Overview

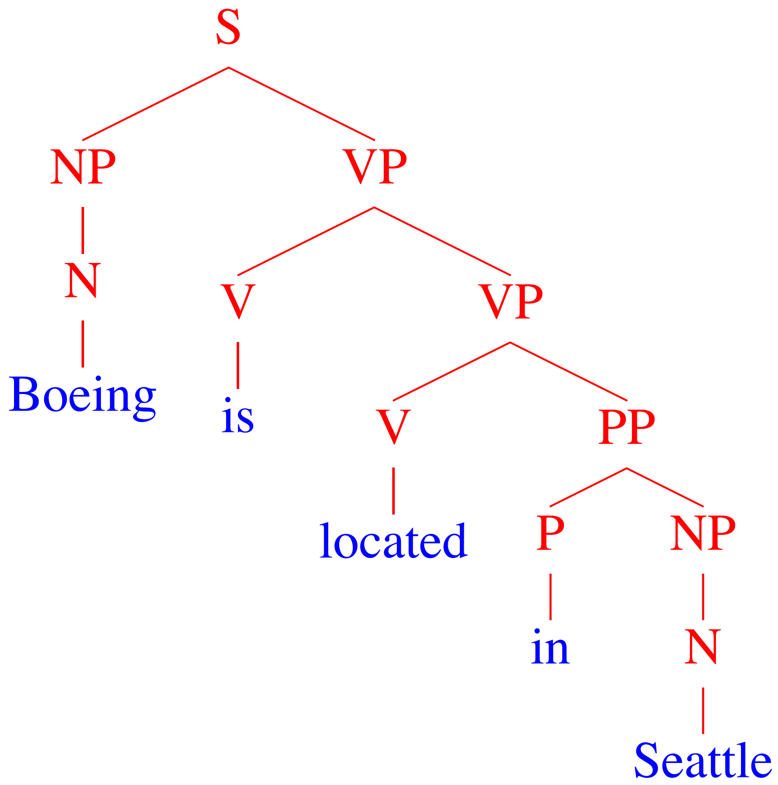
- An introduction to the parsing problem
- Context free grammars
- A brief(!) sketch of the syntax of English
- Examples of ambiguous structures
- PCFGs, their formal properties, and useful algorithms
- Weaknesses of PCFGs

Parsing (Syntactic Structure)

INPUT:

Boeing is located in Seattle.

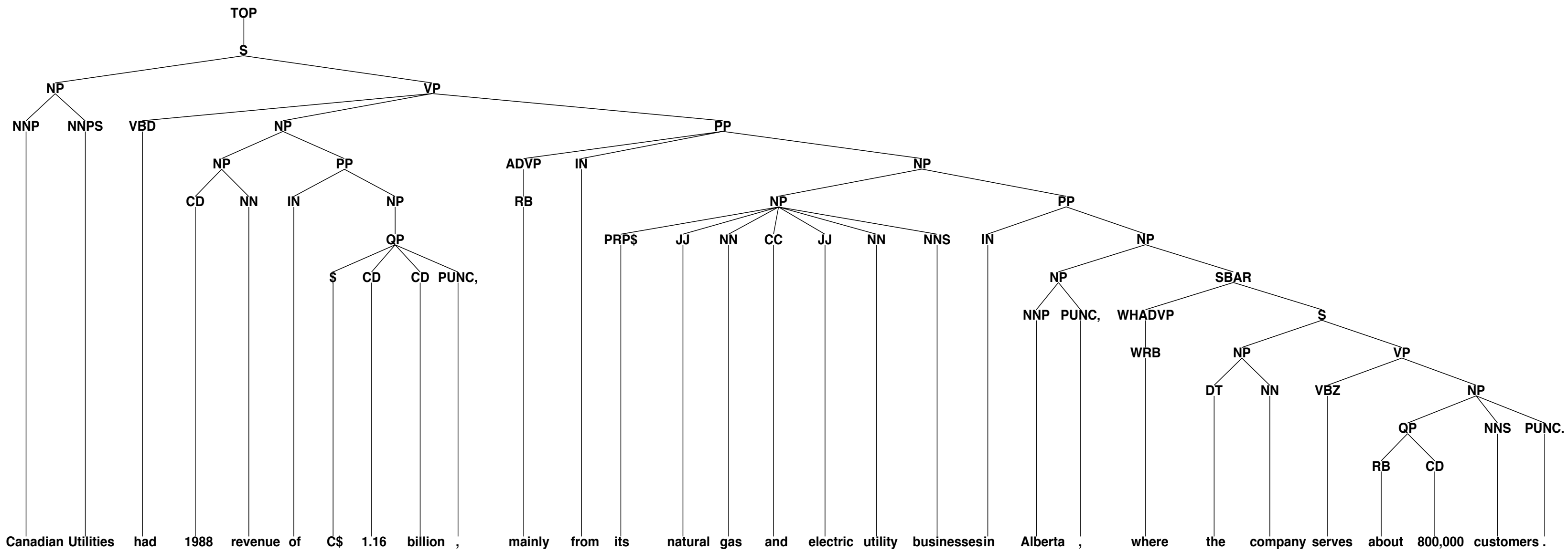
OUTPUT:



Data for Parsing Experiments

- Penn WSJ Treebank = 50,000 sentences with associated trees
- Usual set-up: 40,000 training sentences, 2400 test sentences

An example tree:

