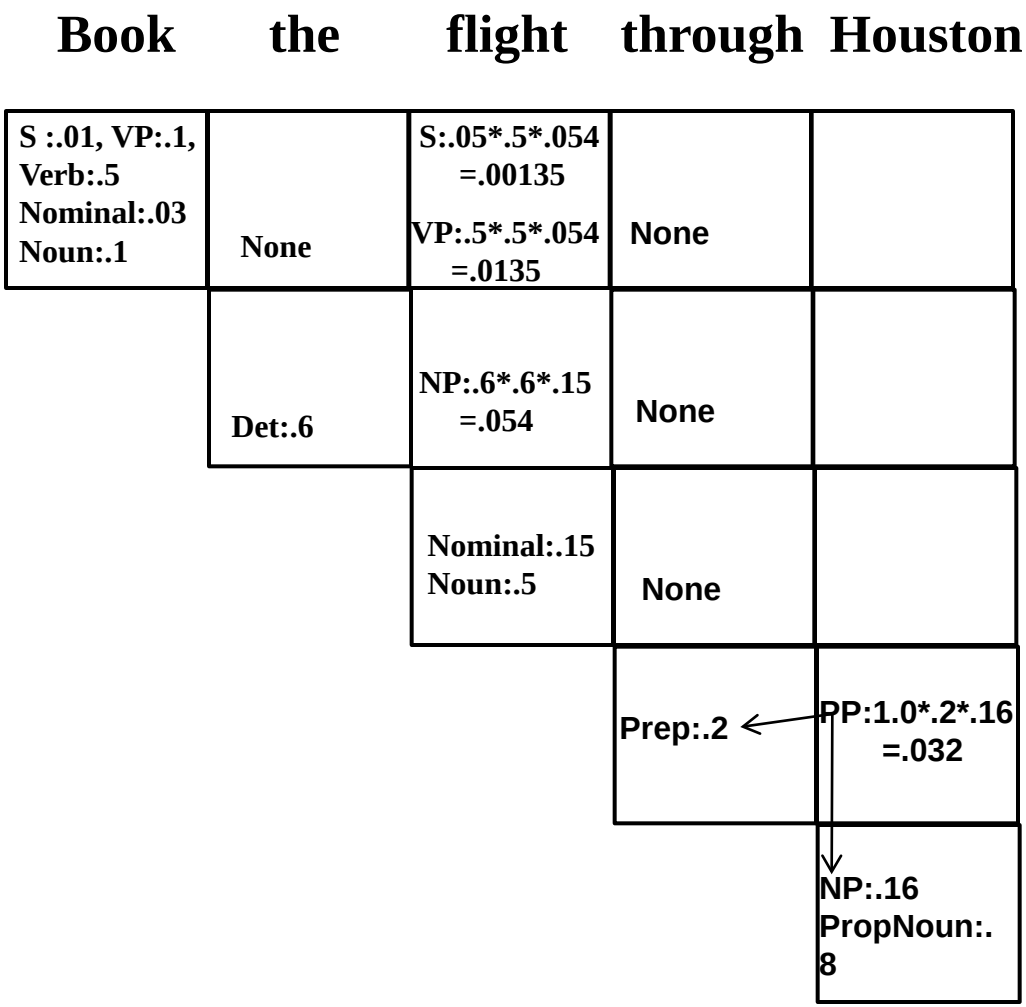
基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1,
Verb:.5
Nominal:.03
Noun:.1
None
S:.05*.5*.054
=.00135
VP:.5*.5*.054
=.0135
Det:.6
NP:.6*.6*.15
=.054
Nominal:.15
Noun:.5

