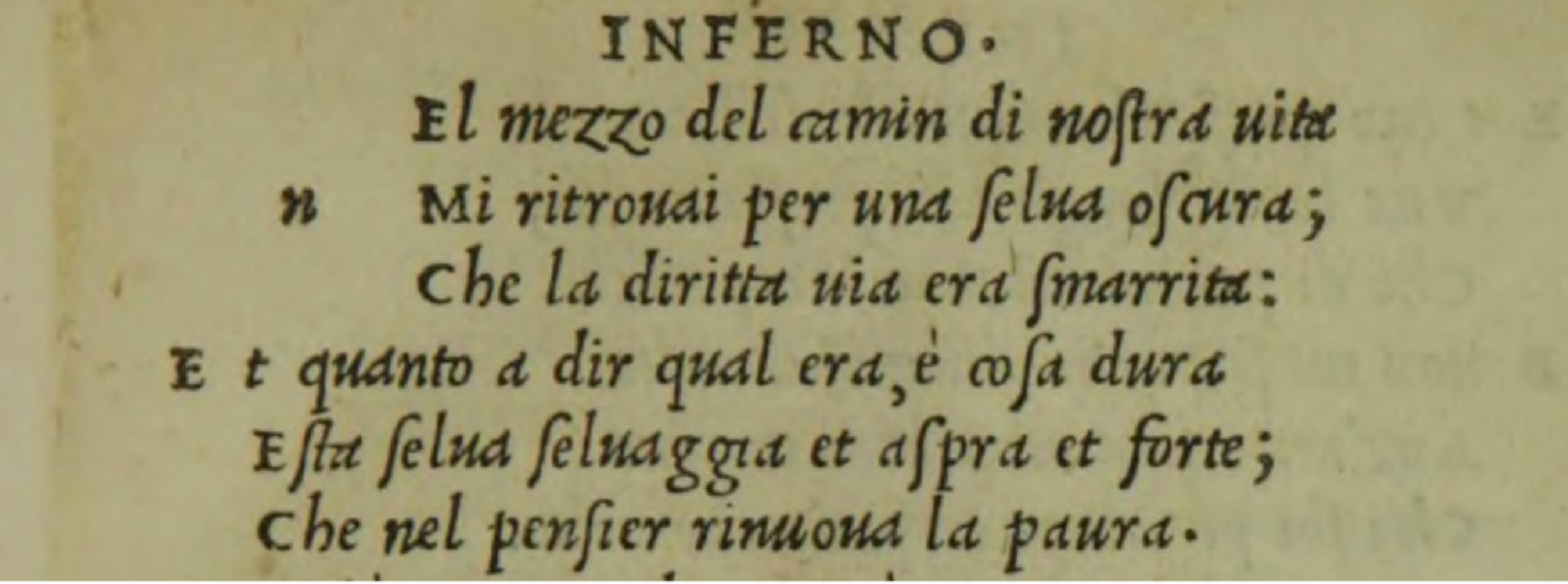

INFERNO.
El mezzo del camin di noſtra uita
n Mi ritrouai per una ſelua oſcura;
Che la diritta uia era ſmarrita:
E t quanto a dir qual era, è coſa dura
Eſta ſelua ſeluaggia et aſpra et forte;
Che nel penſier rinuoua la paura.

Natural Language Processing

Info 159/259

Lecture 14: Phrase-structure parsing (Mar 5, 2020)

David Bamman, UC Berkeley

Context-free grammar

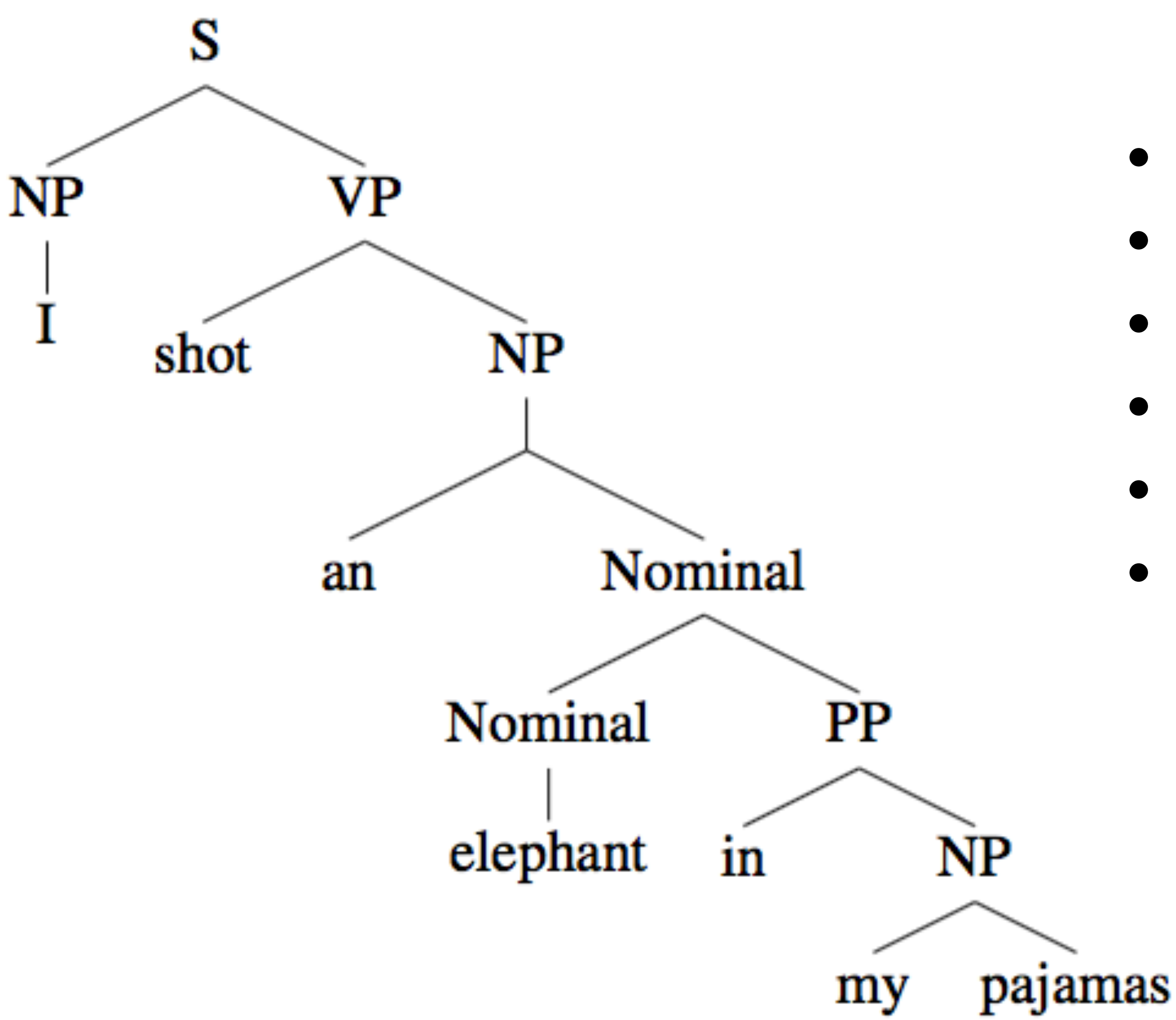
A context-free grammar defines how symbols in a language combine to form valid structures

NP	→	Det Nominal
NP	→	ProperNoun
Nominal	→	Noun | Nominal Noun
Det	→	a | the
Noun	→	flight

non-terminals

lexicon/
terminals

Constituents



Every internal node is a phrase

- my pajamas
- in my pajamas
- elephant in my pajamas
- an elephant in my pajamas
- shot an elephant in my pajamas
- I shot an elephant in my pajamas

Each phrase could be replaced by another of the same type of constituent

PCFG

- Probabilistic context-free grammar: each production is also associated with a probability.

- This lets us calculate the probability of a parse for a given sentence; for a given parse tree T for sentence S comprised of n rules from R (each A → β):

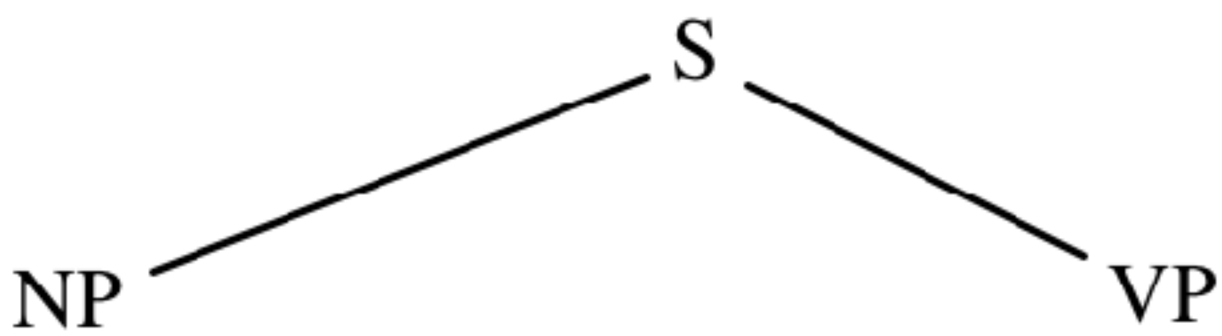
$$P(T, S) = \prod_{i}^{n} P(\beta \mid A)$$

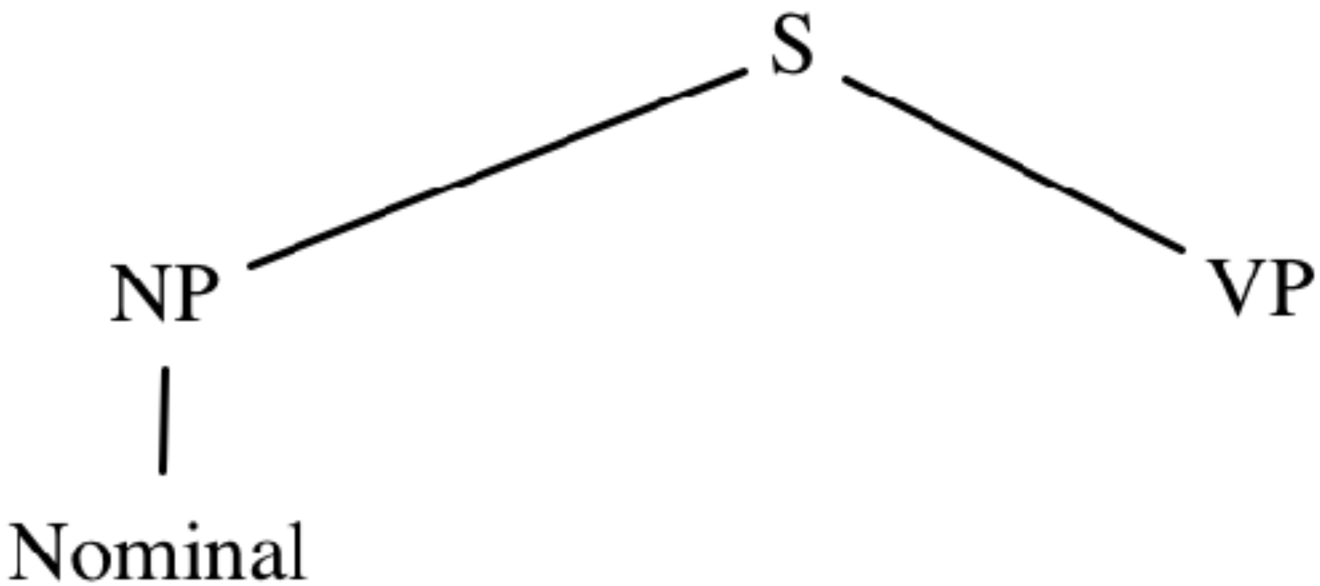
PCFG

N	Finite set of non-terminal symbols	NP, VP, S
Σ	Finite alphabet of terminal symbols	the, dog, a
R	Set of production rules, each $A \rightarrow \beta$ [p] $p = P(\beta \mid A)$	$S \rightarrow NP\ VP$ $Noun \rightarrow dog$
S	Start symbol	