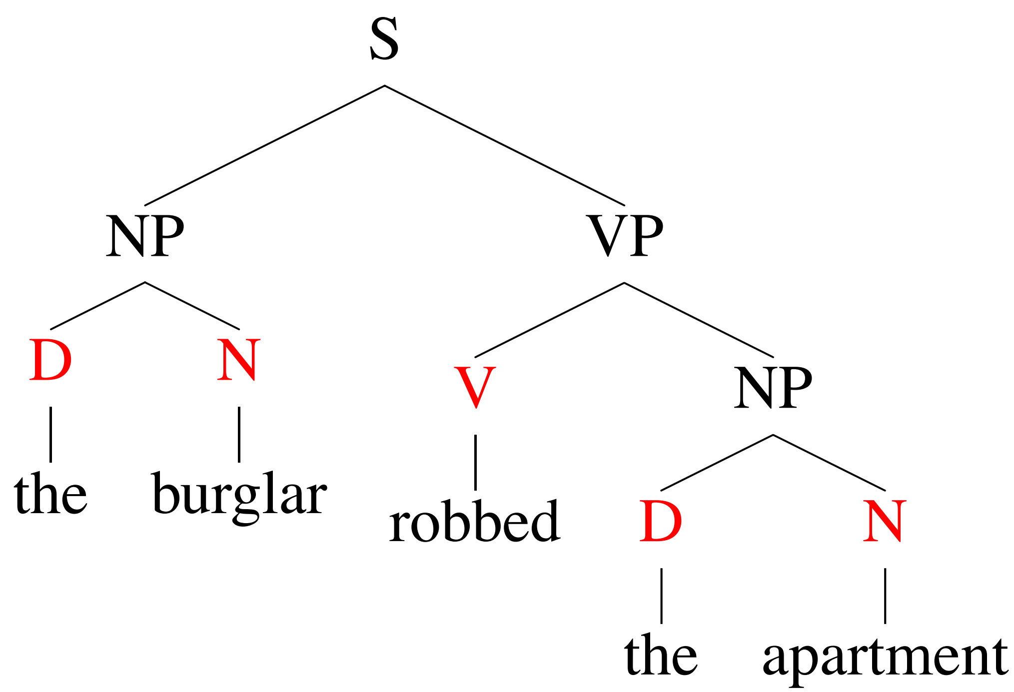


Canadian Utilities had 1988 revenue of C$ 1.16 billion , mainly from its natural gas and electric utility businesses in Alberta , where the company serves about 800,000 customers .

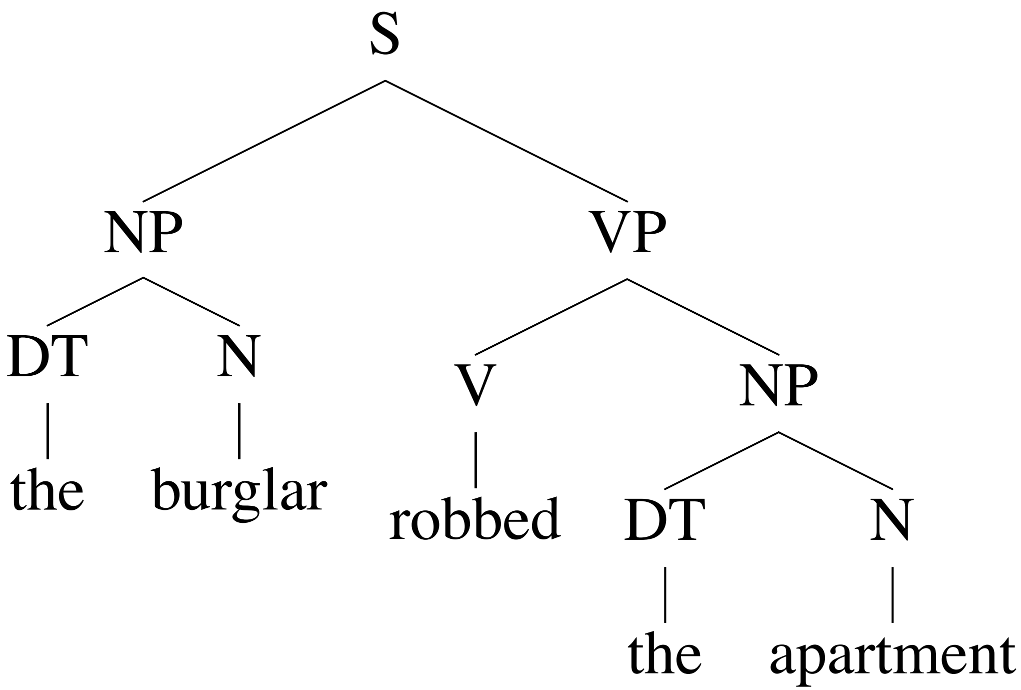
The Information Conveyed by Parse Trees

1) Part of speech for each word

(N = noun, V = verb, D = determiner)