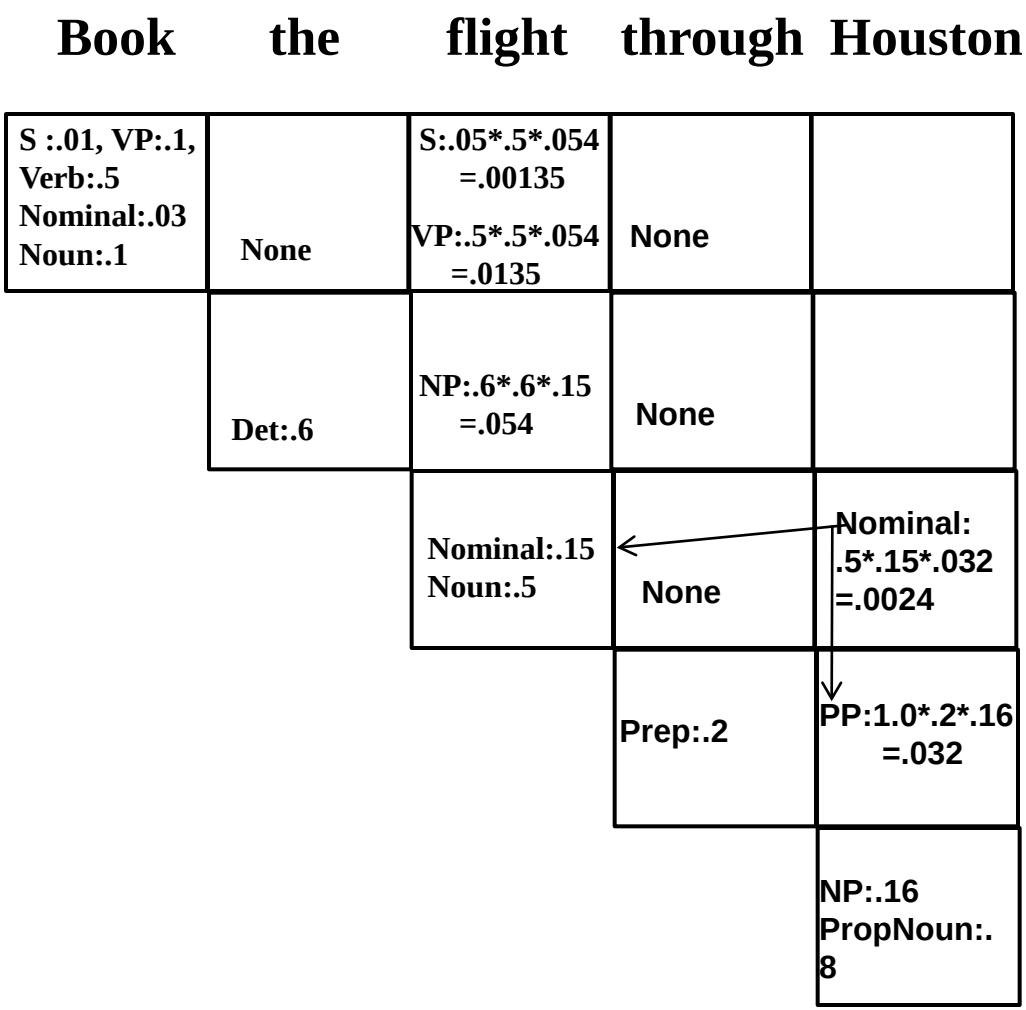
基于概率的CKY算法

Book	the	flight	through	Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1	None	S:.05*.5*.054 =.00135 VP:.5*.5*.054 =.0135	None	
	Det:.6	NP:.6*.6*.15 =.054	None	
		Nominal:.15 Noun:.5	None	
			Prep:.2	

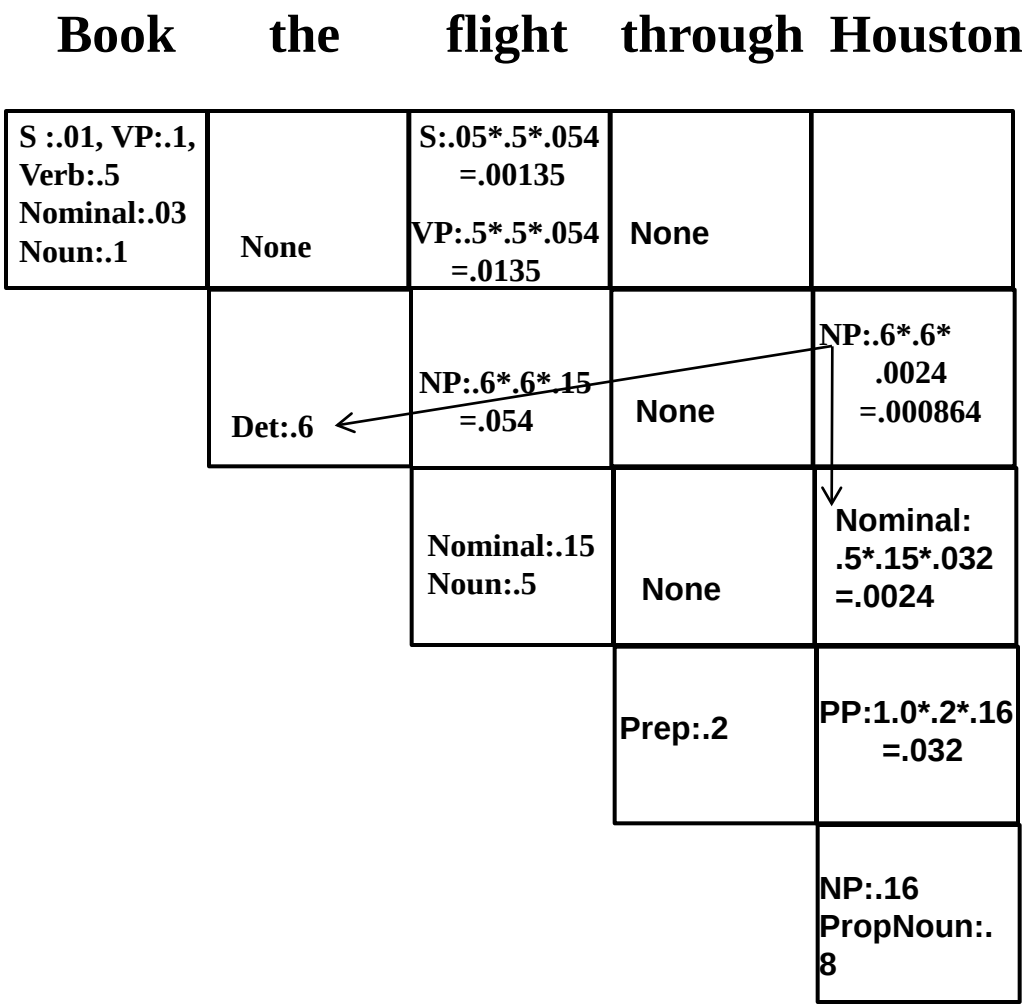
基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1
None
S:.05*.5*.054 =.00135
VP:.5*.5*.054 =.0135
None
Det:.6
NP:.6*.6*.15 =.054
None
Nominal:.15 Noun:.5
None
Prep:.2
PP:1.0*.2*.16 =.032
NP:.16 PropNoun:.8