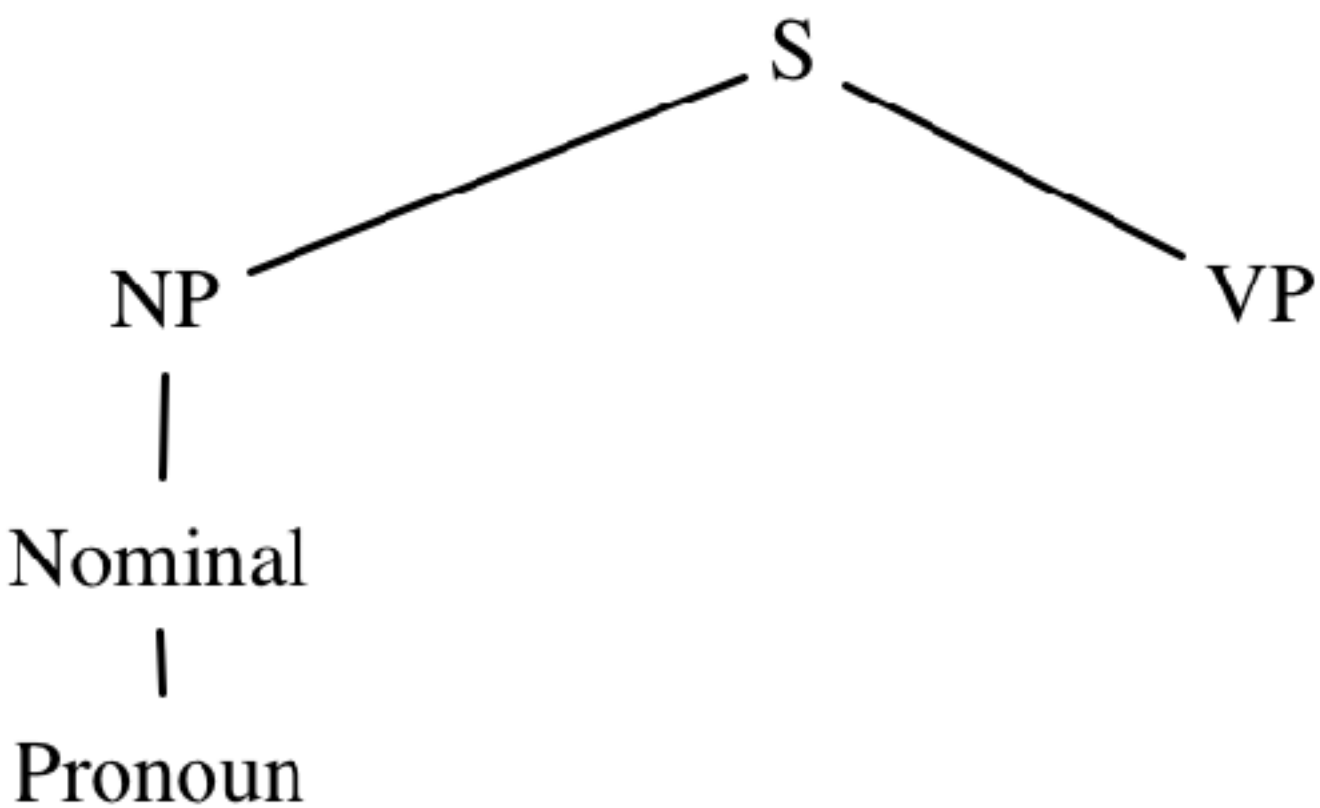
S

$P(\mathrm{NP\ VP} \mid \mathrm{S})$

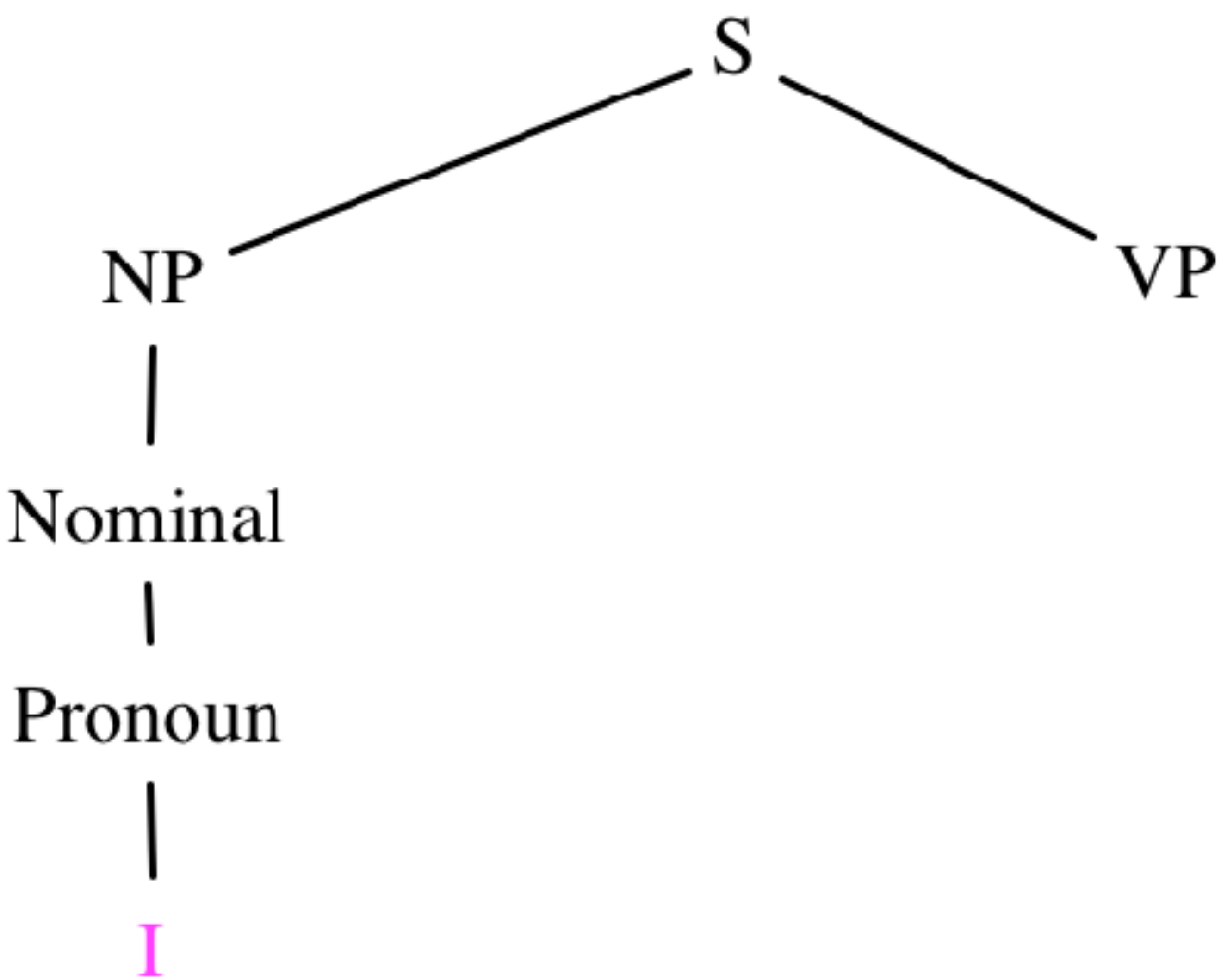




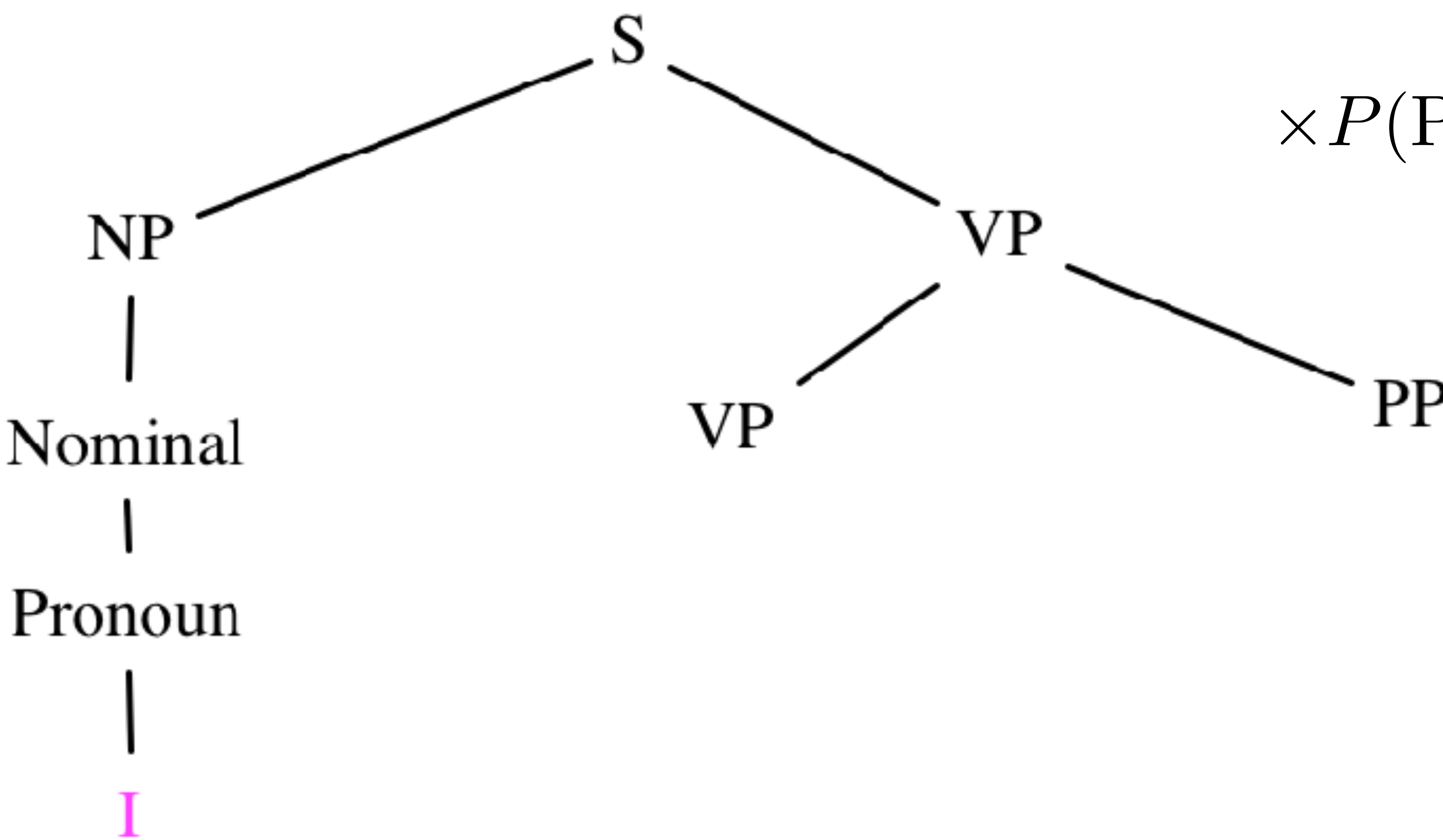
$$P(\text{NP VP} \mid \text{S})$$
$$\times P(\text{Nominal} \mid \text{NP})$$



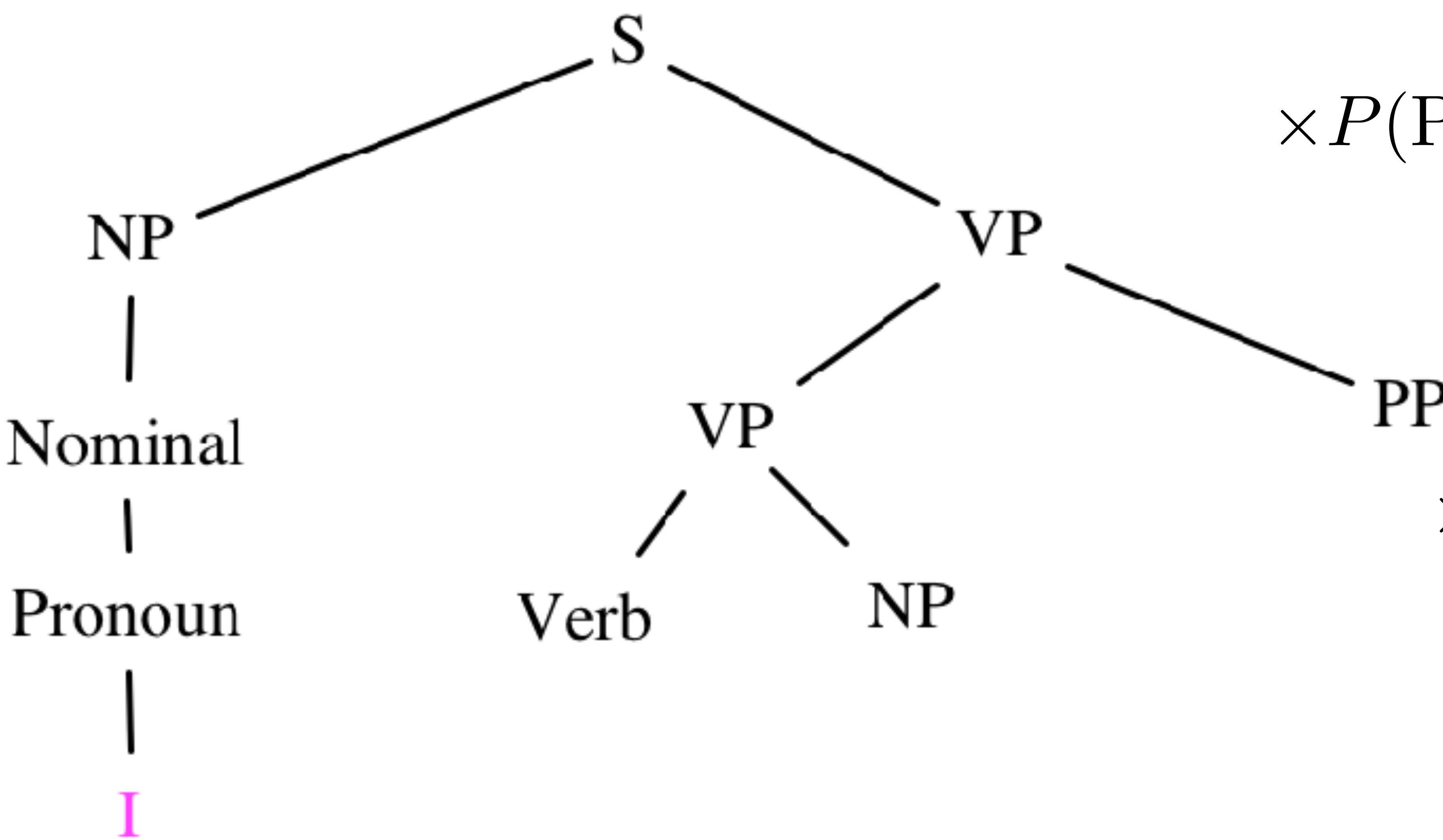
$$P(\text{NP VP} \mid \text{S}) \\ \times P(\text{Nominal} \mid \text{NP}) \\ \times P(\text{Pronoun} \mid \text{Nominal})$$