2) Phrases

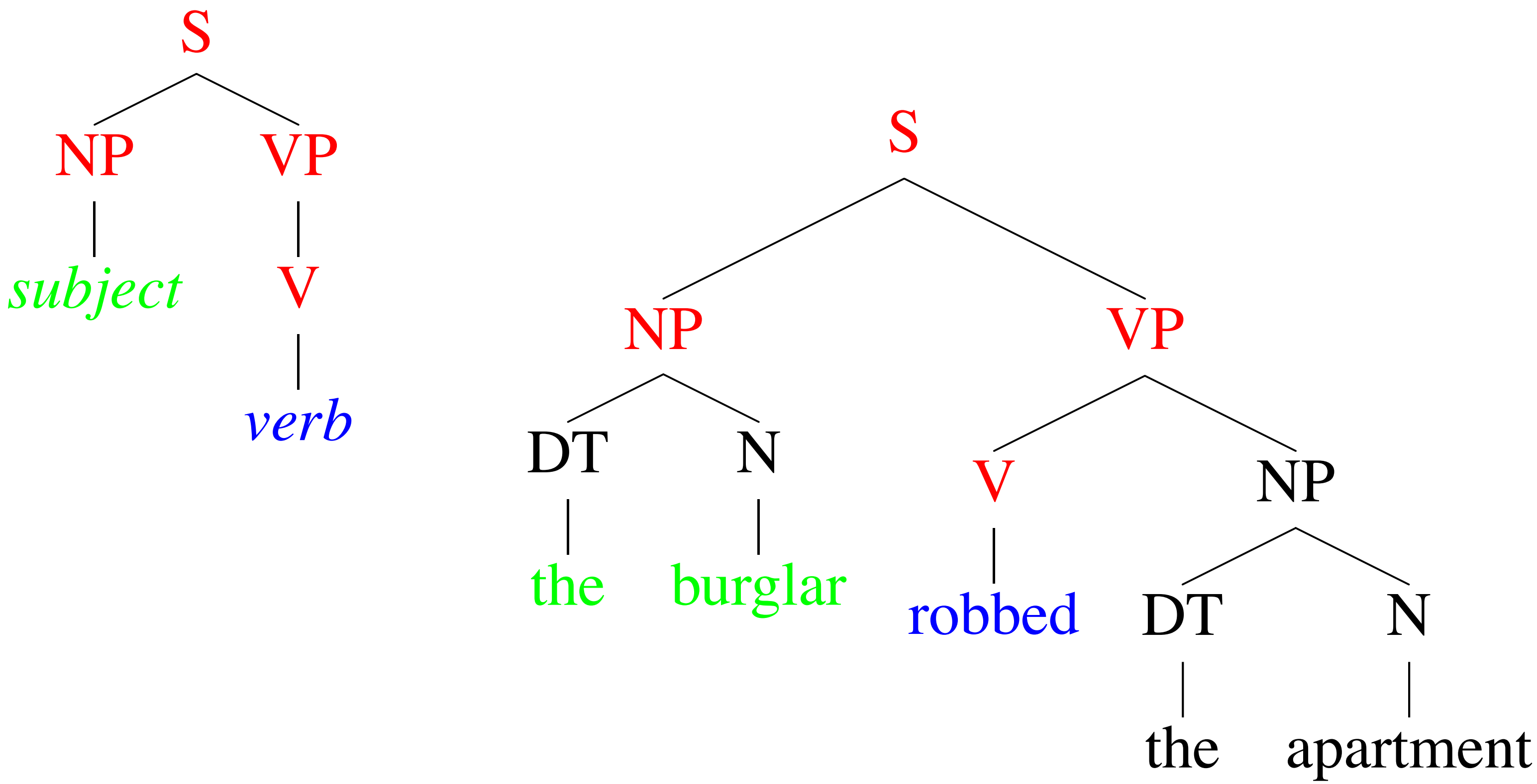


Noun Phrases (NP): "the burglar", "the apartment"

Verb Phrases (VP): "robbed the apartment"

Sentences (S): "the burglar robbed the apartment"

3) Useful Relationships



⇒ "the burglar" is the subject of "robbed"

An Example Application: Machine Translation

- English word order is *subject – verb – object*
- Japanese word order is *subject – object – verb*

English:	IBM bought Lotus
Japanese:	*IBM Lotus bought*

English:	Sources said that IBM bought Lotus yesterday
Japanese:	*Sources yesterday IBM Lotus bought that said*

Syntax and Compositional Semantics

- S:$bought(IBM, Lotus)$
  - NP:***IBM***
    - IBM
  - VP:$\lambda y\ bought(y, Lotus)$
    - V:$\lambda x, y\ bought(y, x)$
      - bought
    - NP:***Lotus***
      - Lotus

- Each syntactic non-terminal now has an associated semantic expression
- (We'll see more of this later in the course)

Context-Free Grammars

[Hopcroft and Ullman 1979]
A context free grammar $G = (N, \Sigma, R, S)$ where:

- N is a set of non-terminal symbols
- Σ is a set of terminal symbols
- R is a set of rules of the form $X \rightarrow Y_1 Y_2 \ldots Y_n$ for $n \geq 0$, $X \in N$, $Y_i \in (N \cup \Sigma)$
- $S \in N$ is a distinguished start symbol

A Context-Free Grammar for English

N = {S, NP, VP, PP, DT, Vi, Vt, NN, IN}
S = S
Σ = {sleeps, saw, man, woman, telescope, the, with, in}

R =

S	$\Rightarrow$	NP	VP
VP	$\Rightarrow$	Vi	
VP	$\Rightarrow$	Vt	NP
VP	$\Rightarrow$	VP	PP
NP	$\Rightarrow$	DT	NN
NP	$\Rightarrow$	NP	PP
PP	$\Rightarrow$	IN	NP

Vi	$\Rightarrow$	sleeps
Vt	$\Rightarrow$	saw
NN	$\Rightarrow$	man
NN	$\Rightarrow$	woman
NN	$\Rightarrow$	telescope
DT	$\Rightarrow$	the
IN	$\Rightarrow$	with
IN	$\Rightarrow$	in

Note: S=sentence, VP=verb phrase, NP=noun phrase, PP=prepositional phrase, DT=determiner, Vi=intransitive verb, Vt=transitive verb, NN=noun, IN=preposition

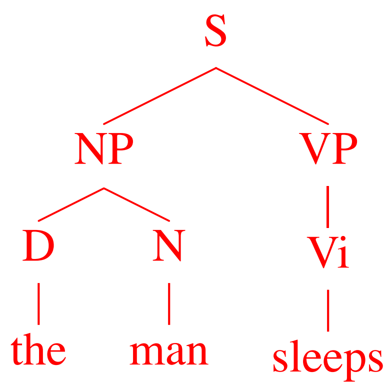
Left-Most Derivations

A left-most derivation is a sequence of strings $s_1 \ldots s_n$, where

- $s_1 = S$, the start symbol
- $s_n \in \Sigma^*$, i.e. s_n is made up of terminal symbols only
- Each s_i for $i = 2 \ldots n$ is derived from s_{i-1} by picking the left-most non-terminal X in s_{i-1} and replacing it by some β where $X \rightarrow \beta$ is a rule in R

For example: [S], [NP VP], [D N VP], [the N VP], [the man VP], [the man Vi], [the man sleeps]

Representation of a derivation as a tree:



DERIVATION	RULES USED
S	

DERIVATION	RULES USED
S	S $\longrightarrow$ NP VP
NP VP	

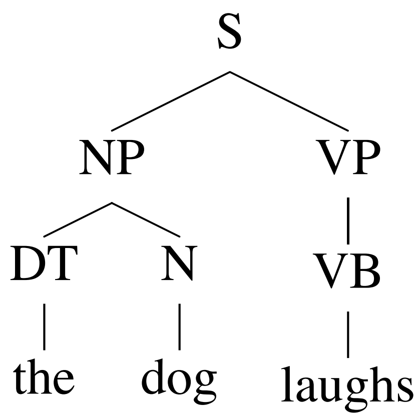
DERIVATION	RULES USED
S	S $\longrightarrow$ NP VP
NP VP	NP $\longrightarrow$ DT N
DT N VP	