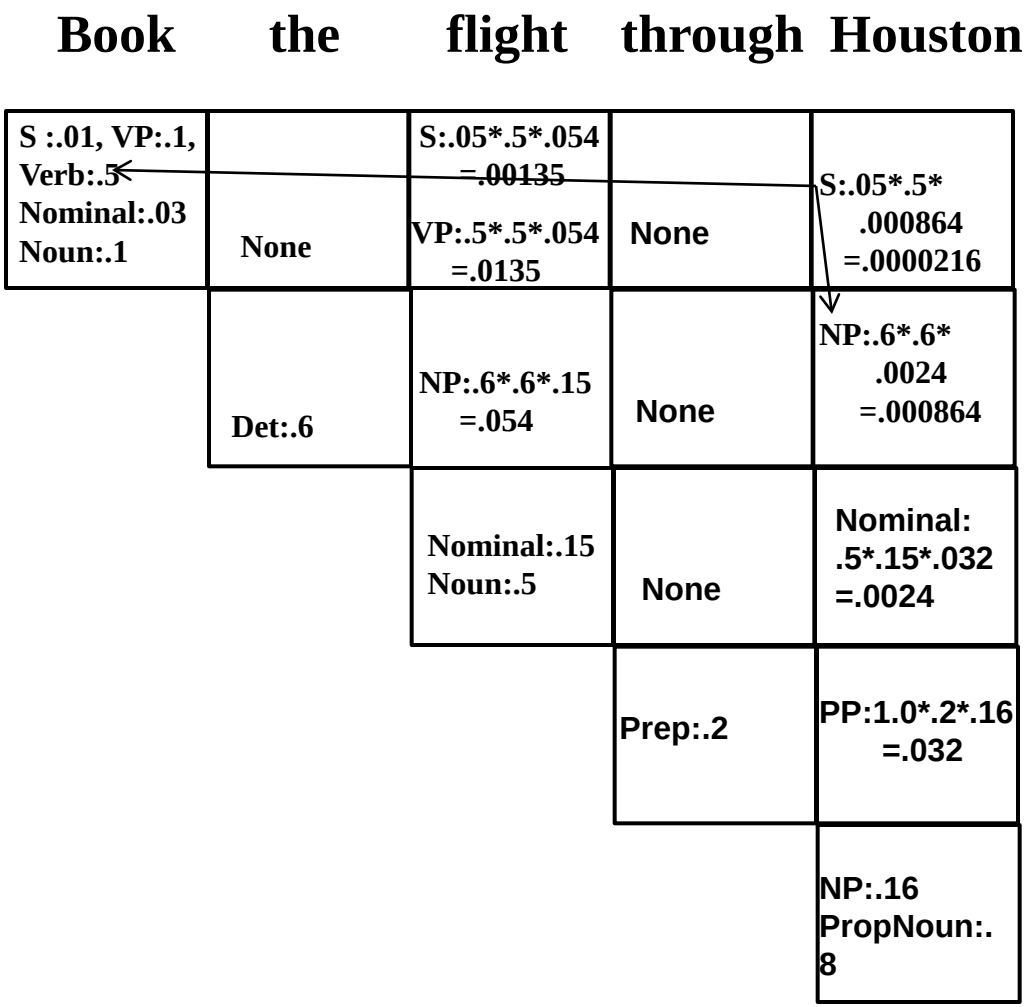
基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1
None
S:.05*.5*.054 =.00135
VP:.5*.5*.054 =.0135
None
Det:.6
NP:.6*.6*.15 =.054
None
Nominal:.15 Noun:.5
None
Nominal: .5*.15*.032 =.0024
Prep:.2
PP:1.0*.2*.16 =.032
NP:.16 PropNoun:. 8