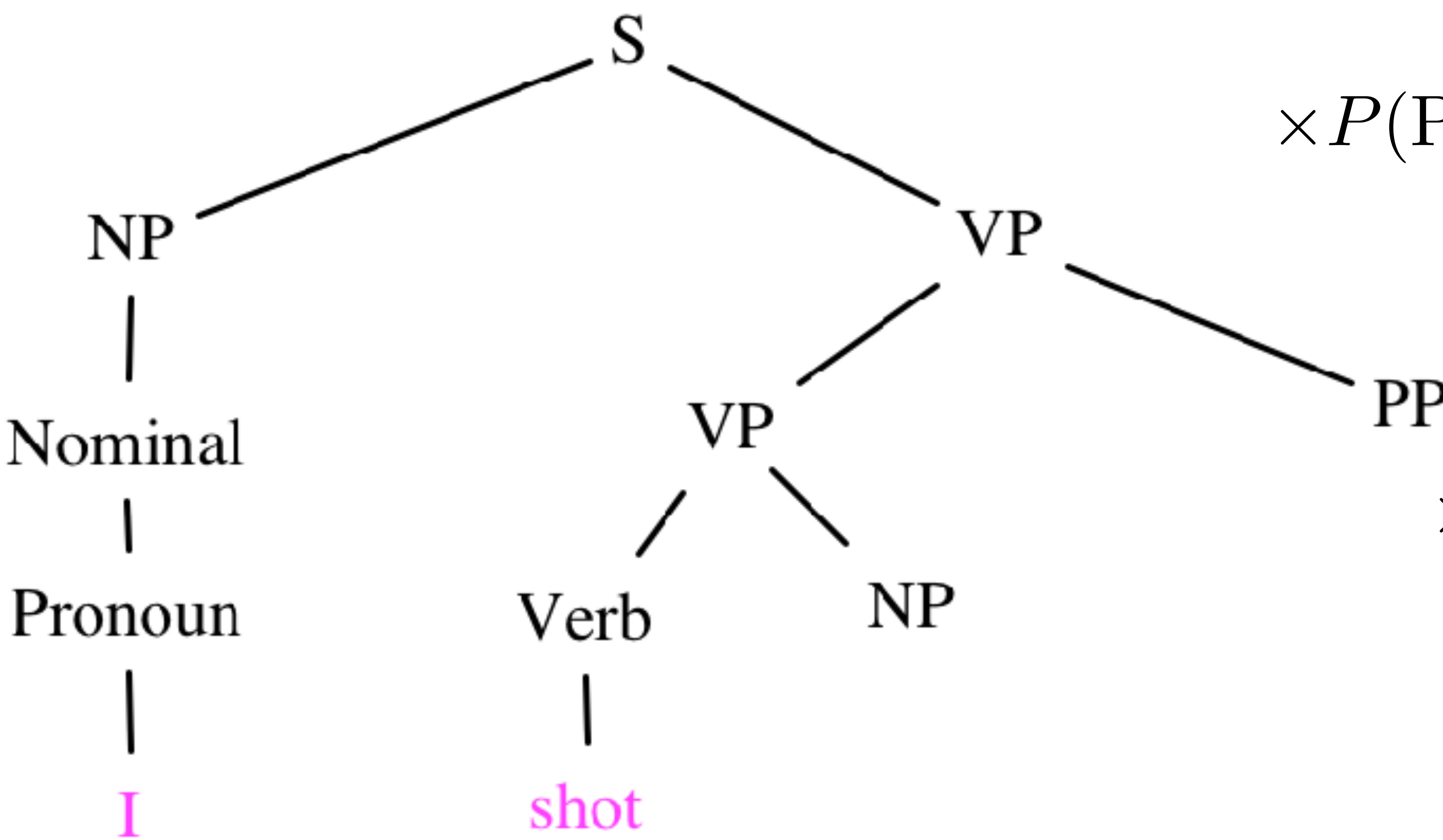
$$
\begin{aligned}
&P(\text{NP VP} \mid \text{S}) \\
&\times P(\text{Nominal} \mid \text{NP}) \\
&\times P(\text{Pronoun} \mid \text{Nominal}) \\
&\times P(\text{I} \mid \text{Pronoun})
\end{aligned}
$$



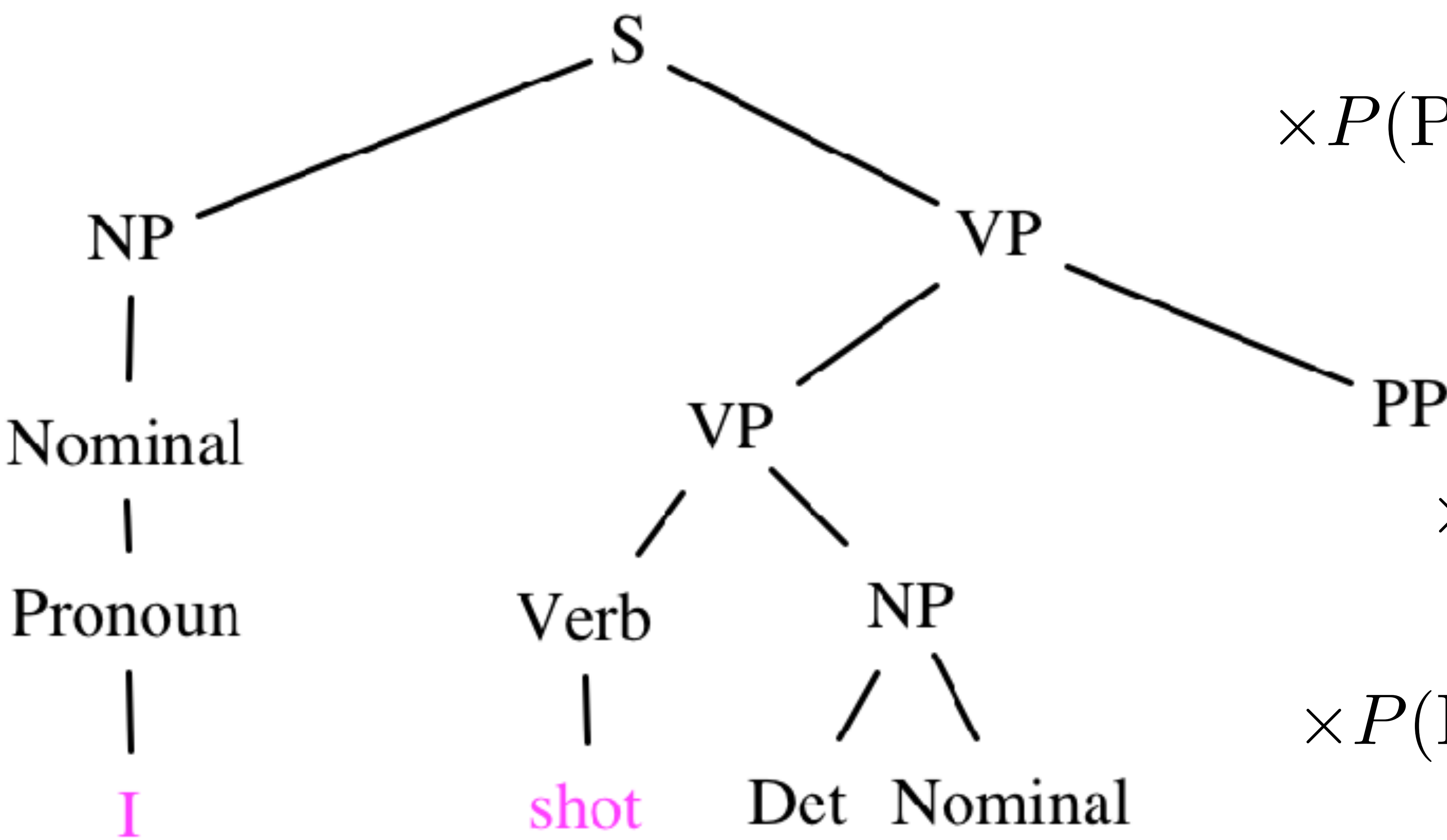
$$
\begin{aligned}
&P(\text{NP VP} \mid \text{S}) \\
&\times P(\text{Nominal} \mid \text{NP}) \\
&\times P(\text{Pronoun} \mid \text{Nominal}) \\
&\times P(\text{I} \mid \text{Pronoun}) \\
&\times P(\text{VP PP} \mid \text{VP})
\end{aligned}
$$



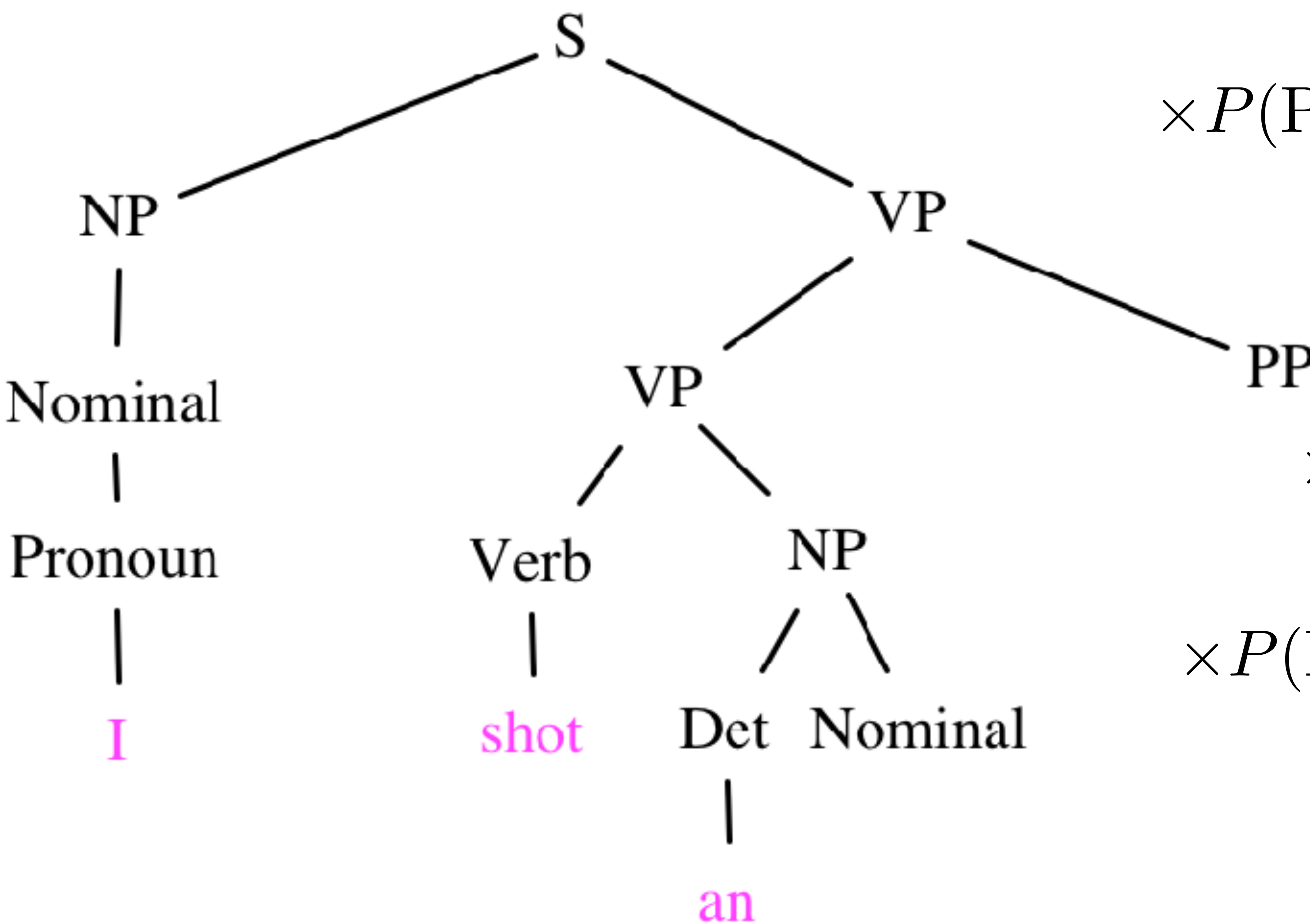
$$P(\text{NP VP} \mid \text{S})$$
$$\times P(\text{Nominal} \mid \text{NP})$$
$$\times P(\text{Pronoun} \mid \text{Nominal})$$
$$\times P(\text{I} \mid \text{Pronoun})$$
$$\times P(\text{VP PP} \mid \text{VP})$$
$$\times P(\text{Verb NP} \mid \text{VP})$$