DERIVATION	RULES USED
S	$S \longrightarrow NP\ VP$
NP VP	$NP \longrightarrow DT\ N$
DT N VP	$DT \longrightarrow$ the
the N VP	

DERIVATION	RULES USED
S	S $\rightarrow$ NP VP
NP VP	NP $\rightarrow$ DT N
DT N VP	DT $\rightarrow$ the
the N VP	N $\rightarrow$ dog
the dog VP	

DERIVATION	RULES USED
S	S $\longrightarrow$ NP VP
NP VP	NP $\longrightarrow$ DT N
DT N VP	DT $\longrightarrow$ the
the N VP	N $\longrightarrow$ dog
the dog VP	VP $\longrightarrow$ VB
the dog VB	

DERIVATION	RULES USED
S	S $\longrightarrow$ NP VP
NP VP	NP $\longrightarrow$ DT N
DT N VP	DT $\longrightarrow$ the
the N VP	N $\longrightarrow$ dog
the dog VP	VP $\longrightarrow$ VB
the dog VB	VB $\longrightarrow$ laughs
the dog laughs	



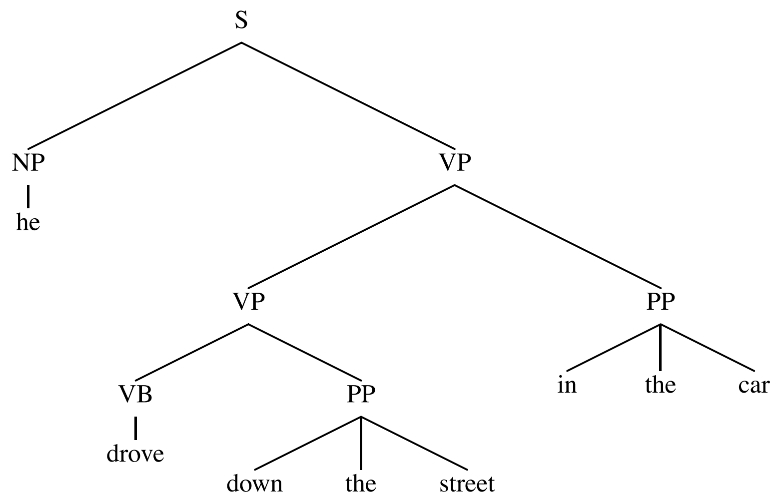
Properties of CFGs

- A CFG defines a set of possible derivations
- A string $s \in \Sigma^*$ is in the *language* defined by the CFG if there is at least one derivation which yields s
- Each string in the language generated by the CFG may have more than one derivation (“ambiguity”)