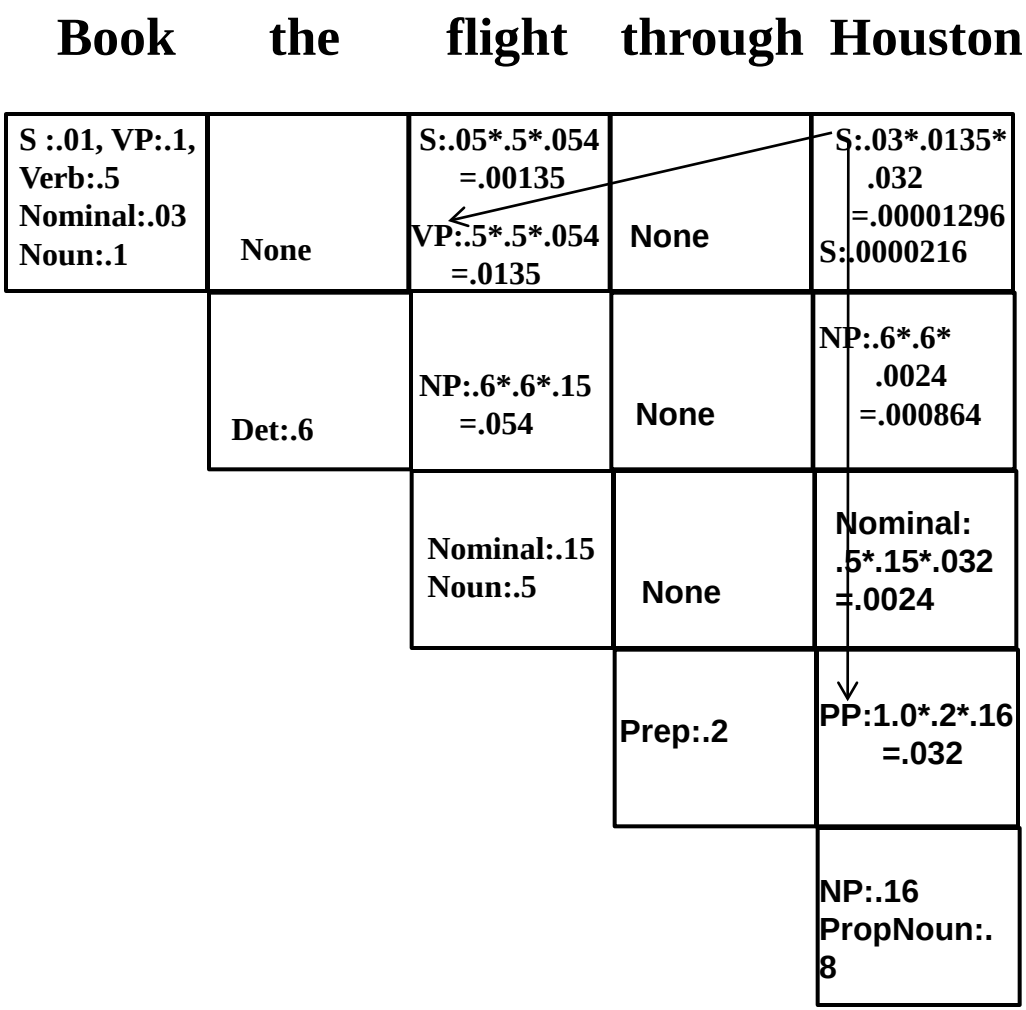
基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1
None
S:.05*.5*.054 =.00135
VP:.5*.5*.054 =.0135
None
NP:.6*.6* .0024 =.000864
Det:.6
NP:.6*.6*.15 =.054
None
Nominal: .5*.15*.032 =.0024
Nominal:.15 Noun:.5
None
PP:1.0*.2*.16 =.032
Prep:.2
NP:.16 PropNoun:. 8