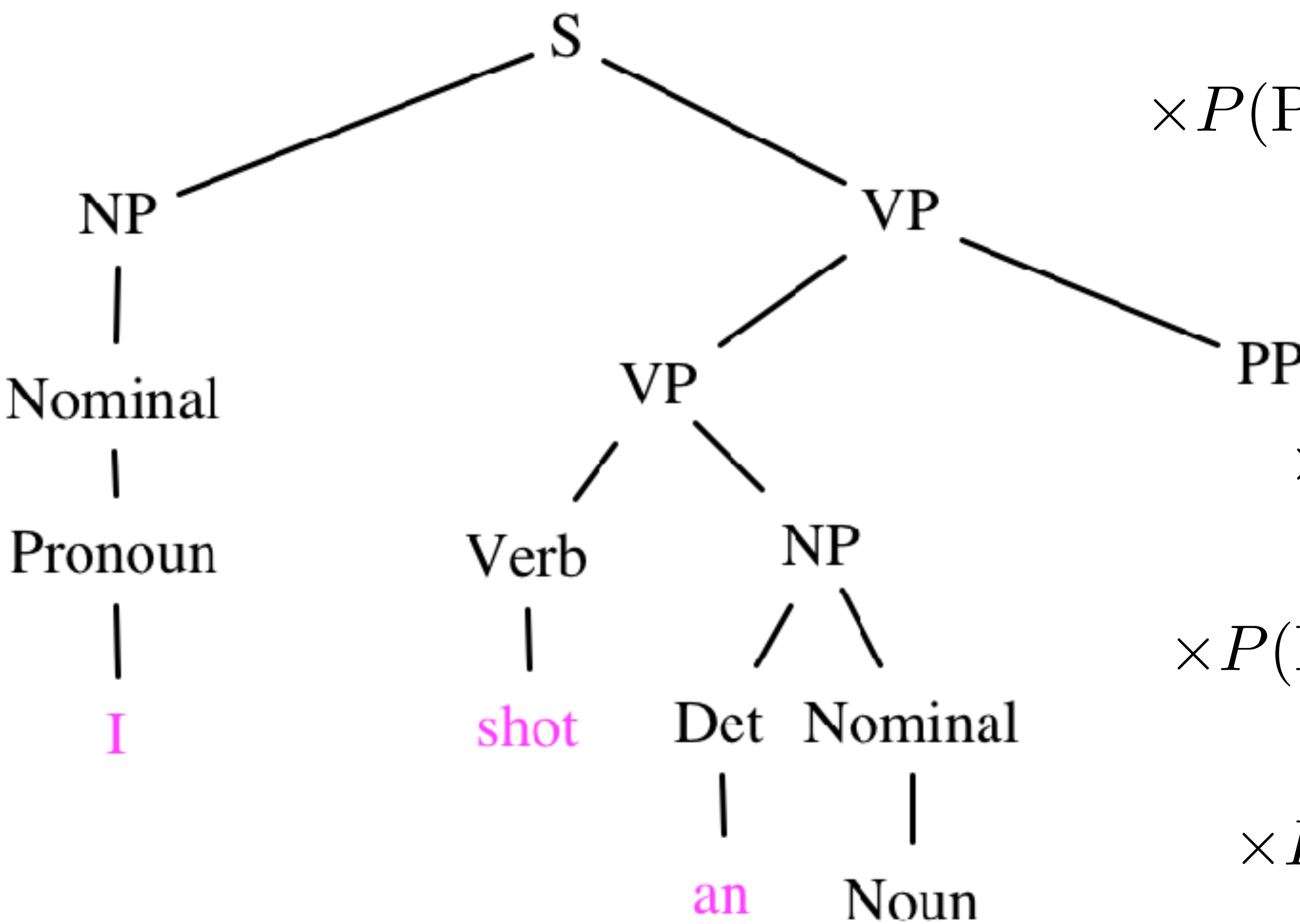
$$P(\text{NP VP} \mid \text{S})$$
$$\times P(\text{Nominal} \mid \text{NP})$$
$$\times P(\text{Pronoun} \mid \text{Nominal})$$
$$\times P(\text{I} \mid \text{Pronoun})$$
$$\times P(\text{VP PP} \mid \text{VP})$$
$$\times P(\text{Verb NP} \mid \text{VP})$$
$$\times P(\text{shot} \mid \text{Verb})$$



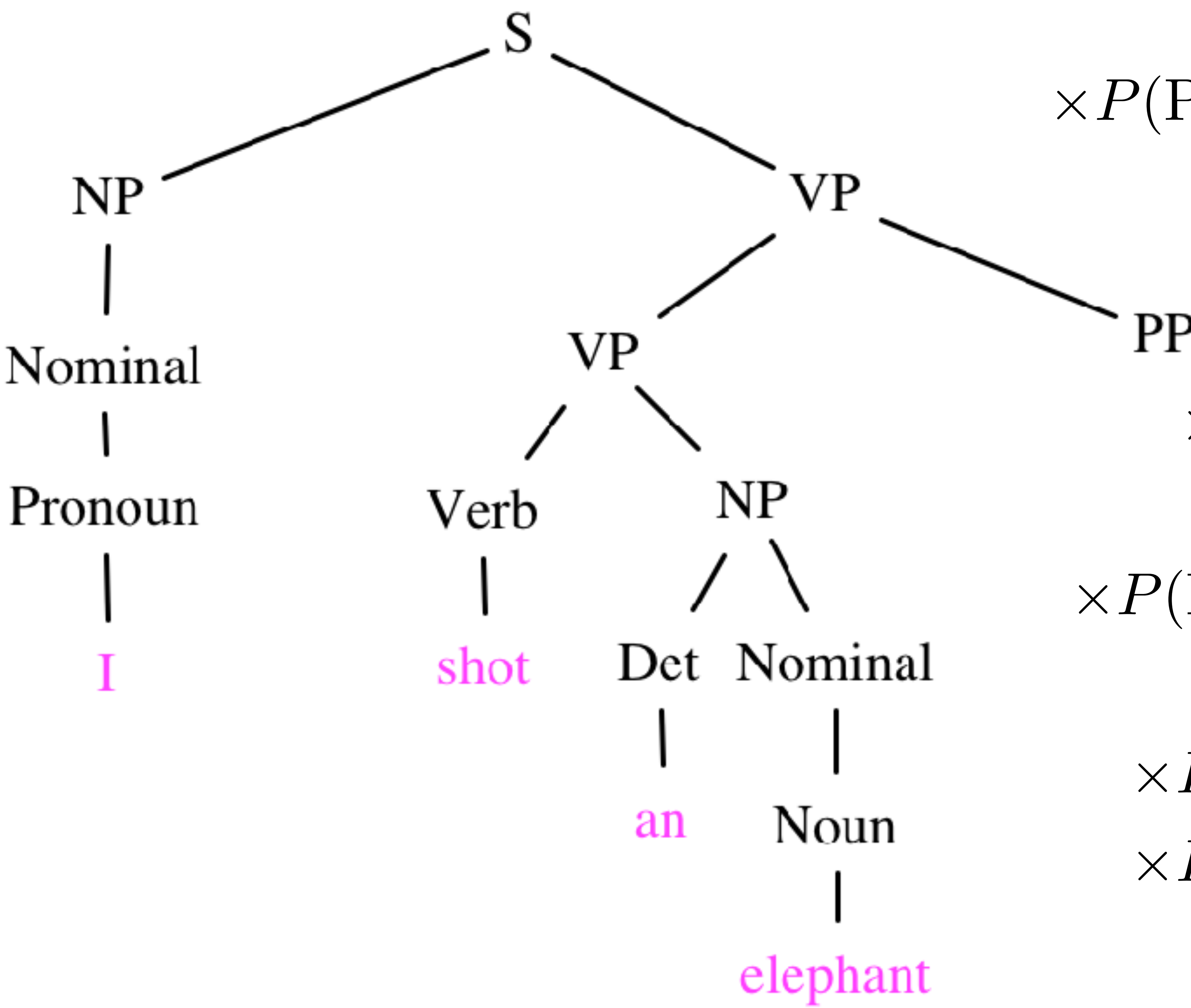
$$
\begin{aligned}
&P(\text{NP VP} \mid \text{S}) \\
&\times P(\text{Nominal} \mid \text{NP}) \\
&\times P(\text{Pronoun} \mid \text{Nominal}) \\
&\times P(\text{I} \mid \text{Pronoun}) \\
&\times P(\text{VP PP} \mid \text{VP}) \\
&\times P(\text{Verb NP} \mid \text{VP}) \\
&\times P(\text{shot} \mid \text{Verb}) \\
&\times P(\text{Det Nominal} \mid \text{NP})
\end{aligned}
$$



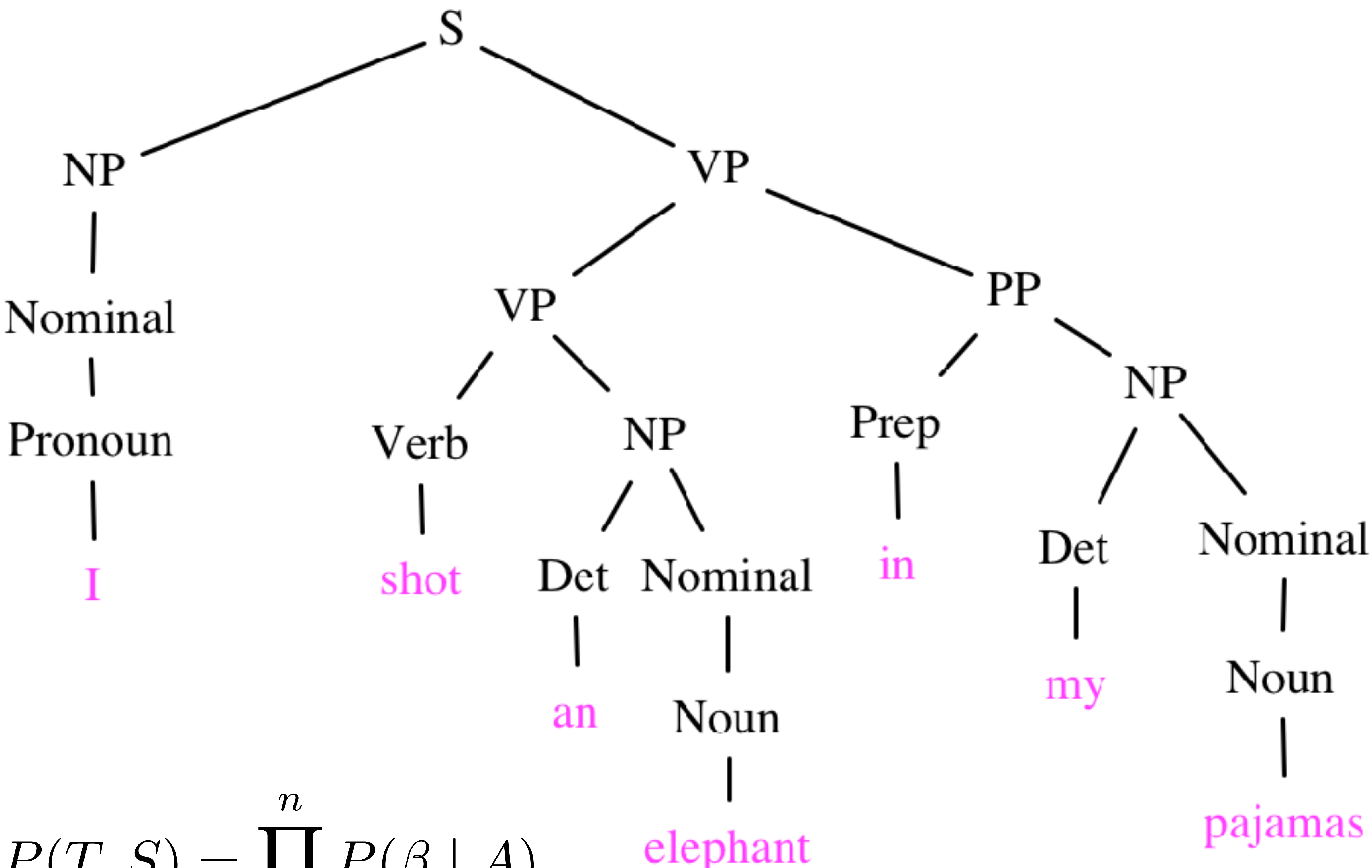
$$P(\text{NP VP} \mid \text{S})$$
$$\times P(\text{Nominal} \mid \text{NP})$$
$$\times P(\text{Pronoun} \mid \text{Nominal})$$
$$\times P(\text{I} \mid \text{Pronoun})$$
$$\times P(\text{VP PP} \mid \text{VP})$$
$$\times P(\text{Verb NP} \mid \text{VP})$$
$$\times P(\text{shot} \mid \text{Verb})$$
$$\times P(\text{Det Nominal} \mid \text{NP})$$
$$\times P(\text{an} \mid \text{Det})$$