DERIVATION

S

RULES USED



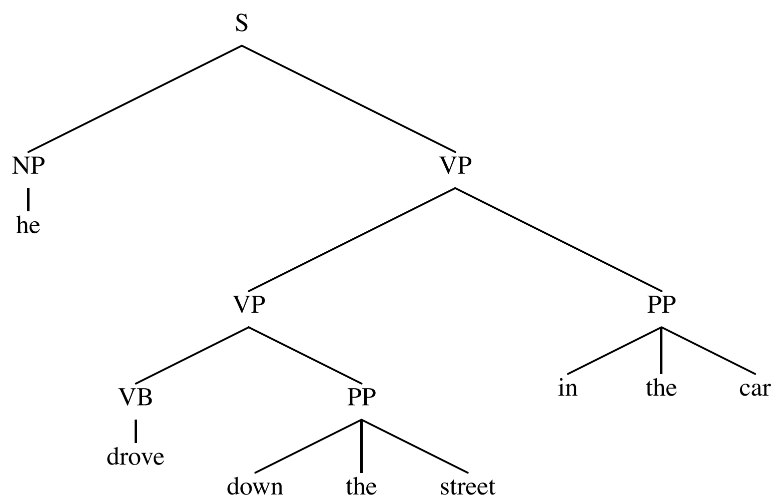
DERIVATION

S

NP VP

RULES USED

$S \rightarrow NP\ VP$



DERIVATION

S

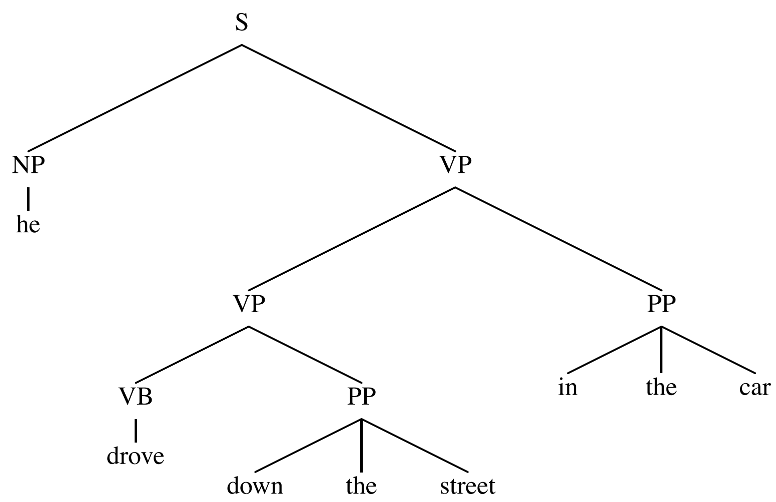
NP VP

he VP

RULES USED

S $\longrightarrow$ NP VP

NP $\longrightarrow$ he



DERIVATION

S

NP VP

he VP

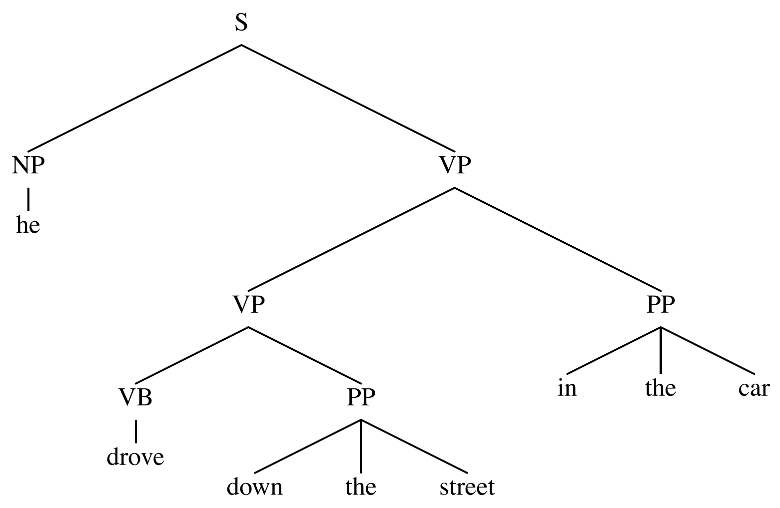
he VP PP

RULES USED

S $\longrightarrow$ NP VP

NP $\longrightarrow$ he

VP $\longrightarrow$ VP PP



DERIVATION

S

NP VP

he VP

he VP PP

he VB PP PP

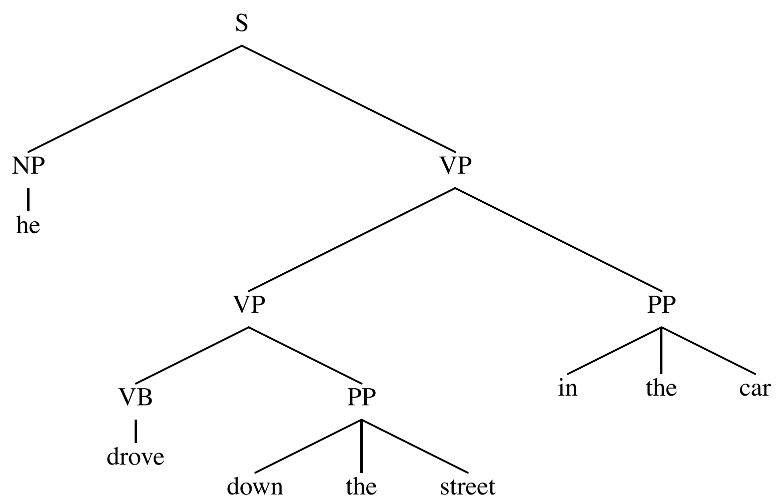
RULES USED

S $\longrightarrow$ NP VP

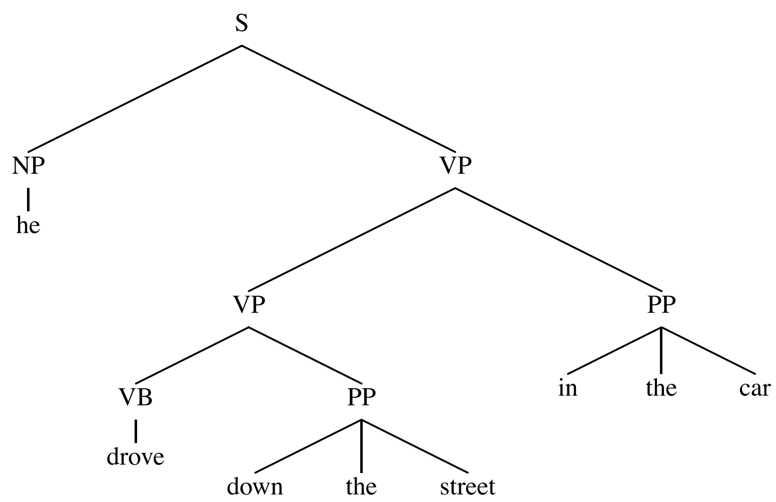
NP $\longrightarrow$ he

VP $\longrightarrow$ VP PP

VP $\longrightarrow$ VB PP



DERIVATION	RULES USED
S	S $\longrightarrow$ NP VP
NP VP	NP $\longrightarrow$ he
he VP	VP $\longrightarrow$ VP PP
he VP PP	VP $\longrightarrow$ VB PP
he VB PP PP	VB $\longrightarrow$ drove
he drove PP PP	