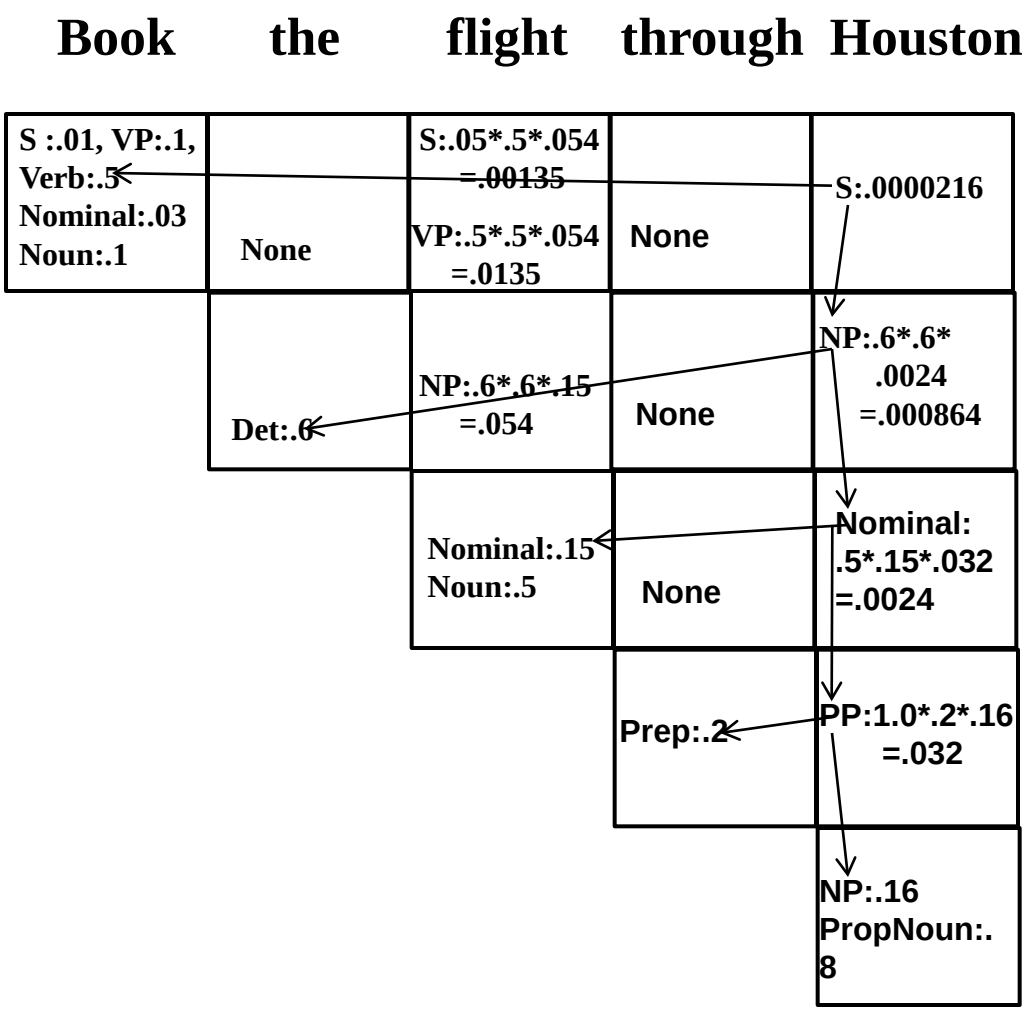
基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1
None
S:.05*.5*.054 =.00135
VP:.5*.5*.054 =.0135
None
S:.05*.5* .000864 =.0000216
Det:.6
NP:.6*.6*.15 =.054
None
NP:.6*.6* .0024 =.000864
Nominal:.15 Noun:.5
None
Nominal: .5*.15*.032 =.0024
Prep:.2
PP:1.0*.2*.16 =.032
NP:.16 PropNoun:.8

基于概率的CKY算法


Book
the
flight
through
Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1
None
S:.05*.5*.054 =.00135
VP:.5*.5*.054 =.0135
None
S:.03*.0135* .032 =.00001296
S:.0000216
Det:.6
NP:.6*.6*.15 =.054
None
NP:.6*.6* .0024 =.000864
Nominal:.15 Noun:.5
None
Nominal: .5*.15*.032 =.0024
Prep:.2
PP:1.0*.2*.16 =.032
NP:.16 PropNoun:.8

基于概率的CKY算法

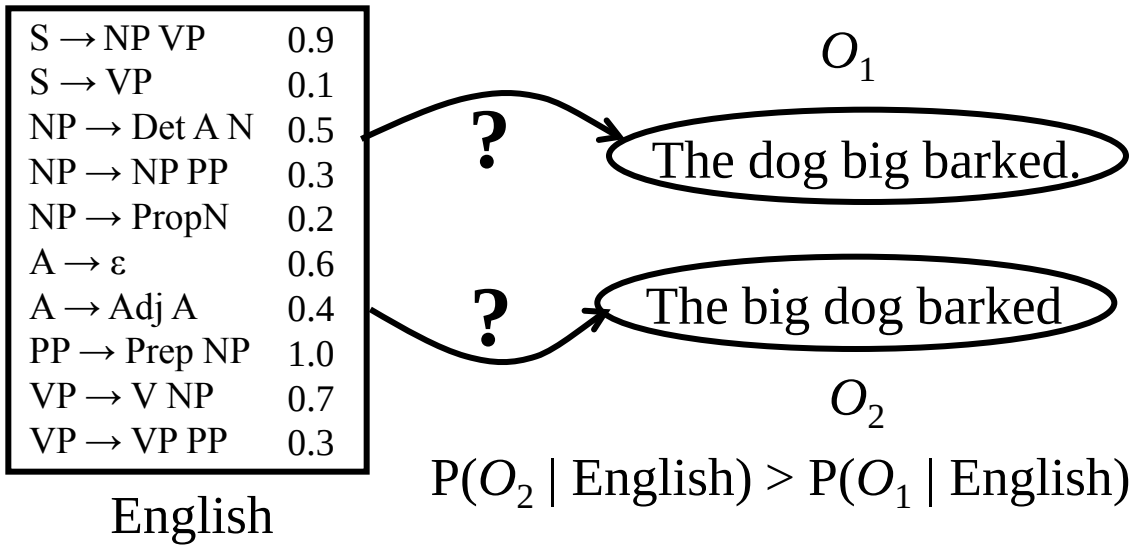


计算最大概率的句法树

PCFG: 观测概率的计算(Observation Likelihood)