$$P(\text{NP VP} \mid \text{S})$$
$$\times P(\text{Nominal} \mid \text{NP})$$
$$\times P(\text{Pronoun} \mid \text{Nominal})$$
$$\times P(\text{I} \mid \text{Pronoun})$$
$$\times P(\text{VP PP} \mid \text{VP})$$
$$\times P(\text{Verb NP} \mid \text{VP})$$
$$\times P(\text{shot} \mid \text{Verb})$$
$$\times P(\text{Det Nominal} \mid \text{NP})$$
$$\times P(\text{an} \mid \text{Det})$$
$$\times P(\text{Noun} \mid \text{Nominal})$$



$$
\begin{aligned}
&P(\text{NP VP} \mid \text{S})\\
&\times P(\text{Nominal} \mid \text{NP})\\
&\times P(\text{Pronoun} \mid \text{Nominal})\\
&\times P(\text{I} \mid \text{Pronoun})\\
&\times P(\text{VP PP} \mid \text{VP})\\
&\times P(\text{Verb NP} \mid \text{VP})\\
&\times P(\text{shot} \mid \text{Verb})\\
&\times P(\text{Det Nominal} \mid \text{NP})\\
&\times P(\text{an} \mid \text{Det})\\
&\times P(\text{Noun} \mid \text{Nominal})\\
&\times P(\text{elephant} \mid \text{Noun})
\end{aligned}
$$



$$P(T, S) = \prod_{i}^{n} P(\beta \mid A)$$

PCFGs

- A PCFG gives us a mechanism for assigning scores (here, probabilities) to different parses for the same sentence.

- But we often care about is finding the single best parse with the highest probability.

Context-free grammar

N	Finite set of non-terminal symbols	NP, VP, S
Σ	Finite alphabet of terminal symbols	the, dog, a
R	Set of production rules, each $A \rightarrow \beta$ $\beta \in (\Sigma, N)$	NP → DT JJ NN Noun → dog
S	Start symbol	

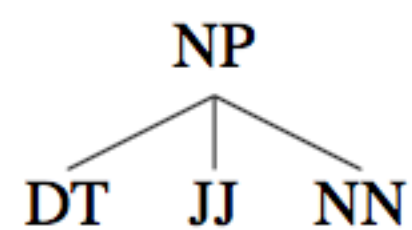
Chomsky Normal Form (CNF)

N	Finite set of non-terminal symbols	NP, VP, S
Σ	Finite alphabet of terminal symbols	the, dog, a
R	Set of production rules, each $A \rightarrow \beta$ β = single terminal (from Σ) or two non-terminals (from N)	$S \rightarrow NP\ VP$ $Noun \rightarrow dog$
S	Start symbol	

Chomsky Normal Form (CNF)

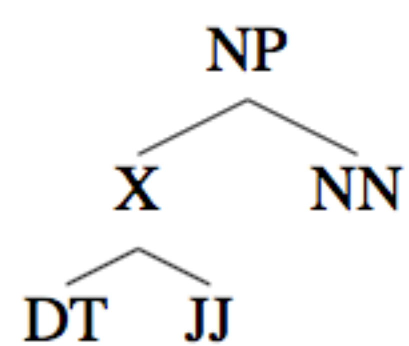
- Any CFG can be converted into weakly equivalent CNF (recognizing the same set of sentences as existing in the grammar but differing in their derivation).

NP → DT JJ NN



NP → X NN

X → DT JJ



S	→	NP VP
VP	→	VBD NP
VP	→	VP PP
Nominal	→	Nominal PP
Nominal	→	NN
Nominal	→	NNS
Nominal	→	PRP
PP	→	IN NP
NP	→	DT NN
NP	→	Nominal
NP	→	PRP$ Nominal

VBD	→	shot
DT	→	an | my
NN	→	elephant
NNS	→	pajamas
PRP	→	I
PRP$	→	my
IN	→	in

I shot an elephant in my pajamas

S	→	NP VP
VP	→	VBD NP
VP	→	VP PP
Nominal	→	Nominal PP
Nominal	→	pajamas | elephant | I
PP	→	IN NP
NP	→	DT NN
NP	→	pajamas | elephant | I
NP	→	PRP$ Nominal

VBD	→	shot
DT	→	an | my
PRP	→	I
PRP$	→	my
IN	→	in

I shot an elephant in my pajamas

CKY

- Cocke-Kasami-Younger algorithm (also CYK) for parsing from a grammar expressed in CNF.
    - Kasami (1965)
    - Younger (1967)
    - Cocke and Schwartz (1970)

- Bottom-up dynamic programming: once we discover a constituent, we can make it available for any rule that needs it.

0	I	1	shot	2	an	3	elephant	4	in	5	my	6	pajamas	7

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]						
	VBD [1,2]					
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Each cell i,j keeps track of all phrase types that can be formed from *all* words from position i through position j

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]						
	VBD [1,2]					
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

What phrases can be formed from “shot an elephant in”

I	shot	an	elephant	in	my	pajamas

NP, PRP [0,1]						
	VBD [1,2]					
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

What phrases can be formed from “I shot an elephant in my pajamas”

CNF

- In CNF, each non-terminal generates two non-terminals

$$S \rightarrow NP\ VP$$

$[_S\ [_{NP}\ \text{I}]\ [_{VP}\ \text{shot an elephant in my pajamas}]\ ]$

- If the left-side non-terminal (S) spans tokens i-j, the right side (NP VP) must also span i-j, and there must be a single position k that separates them.

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]						
	VBD [1,2]					
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Does any rule generate PRP VBD?

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅					
	VBD [1,2]					
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Does any rule generate VBD DT?

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅					
	VBD [1,2]	∅				
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Two possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅					
	VBD [1,2]	∅				
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Two possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅					
	VBD [1,2]	∅				
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Two possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅				
	VBD [1,2]	∅				
		DT [2,3]				
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Does any rule generate DT NN?

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅				
	VBD [1,2]	∅				
		DT [2,3]	NP [2,4]			
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

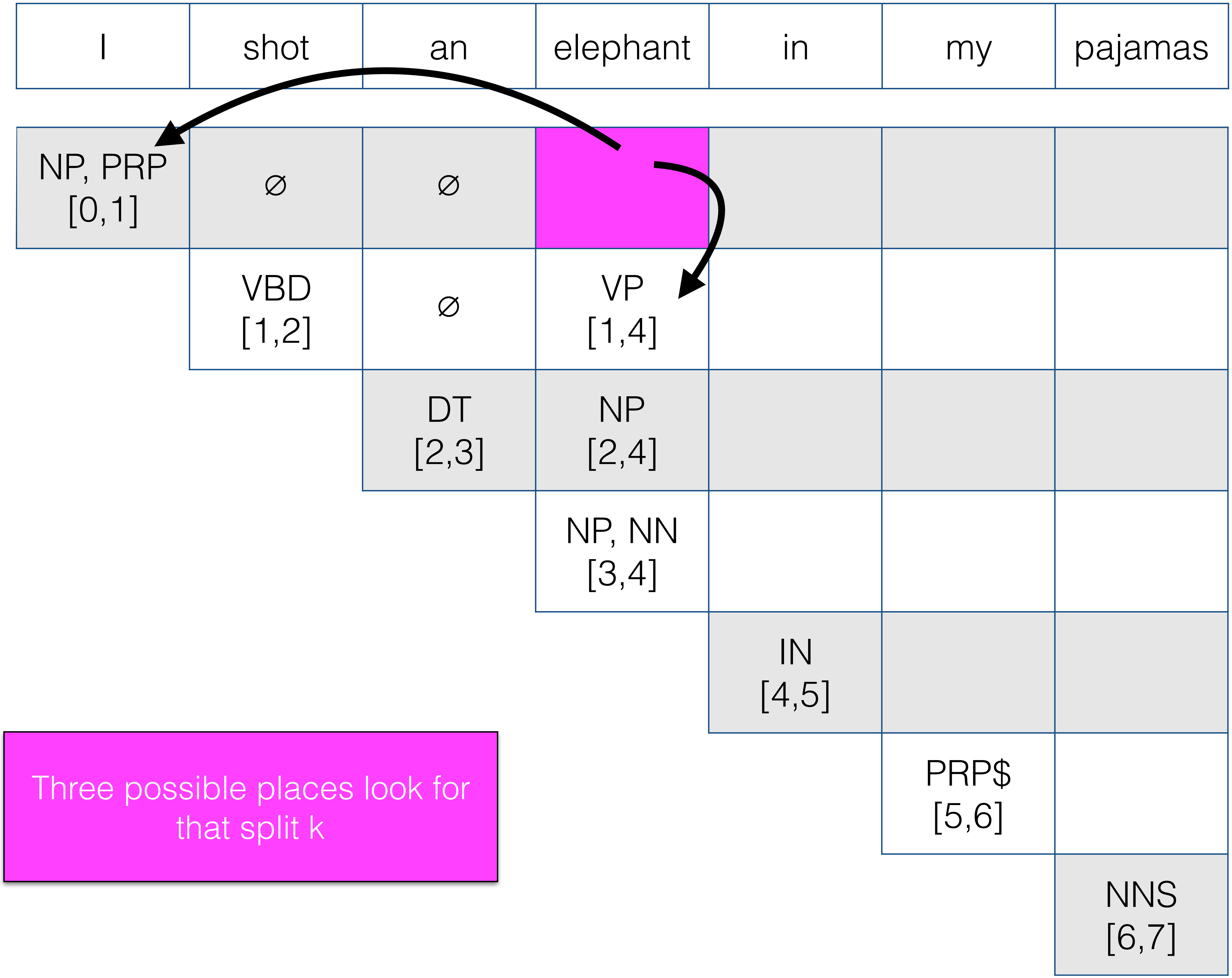
Two possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅				
	VBD [1,2]	∅				
		DT [2,3]	NP [2,4]			
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Two possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅				
	VBD [1,2]	∅	VP [1,4]			
		DT [2,3]	NP [2,4]			
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Three possible places look for that split k


I
shot
an
elephant
in
my
pajamas
NP, PRP [0,1]
∅
∅
VBD [1,2]
∅
VP [1,4]
DT [2,3]
NP [2,4]
NP, NN [3,4]
IN [4,5]
PRP$ [5,6]
NNS [6,7]
Three possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅				
	VBD [1,2]	∅	VP [1,4]			
		DT [2,3]	NP [2,4]			
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Three possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅				
	VBD [1,2]	∅	VP [1,4]			
		DT [2,3]	NP [2,4]			
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

Three possible places look for that split k

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]			
	VBD [1,2]	∅	VP [1,4]			
		DT [2,3]	NP [2,4]			
			NP, NN [3,4]			
				IN [4,5]		
					PRP$ [5,6]	
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	
			NP, NN [3,4]	∅	∅	
				IN [4,5]	∅	
					PRP$ [5,6]	
						NNS [6,7]

*elephant in
*an elephant in
*shot an elephant in
*I shot an elephant in

*in my
*elephant in my
*an elephant in my
*shot an elephant in my
*I shot an elephant in my

I	shot	an	elephant	in	my	pajamas

NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	
			NP, NN [3,4]	∅	∅	
				IN [4,5]	∅	
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	
			NP, NN [3,4]	∅	∅	
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	
		DT [2,3]	NP [2,4]	∅	∅	NP [3,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas

NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	VP_1, VP_2 [1,7]
		DT [2,3]	NP [2,4]	∅	∅	NP [2,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	
	VBD [1,2]	∅	VP [1,4]	∅	∅	VP_1, VP_2 [1,7]
		DT [2,3]	NP [2,4]	∅	∅	NP [2,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

Possibilities:

$S_1 \rightarrow NP\ VP_1$
$S_2 \rightarrow NP\ VP_2$
$? \rightarrow S\ PP$
$? \rightarrow PRP\ VP_1$
$? \rightarrow PRP\ VP_2$

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	S_1, S_2 [0,7]
	VBD [1,2]	∅	VP [1,4]	∅	∅	VP_1, VP_2 [1,7]
		DT [2,3]	NP [2,4]	∅	∅	NP [2,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

Success! We've recognized a total of two valid parses

CKY algorithm

function CKY-PARSE(*words, grammar*) **returns** *table*

  for $j \leftarrow$ **from** 1 **to** LENGTH(*words*) **do**
    for all $\{A \mid A \rightarrow words[j] \in grammar\}$
      $table[j-1, j] \leftarrow table[j-1, j] \cup A$
    for $i \leftarrow$ **from** $j-2$ **downto** 0 **do**
      for $k \leftarrow i+1$ **to** $j-1$ **do**
        for all $\{A \mid A \rightarrow BC \in grammar$ **and** $B \in table[i,k]$ **and** $C \in table[k,j]\}$
          $table[i,j] \leftarrow table[i,j] \cup A$

Figure 12.5 The CKY algorithm.