DERIVATION

S

NP VP

he VP

he VP PP

he VB PP PP

he drove PP PP

he drove down the street PP

RULES USED

S $\longrightarrow$ NP VP

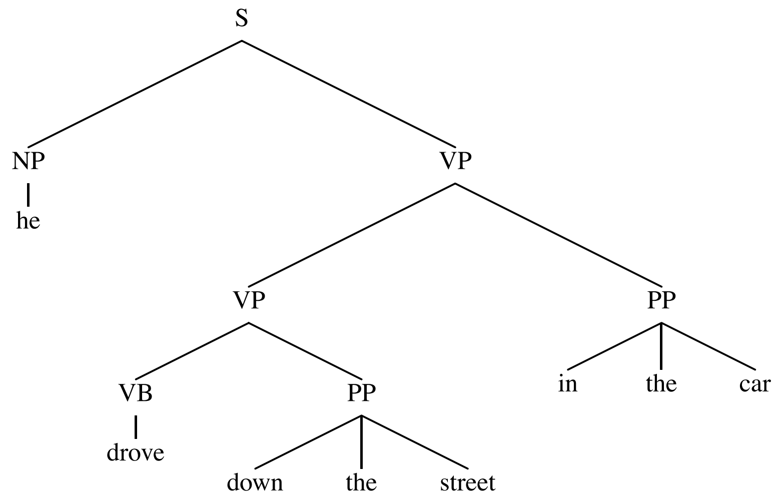
NP $\longrightarrow$ he

VP $\longrightarrow$ VP PP

VP $\longrightarrow$ VB PP

VB$\longrightarrow$ drove

PP$\longrightarrow$ down the street



DERIVATION

S

NP VP

he VP

he VP PP

he VB PP PP

he drove PP PP

he drove down the street PP

he drove down the street in the car

RULES USED

S $\longrightarrow$ NP VP

NP $\longrightarrow$ he

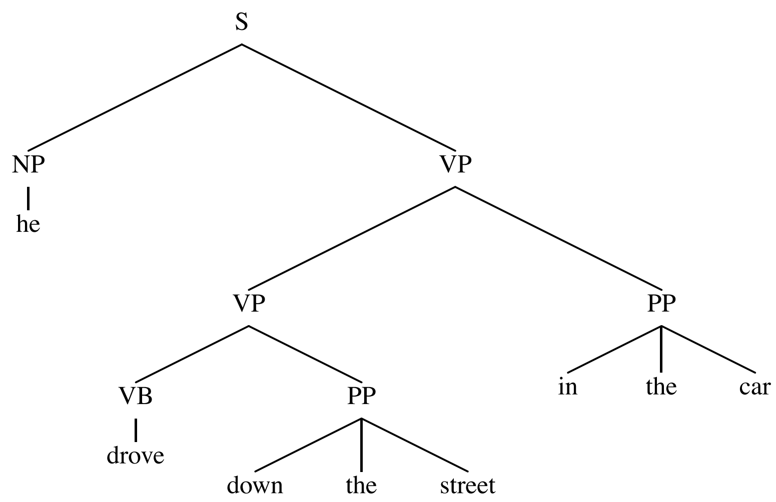
VP $\longrightarrow$ VP PP

VP $\longrightarrow$ VB PP

VB $\longrightarrow$ drove

PP $\longrightarrow$ down the street

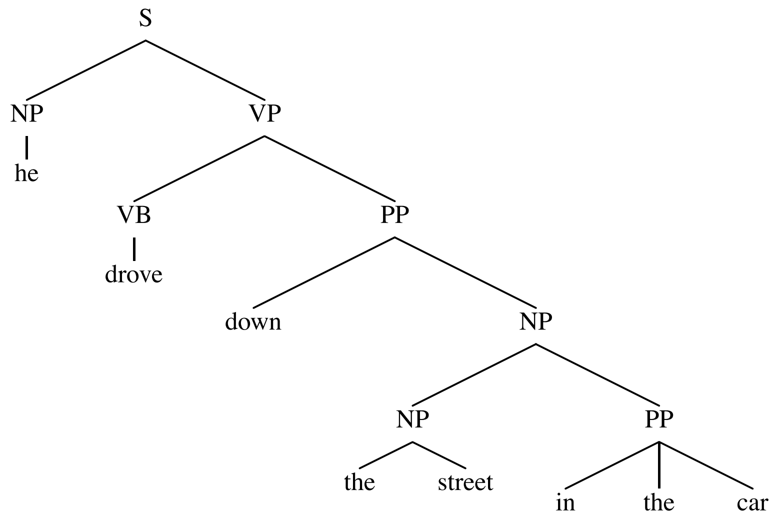
PP $\longrightarrow$ in the car



DERIVATION

S

RULES USED



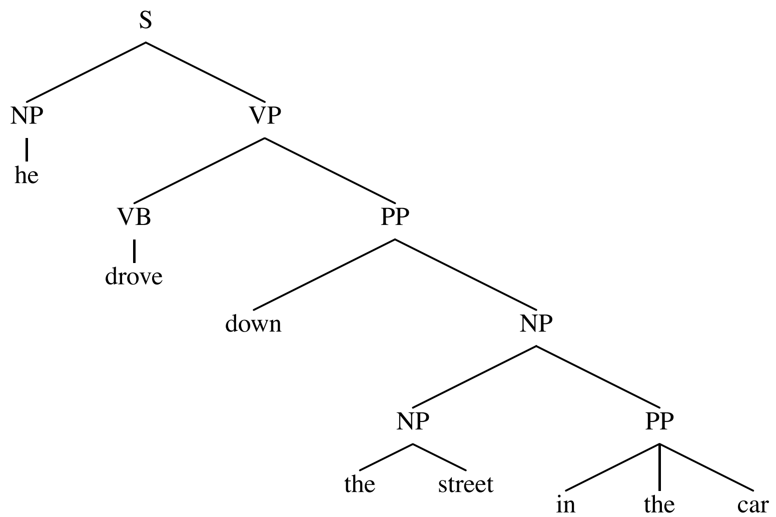