- 类似于序列标注问题中的观测概率的计算方法—前向算法(**Forward algorithm**)，可以使用 **Inside algorithm** 这个算法进行动态规划计算观测概率
  - 把Viterbi算法的max计算改为sum计算

- 可以使用**PCFG**作为一个语言模型，从而对句子的概率进行计算，用于语音识别、机器翻译等



PCFG: 观测概率的计算(Observation Likelihood)

Probabilistic CKY Parser for Inside Computation

Book	the	flight	through	Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1	None	S:.05*.5*.054 =.00135 VP:.5*.5*.054 =.0135	None	S:..00001296 S:.0000216
	Det:.6	NP:.6*.6*.15 =.054	None	NP:.6*.6* .0024 =.000864
		Nominal:.15 Noun:.5	None	Nominal: .5*.15*.032 =.0024
			Prep:.2	PP:1.0*.2*.16 =.032
				NP:.16 PropNoun:. 8

PCFG: 观测概率的计算(Observation Likelihood)

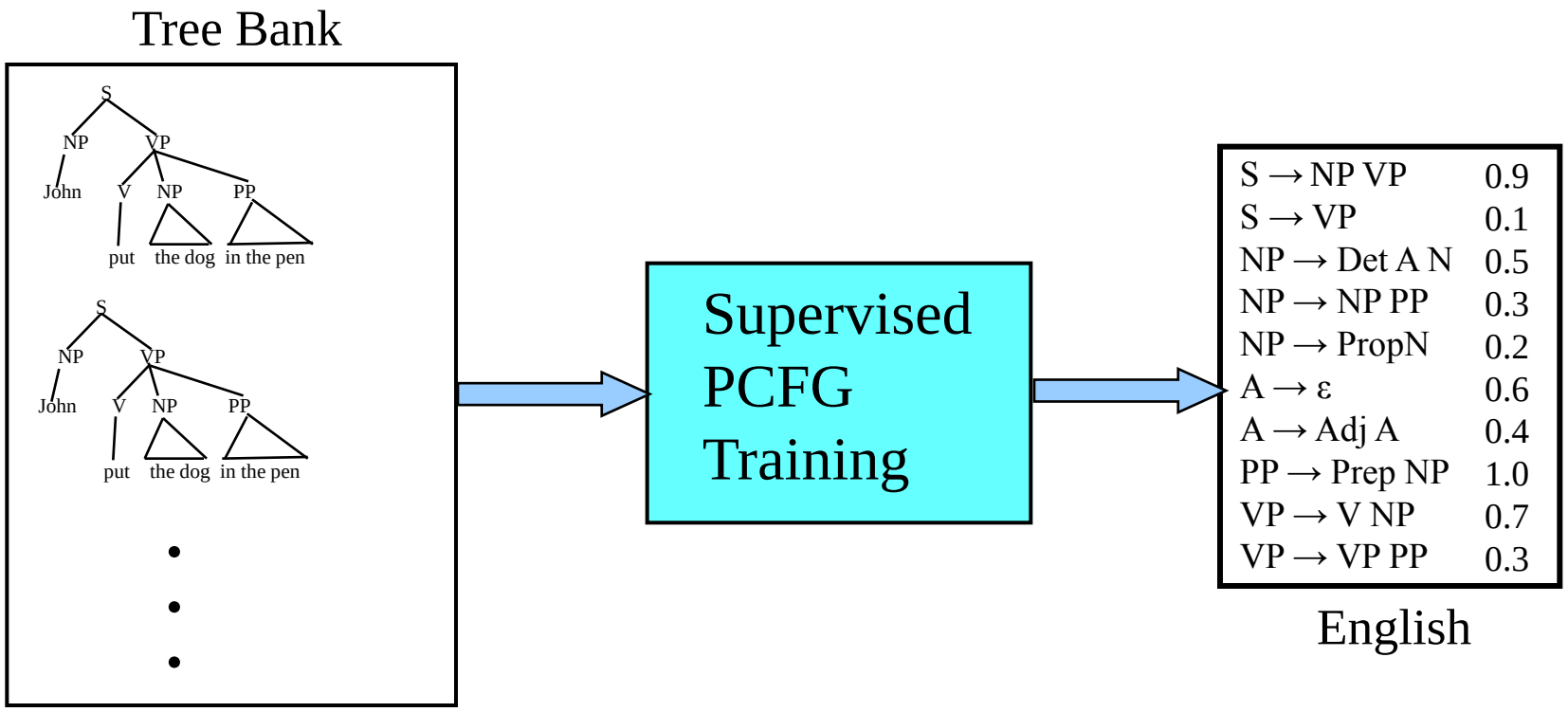
Probabilistic CKY Parser for Inside Computation

Book	the	flight	through	Houston
S :.01, VP:.1, Verb:.5 Nominal:.03 Noun:.1	None	S:.05*.5*.054 =.00135 VP:.5*.5*.054 =.0135	None	S: .00001296 +.0000216 =.00003456
	Det:.6	NP:.6*.6*.15 =.054	None	NP:.6*.6* .0024 =.000864
		Nominal:.15 Noun:.5	None	Nominal: .5*.15*.032 =.0024
			Prep:.2	PP:1.0*.2*.16 =.032
				NP:.16 PropNoun:.8