I	shot	an	elephant	in	my	pajamas
NP, PRP [0,1]	∅	∅	S [0,4]	∅	∅	S_1, S_2 [0,7]
	VBD [1,2]	∅	VP [1,4]	∅	∅	VP_1, VP_2 [1,7]
		DT [2,3]	NP [2,4]	∅	∅	NP [2,7]
			NP, NN [3,4]	∅	∅	NP [3,7]
				IN [4,5]	∅	PP [4,7]
					PRP$ [5,6]	NP [5,7]
						NNS [6,7]

Runtime complexity?

CFG

- This use of CKY allows us to:
    - check whether a sentence in grammatical in the language defined by the CFG
    - enumerate all possible parses for a sentence
- But it doesn’t tell us on its one which of those possible parses is most likely.

PCFGs

- A PCFG gives us a mechanism for assigning scores (here, probabilities) to different parses for the same sentence.
- But we often care about is finding the single best parse with the highest probability.
- We calculate the max probability parse using CKY by storing the probability of each phrase within each cell as we build it up.

$$table(i, j, A) = P(A \to BC) \times table(i, k, B) \times table(k, j, C)$$

I	shot	an	elephant	in	my	pajamas
PRP:0.04 [0,1]						
	VBD:0.04 [1,2]					
		DT:0.05 [2,3]				
			NN:0.03 [3,4]			
				IN:0.10 [4,5]		
					PRP$: 0.12 [5,6]	
						NNS:0.01 [6,7]

Probability of a terminal (word) given its tag

$$P(A \rightarrow \beta)$$

I	shot	an	elephant	in	my	pajamas
PRP:0.04 [0,1]	∅	∅				
	VBD:0.04 [1,2]	∅				
		DT:0.05 [2,3]	NP: 0.00015 [2,4]			
			NN:0.03 [3,4]			
				IN:0.10 [4,5]		
					PRP$:0.12 [5,6]	
						NNS:0.01 [6,7]

$$table(2,4,NP) = P(\text{NP} \rightarrow \text{DT NN}) \times table(2,3,DT) \times table(3,4,NN)$$

I	shot	an	elephant	in	my	pajamas
PRP:0.04 [0,1]	∅	∅				
	VBD:0.04 [1,2]	∅	VP: 0.0000006 [1,4]			
		DT:0.05 [2,3]	NP: 0.00015 [2,4]			
			NN:0.03 [3,4]			
				IN:0.10 [4,5]		
					PRP$:0.12 [5,6]	
						NNS:0.01 [6,7]

We just calculated this value and can use it now

$$table(1,4,VP) = P(\text{VP} \rightarrow \text{VBD NP}) \times table(1,2,VBD) \times table(2,4,NP)$$

I	shot	an	elephant	in	my	pajamas
PRP:0.04 [0,1]	∅	∅	S: 0.0000000048 [0,4]			
	VBD:0.04 [1,2]	∅	VP: 0.0000006 [1,4]			
		DT:0.05 [2,3]	NP: 0.00015 [2,4]			
			NN:0.03 [3,4]			
				IN:0.10 [4,5]		
					PRP$:0.12 [5,6]	
						NNS:0.01 [6,7]

We just calculated this value and can use it now

$$table(0,4,S) = P(\mathrm{S} \rightarrow \mathrm{NP\ VP}) \times table(0,1,NP) \times table(1,4,VP)$$

I	shot	an	elephant	in	my	pajamas
PRP:0.04 [0,1]	∅	∅	S: 0.0000000048 [0,4]			
	VBD:0.04 [1,2]	∅	VP: 0.0000006 [1,4]			
		DT:0.05 [2,3]	NP: 0.00015 [2,4]			
			NN:0.03 [3,4]			
				IN:0.10 [4,5]		
					PRP$:0.12 [5,6]	
						NNS:0.01 [6,7]

Note these values are getting very small! Better to add in log space

I	shot	an	elephant	in	my	pajamas
PRP: -3.21 [0,1]	∅	∅	S: -19.2 [0,4]			
	VBD: -3.21 [1,2]	∅	VP: -14.3 [1,4]			
		DT: -3.0 [2,3]	NP: -8.8 [2,4]			
			NN: -3.5 [3,4]			
				IN: -2.3 [4,5]		
					PRP$: -2.12 [5,6]	
						NNS: -4.6 [6,7]

Note these values are getting very small! Better to add in log space

I	shot	an	elephant	in	my	pajamas
PRP: -3.21 [0,1]	∅	∅	S: -19.2 [0,4]	∅	∅	
	VBD: -3.21 [1,2]	∅	VP: -14.3 [1,4]	∅	∅	VP_1, VP_2 [1,7]
		DT: -3.0 [2,3]	NP: -8.8 [2,4]	∅	∅	NP: -24.7 [2,7]
			NN: -3.5 [3,4]	∅	∅	NP: -19.4 [3,7]
				IN: -2.3 [4,5]	∅	PP: -13.6 [4,7]
					PRP$: -2.12 [5,6]	NP: -9.0 [5,7]
						NNS: -4.6 [6,7]

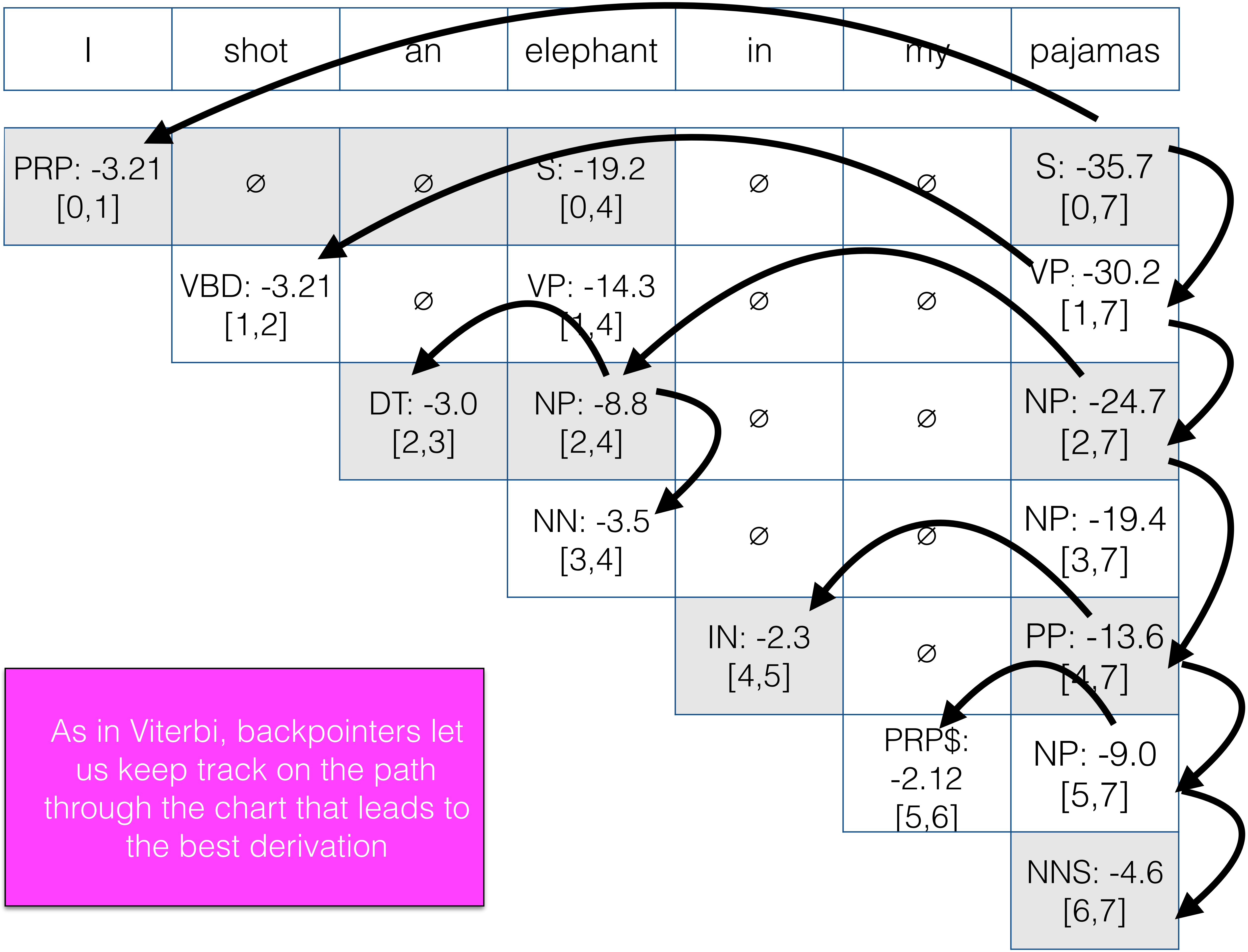
For any phrase type spanning [i,j], we only need to keep the max probability given the assumptions of a PCFG

I	shot	an	elephant	in	my	pajamas
PRP: -3.21 [0,1]	∅	∅	S: -19.2 [0,4]	∅	∅	
	VBD: -3.21 [1,2]	∅	VP: -14.3 [1,4]	∅	∅	VP: -30.2 [1,7]
		DT: -3.0 [2,3]	NP: -8.8 [2,4]	∅	∅	NP: -24.7 [2,7]
			NN: -3.5 [3,4]	∅	∅	NP: -19.4 [3,7]
				IN: -2.3 [4,5]	∅	PP: -13.6 [4,7]
					PRP$: -2.12 [5,6]	NP: -9.0 [5,7]
						NNS: -4.6 [6,7]

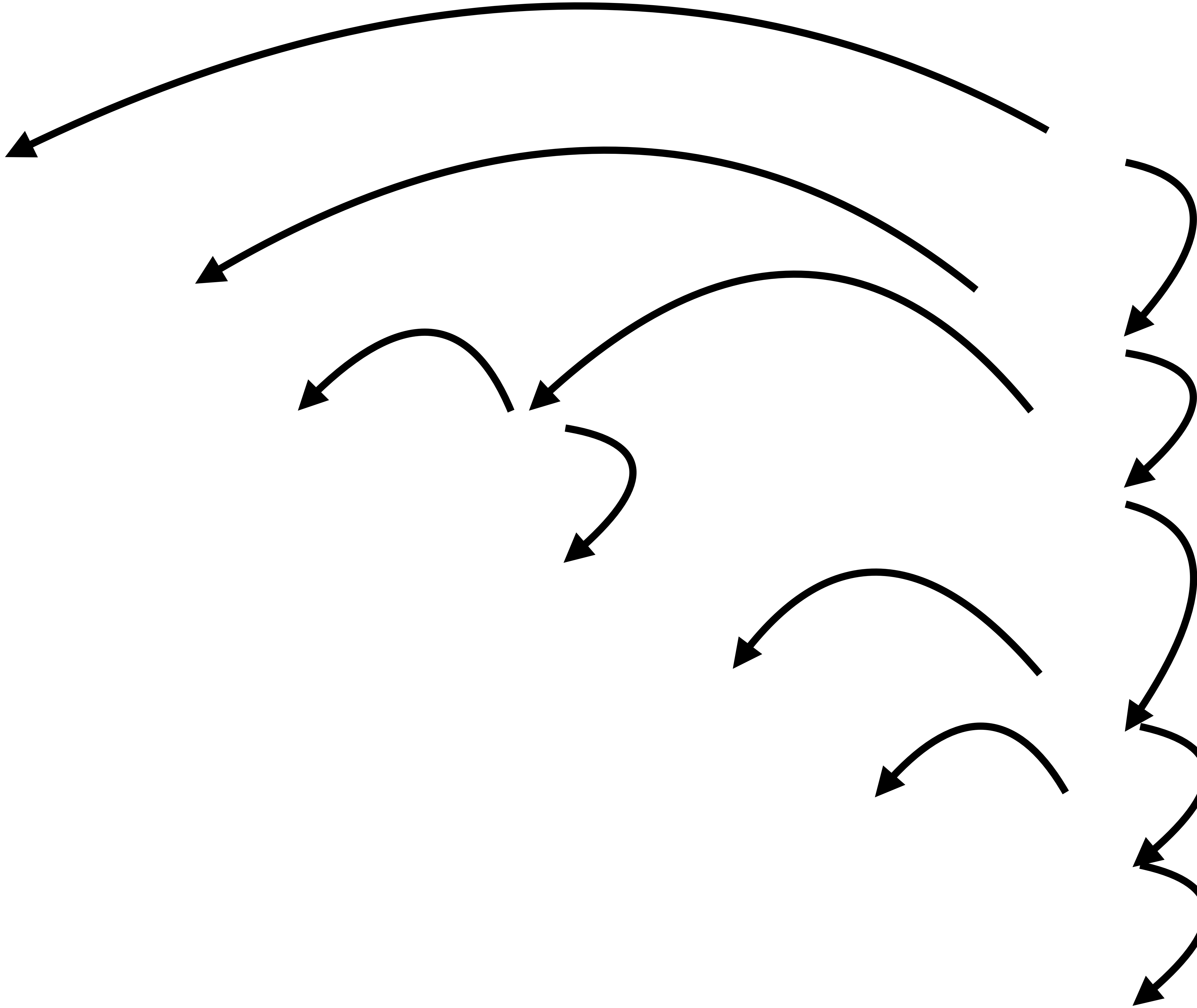
For any phrase type spanning [i,j], we only need to keep the max probability given the assumptions of a PCFG