DERIVATION

S

NP VP

RULES USED

S → NP VP



DERIVATION

S

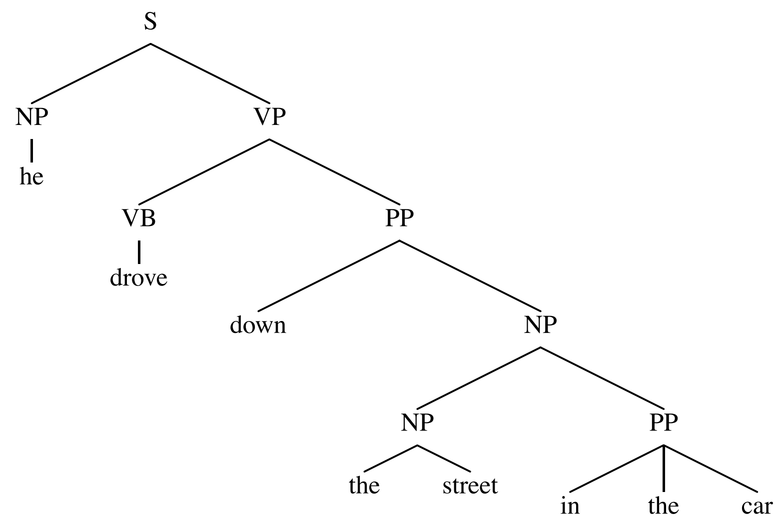
NP VP

he VP

RULES USED

S $\longrightarrow$ NP VP

NP $\longrightarrow$ he



DERIVATION

S

NP VP

he VP

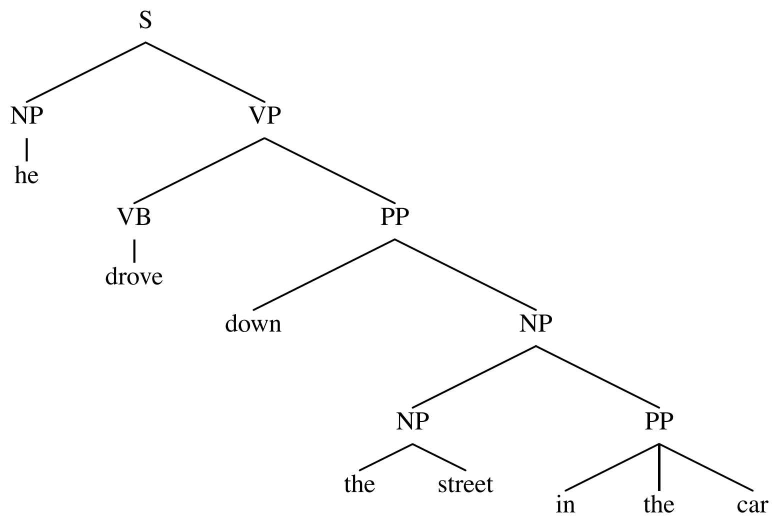
he VB PP

RULES USED

S $\longrightarrow$ NP VP

NP $\longrightarrow$ he

VP $\longrightarrow$ VB PP



DERIVATION

S

NP VP

he VP

he VB PP

he drove PP

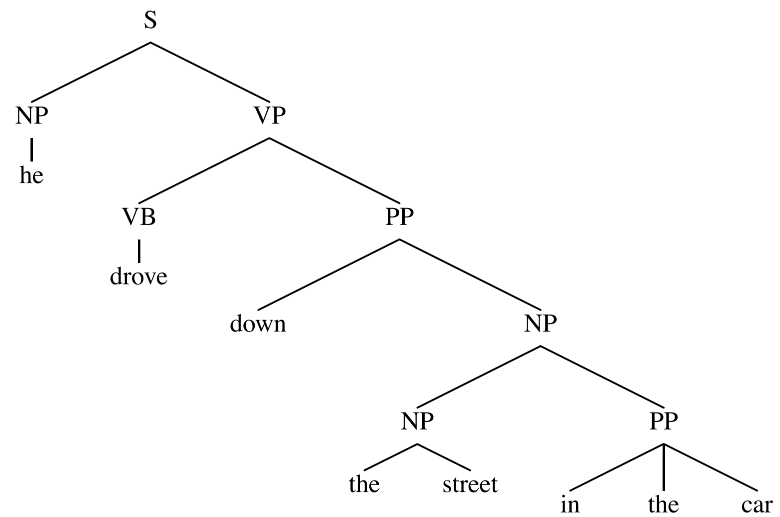
RULES USED

S $\longrightarrow$ NP VP

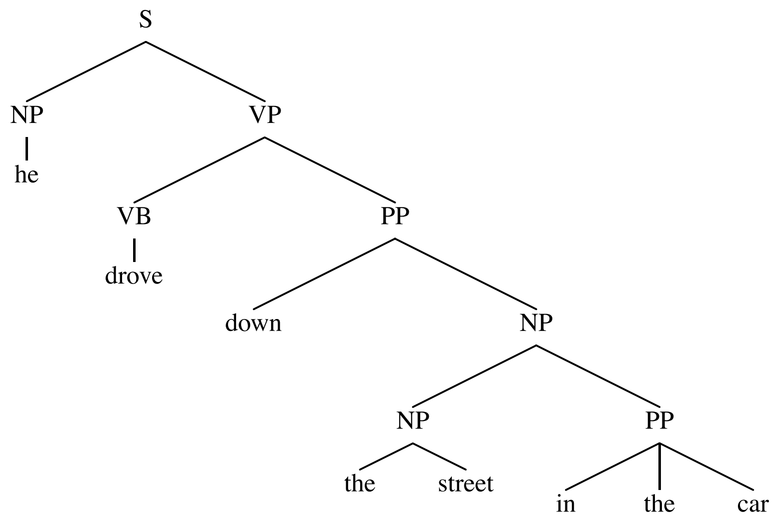
NP $\longrightarrow$ he

VP $\longrightarrow$ VB PP

VB $\longrightarrow$ drove



DERIVATION	RULES USED
S	S $\longrightarrow$ NP VP
NP VP	NP $\longrightarrow$ he
he VP	VP $\longrightarrow$ VB PP
he VB PP	VB $\longrightarrow$ drove
he drove PP	PP $\longrightarrow$ down NP
he drove down NP	