计算观测概率

PCFG: 有监督学习(Supervised Training)

- **给定了训练数据的话（一般是标注好的树库tree bank），PCFG的有监督学习相对比较简单**
  - 可以通过相对频率来计算



PCFG: 有监督学习(Supervised Training)

- 可以从树库收集语法规则
- 语法规则对应的概率可以通过相对频率来计算

$$P(\alpha \to \beta \mid \alpha) = \frac{\text{count}(\alpha \to \beta)}{\sum_{\gamma} \text{count}(\alpha \to \gamma)} = \frac{\text{count}(\alpha \to \beta)}{\text{count}(\alpha)}$$

句法分析的效果打分

- 句法分析器给出了预测的句法树之后，可以计算句法树和已经标注好的句法树的相似度

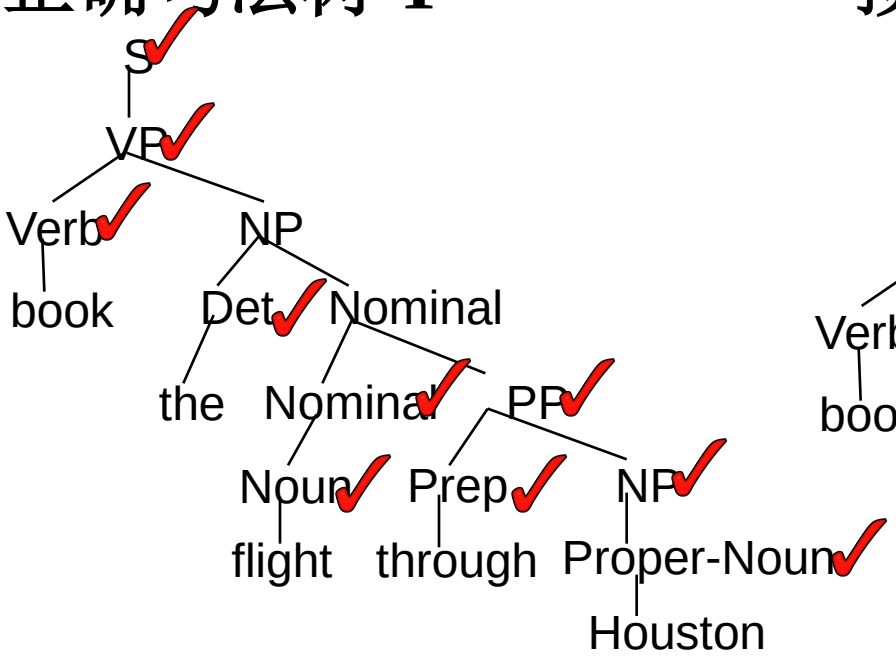
- 假设 P 是系统输出的句法树，假设 T 是标注好的句法树:
  - 召回率(Recall) = (#P中正确的元素) / (#T中正确的元素)
  - 准确率(Precision) = (#P中正确的元素) / (#P中总的元素)

- F值(F-score, F1)是召回率和准确率之间的调和平均数 (harmonic mean)

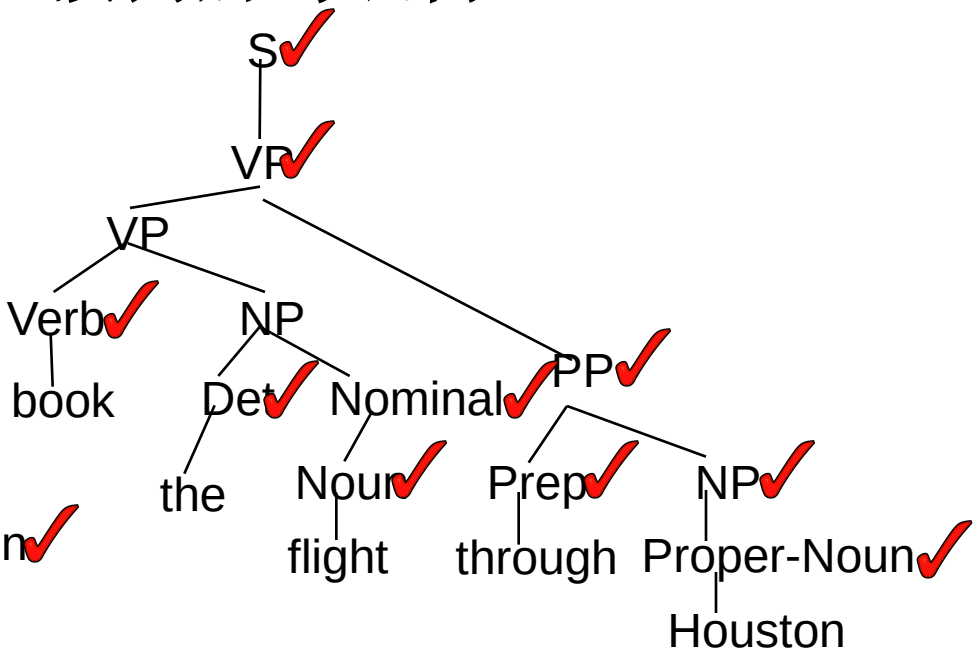
目前好的句法分析系统在标准数据集上可以达到90%以上的Precision, Recall, F-score