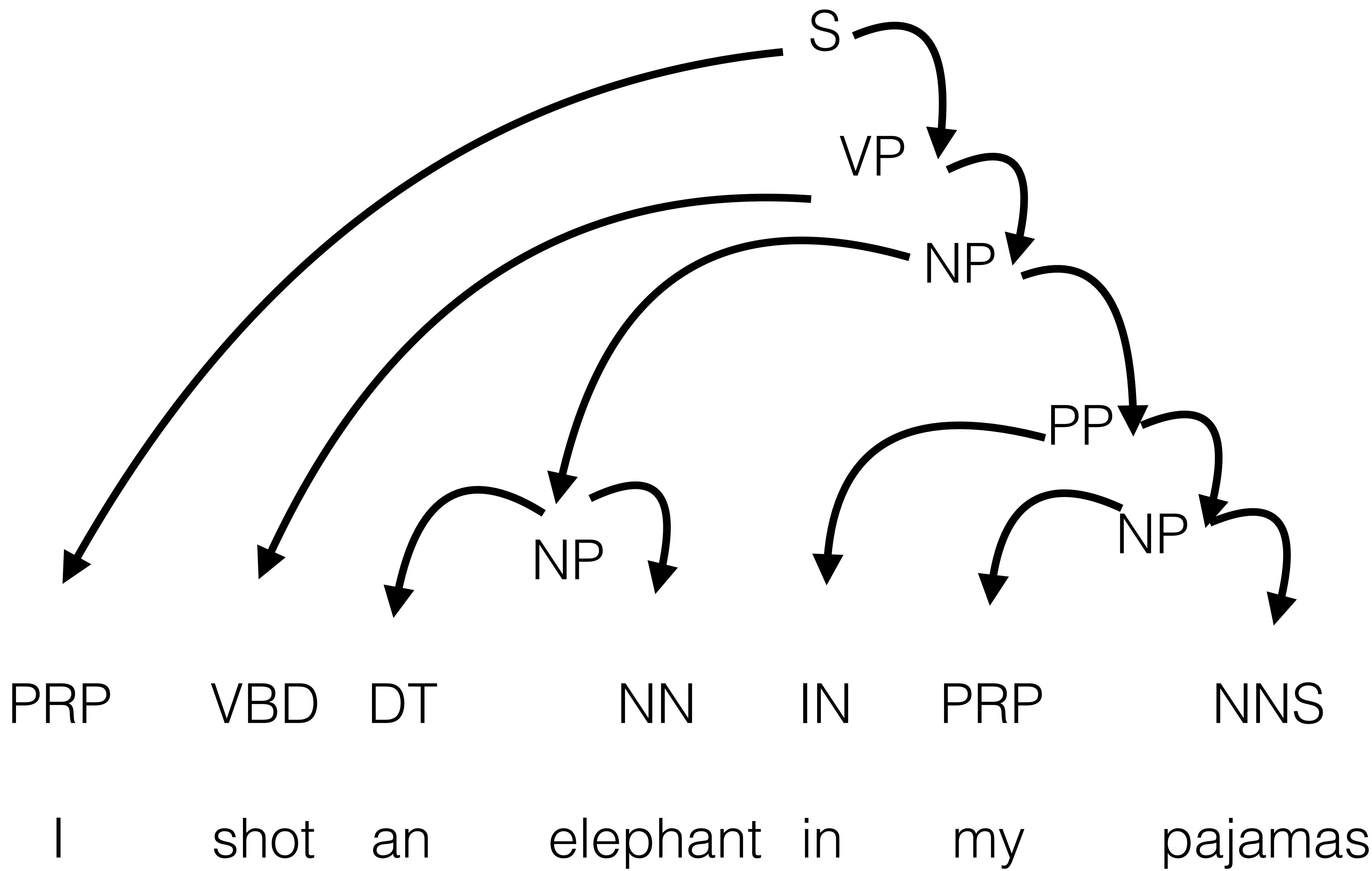
I	shot	an	elephant	in	my	pajamas
PRP: -3.21 [0,1]	∅	∅	S: -19.2 [0,4]	∅	∅	S: -35.7 [0,7]
	VBD: -3.21 [1,2]	∅	VP: -14.3 [1,4]	∅	∅	VP: -30.2 [1,7]
		DT: -3.0 [2,3]	NP: -8.8 [2,4]	∅	∅	NP: -24.7 [2,7]
			NN: -3.5 [3,4]	∅	∅	NP: -19.4 [3,7]
				IN: -2.3 [4,5]	∅	PP: -13.6 [4,7]
					PRP$: -2.12 [5,6]	NP: -9.0 [5,7]
						NNS: -4.6 [6,7]

For any phrase type spanning [i,j], we only need to keep the max probability given the assumptions of a PCFG



As in Viterbi, backpointers let us keep track on the path through the chart that leads to the best derivation




S
VP
NP
PP
NP
NP
PRP
VBD
DT
NN
IN
PRP
NNS
I
shot
an
elephant
in
my
pajamas

PCFG

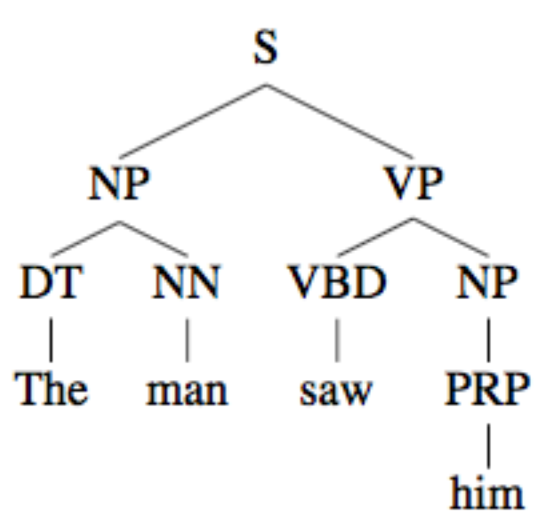
- Probabilistic context-free grammar: each production is also associated with a probability.

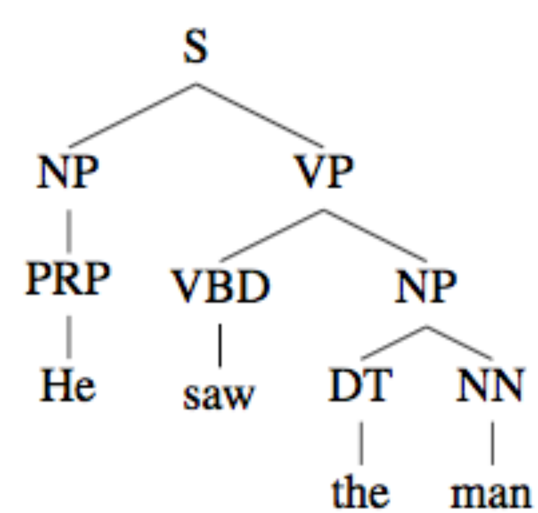
- This lets us calculate the probability of a parse for a given sentence; for a given parse tree T for sentence S comprised of n rules from R (each A → β):

$$P(T, S) = \prod_{i}^{n} P(\beta \mid A)$$

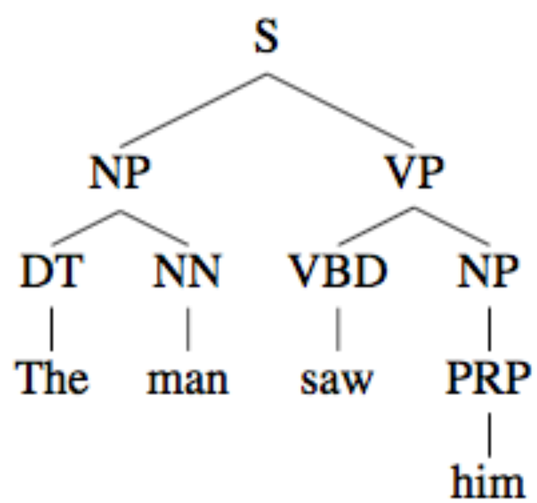
Problems with PCFGs

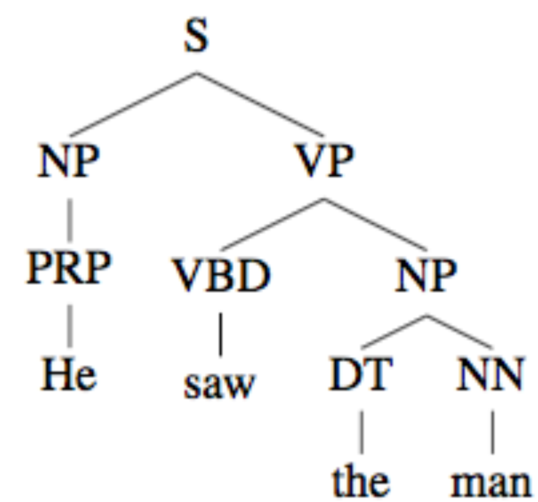
- Strong independence assumptions
  - Each production (e.g., NP → DT NN) is independent of the rest of tree.
  - In real use, productions are strongly dependent on their place in the tree.





Problems with PCFGs





	Pronoun	Non-Pronoun
Subject	91%	9%
Object	34%	66%

SLP3; Francis et al. 1999

Problems with PCFGs

	Pronoun	Non-Pronoun
Subject	91%	9%
Object	34%	66%

Maximum likelihood estimates from Switchboard:

- P(NP → DT NN) = 0.28
- P(NP → PRP) = 0.25

SLP3; Francis et al. 1999

Splitting non-terminals

Rather than having a single rule for each non-terminal $P(NP \rightarrow DT\ NN)$, we can condition on some context (Johnson 1998)

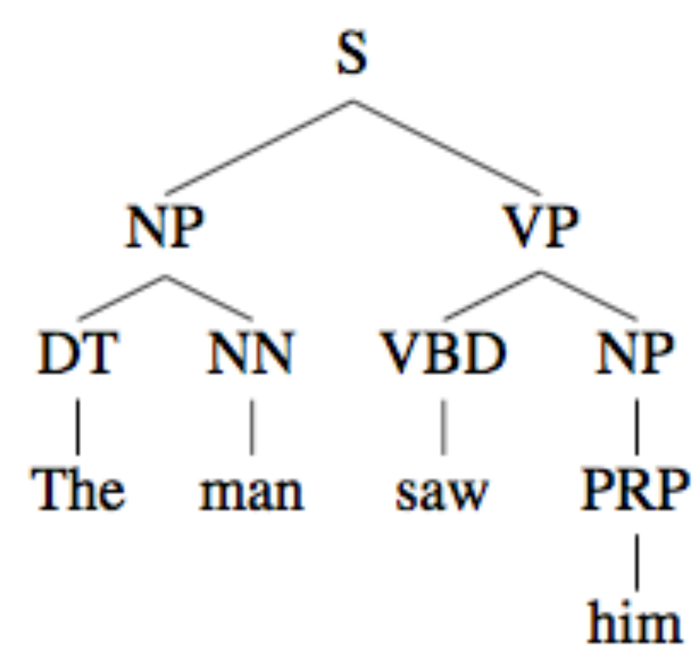
- $P_{subject}(NP \rightarrow DT\ NN)$
- $P_{object}(NP \rightarrow DT\ NN)$

Splitting non-terminals

- Subjects/objects are structural relations in phrase structure trees

- Subject = NP child of S

- Object = NP child of VP child of S



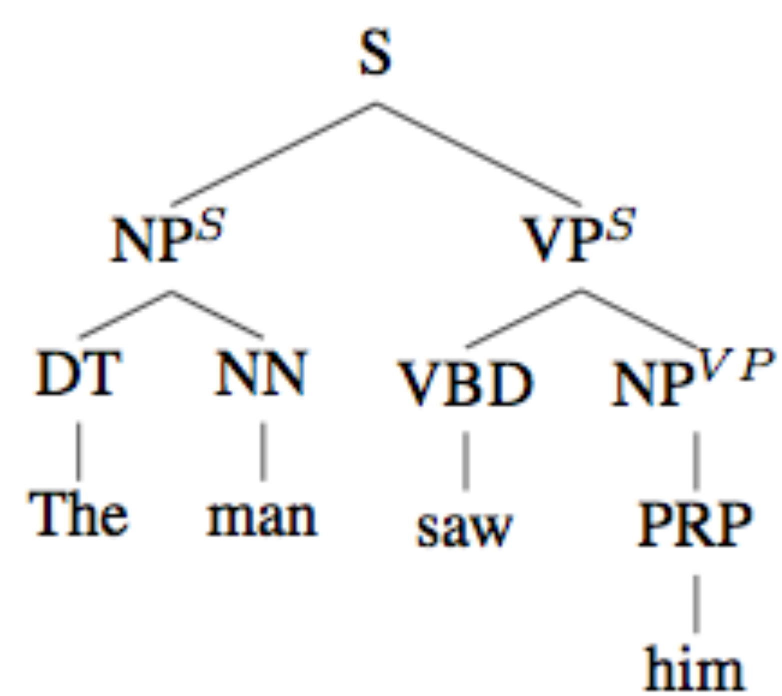
(For some subjects/objects; other rules for embedded structures)