句法分析的效果打分

正确句法树 T



预测的句法树P



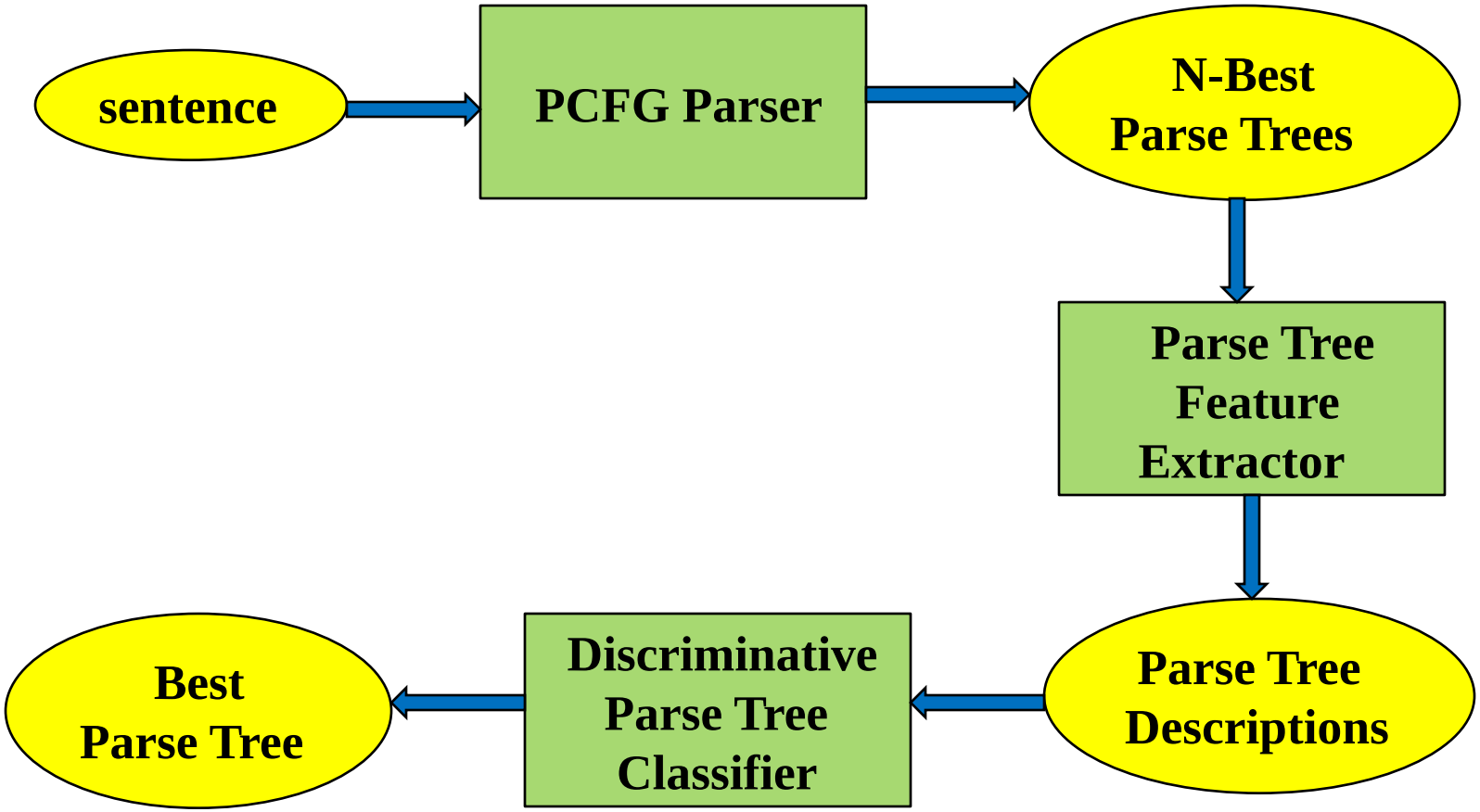
元素: 12

正确的元素: 10

元素: 12

Recall = 10/12= 83.3% Precision = 10/12=83.3% F_1 = 83.3%

基于重排序的句法分析(re-ranking parser)



概率句法分析总结

- 通过概率信息，消解句法分析中的歧义，得到最大概率的句法分析树等
- 通过标注好的树库，可以学习到概率句法分析器
- 现有的概率句法分析技术已经有很高的准确度

进一步阅读/学习

- **参考书**
  - 《统计自然语言处理》第8章：句法分析
    - 8.1 句法结构分析概述
    - Page 179 – 184
    - 8.2 基于PCFG的基本分析方法
    - Page 184 - 192