Parent annotation

We can encode context in a general way by annotating each node in a tree with its parent

This lets us learn different probabilities for:

- NP^S (subjects)
- NP^{VP} (objects)



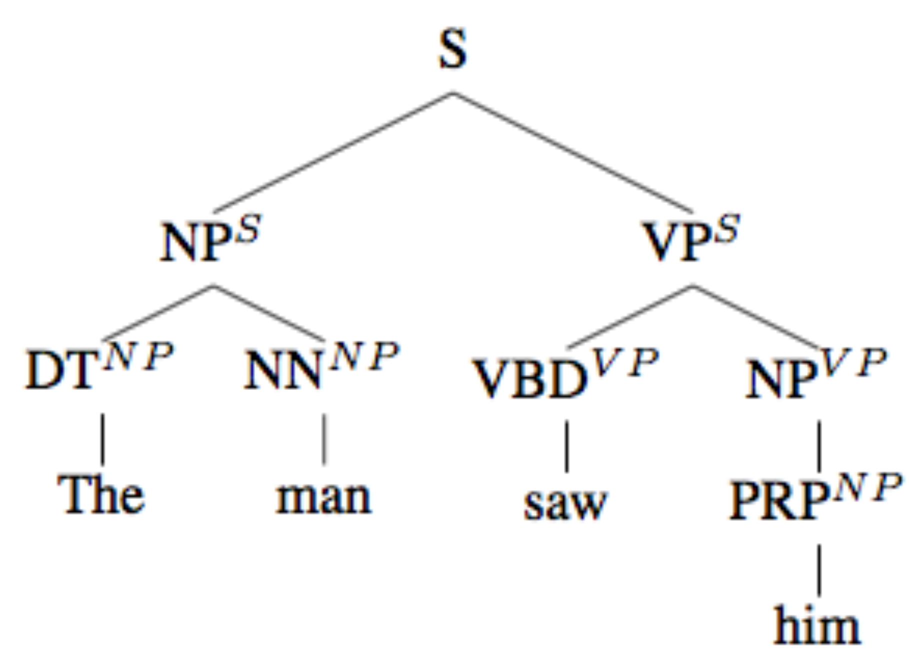
Parent annotation

We can also split the pre-terminal POS tags too

(Klein and Manning 2003)

This lets us learn different probabilities

- $P(RB^{VP} \rightarrow \text{not})$
- $P(RB^{NP} \rightarrow \text{not})$



Splitting non-terminals

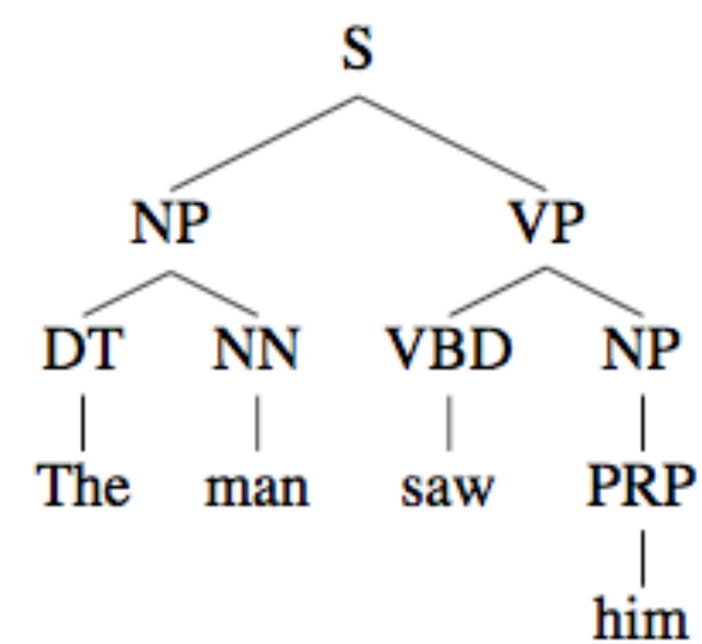
- Dramatically increases the size of the grammar → less training data for each production

- Modern approaches search for best splits that maximize the training data likelihood (Petrov et al 2006)

Problems with PCFGs

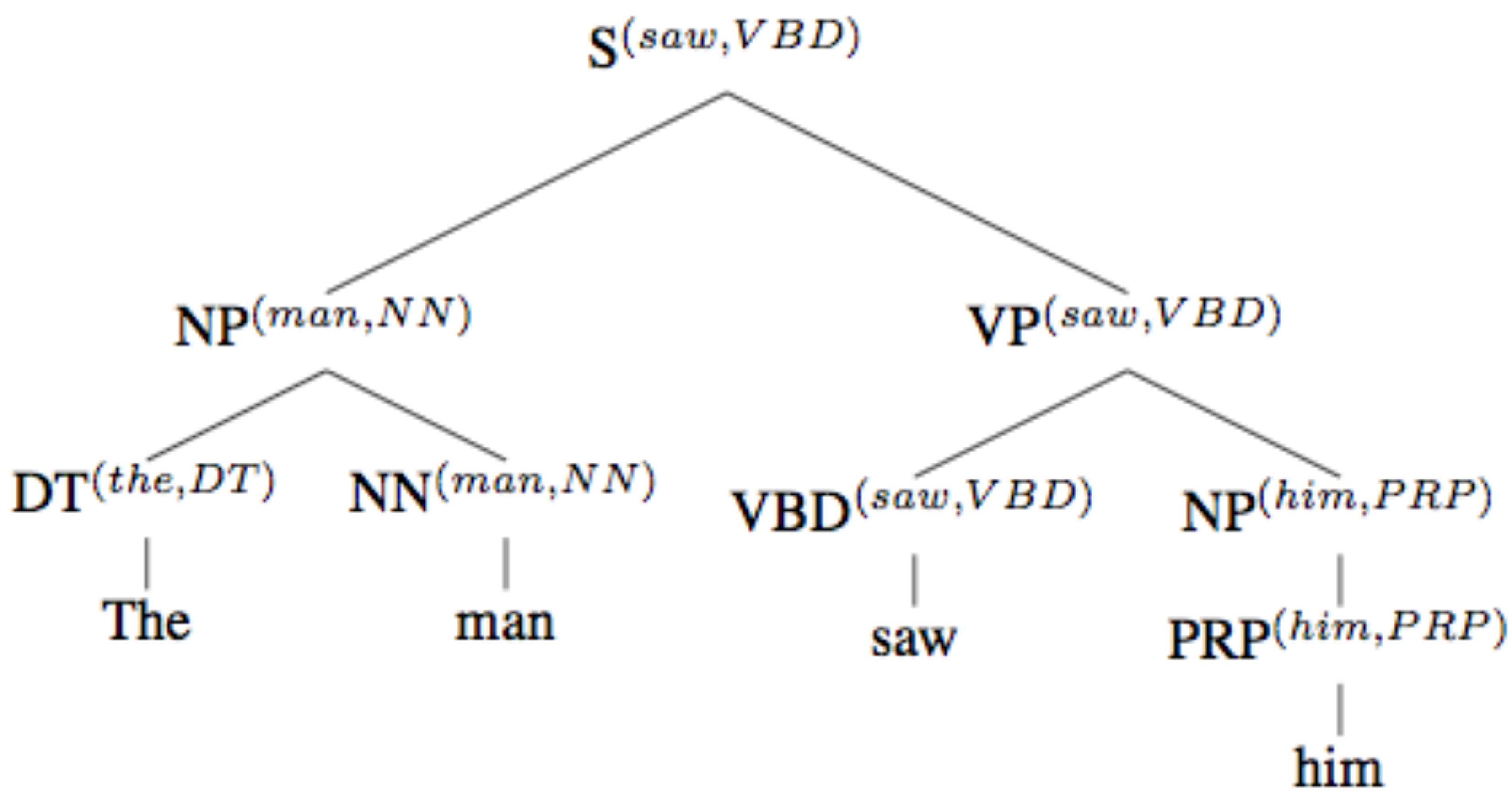
- Lexicon information in a PCFG has little influence on the overall parse structure

- P(VBD → saw) — "saw" itself doesn't influence the structure above it except through that pre-terminal.



Lexicalized PCFGs

- Annotate each node with its head + POS tag of head



Lexicalized PCFGs

- Annotate each node with its head + POS tag of head
- We can't have a rule for each fine-grained production — e.g. $P(S^{(\text{saw, VBD})} \rightarrow NP^{(\text{man, NN})}\ VP^{(\text{saw, VBD})})$
- Different models make different independent assumptions to make this quantity tractable (Collins 1999, Charniak 1997)

Phrase structure parsing

- Discriminative re-ranking (Charniak and Johnson 2005; McClosky et al. 2006)
- Parsing with compositional vector grammars (Socher et al. 2013)
- Parsing as sequence-to-sequence (Vinyals et al. 2015)
- Parsing with recurrent neural network grammars (Dyer et al. 2016)

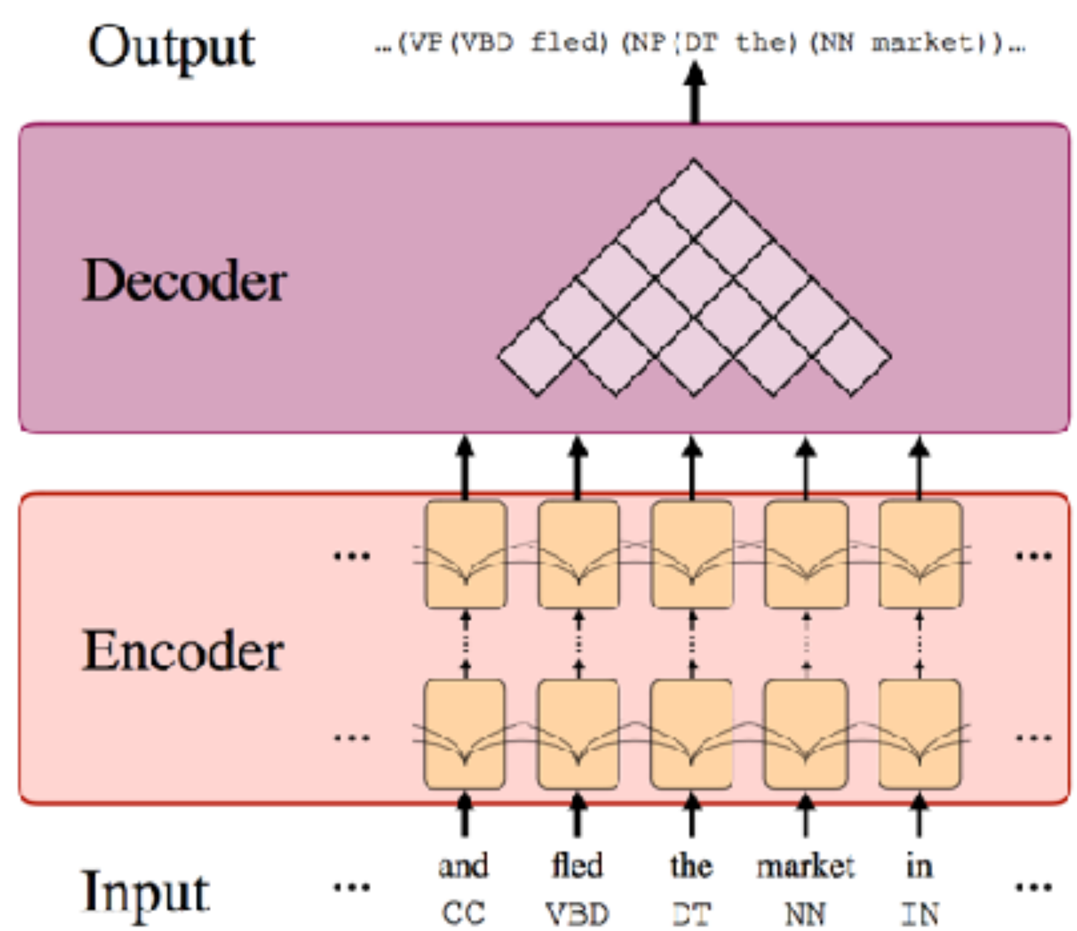
Neural parsing

- Kitaev and Klein (2018), “Constituency Parsing with a Self-Attentive Encoder”

- Neural model (attention encoder) generates representations of each token in a sentence)

- Learned scoring s(i,j,k) function for each span from token i to token j with label k

- CKY for decoding to find the best tree through this space.



Neural parsing

Method	F score
Petrov et al. 2006	89.6
Charniak et al. 2005	91.0
Stern et al. 2017	91.7
Kitaev and Klein 2018	93.6
+ELMO	95.1