DERIVATION

S

NP VP

he VP

he VB PP

he drove PP

he drove down NP

he drove down NP PP

RULES USED

S $\longrightarrow$ NP VP

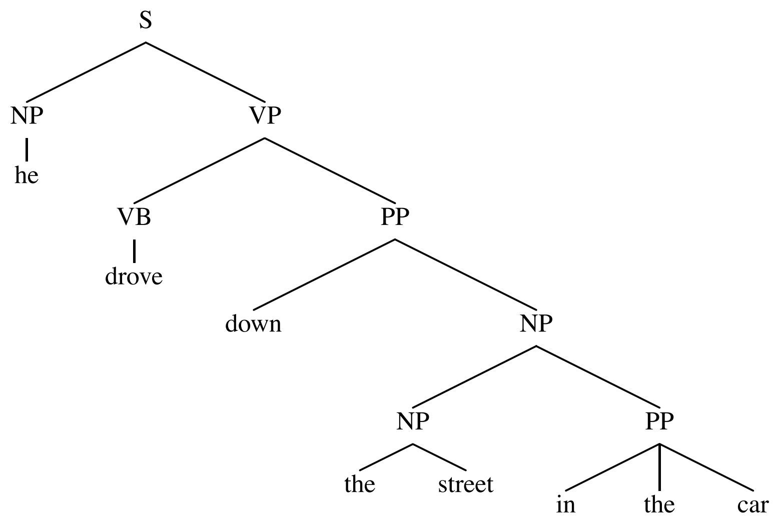
NP $\longrightarrow$ he

VP $\longrightarrow$ VB PP

VB $\longrightarrow$ drove

PP $\longrightarrow$ down NP

NP $\longrightarrow$ NP PP



DERIVATION

S

NP VP

he VP

he VB PP

he drove PP

he drove down NP

he drove down NP PP

he drove down the street PP

RULES USED

S $\rightarrow$ NP VP

NP $\rightarrow$ he

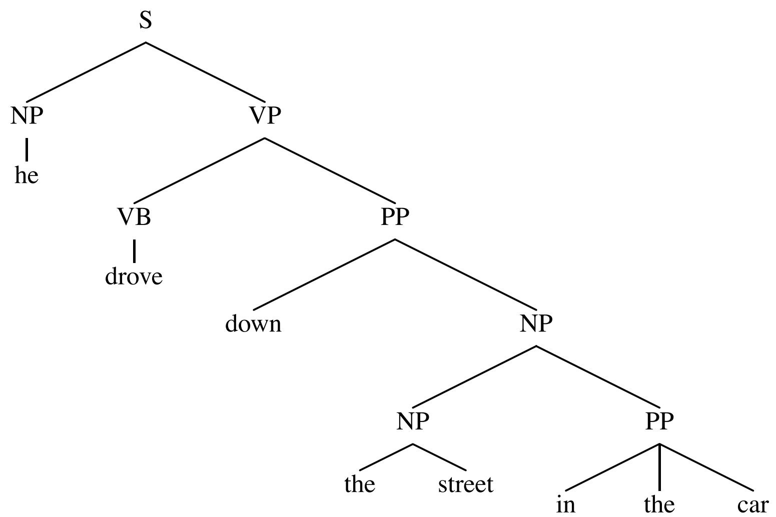
VP $\rightarrow$ VB PP

VB $\rightarrow$ drove

PP $\rightarrow$ down NP

NP $\rightarrow$ NP PP

NP $\rightarrow$ the street



DERIVATION

S

NP VP

he VP

he VB PP

he drove PP

he drove down NP

he drove down NP PP

he drove down the street PP

he drove down the street in the car

RULES USED

S $\longrightarrow$ NP VP

NP $\longrightarrow$ he

VP $\longrightarrow$ VB PP

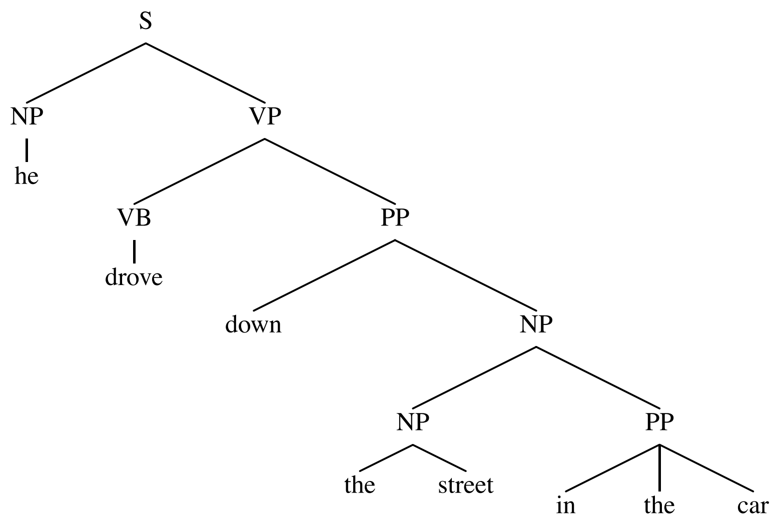
VB $\longrightarrow$ drove

PP $\longrightarrow$ down NP

NP $\longrightarrow$ NP PP

NP $\longrightarrow$ the street

PP $\longrightarrow$ in the car



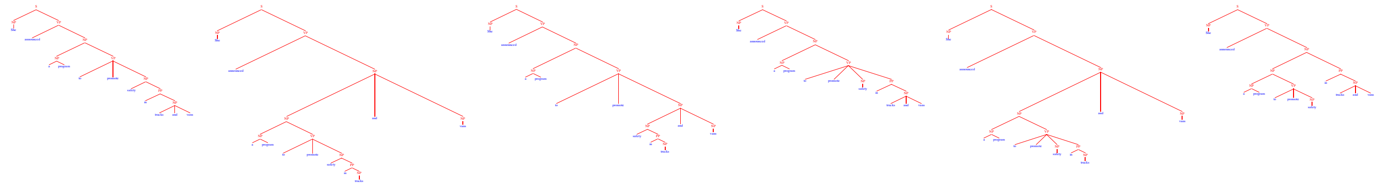
The Problem with Parsing: Ambiguity

INPUT:

She announced a program to promote safety in trucks and vans

$\Downarrow$

POSSIBLE OUTPUTS:



And there are more...

A Brief Overview of English Syntax

Parts of Speech:

- Nouns
  (Tags from the *Brown corpus*)
  NN = singular noun e.g., man, dog, park
  NNS = plural noun e.g., telescopes, houses, buildings
  NNP = proper noun e.g., Smith, Gates, IBM

- Determiners
  DT = determiner e.g., the, a, some, every

- Adjectives
  JJ = adjective e.g., red, green, large, idealistic

A Fragment of a Noun Phrase Grammar

$$\begin{array}{lcll}
\bar{\text{N}} & \Rightarrow & \text{NN} & \\
\bar{\text{N}} & \Rightarrow & \text{NN} & \bar{\text{N}} \\
\bar{\text{N}} & \Rightarrow & \text{JJ} & \bar{\text{N}} \\
\bar{\text{N}} & \Rightarrow & \bar{\text{N}} & \bar{\text{N}} \\
\text{NP} & \Rightarrow & \text{DT} & \bar{\text{N}}
\end{array}$$

$$\begin{array}{lcl}
\text{NN} & \Rightarrow & \text{box} \\
\text{NN} & \Rightarrow & \text{car} \\
\text{NN} & \Rightarrow & \text{mechanic} \\
\text{NN} & \Rightarrow & \text{pigeon} \\
 & & \\
\text{DT} & \Rightarrow & \text{the} \\
\text{DT} & \Rightarrow & \text{a} \\
 & & \\
\text{JJ} & \Rightarrow & \text{fast} \\
\text{JJ} & \Rightarrow & \text{metal} \\
\text{JJ} & \Rightarrow & \text{idealistic} \\
\text{JJ} & \Rightarrow & \text{clay}
\end{array}$$

Generates:

a box, the box, the metal box, the fast car mechanic, . . .

Prepositions, and Prepositional Phrases

- Prepositions
  IN = preposition   e.g., of, in, out, beside, as

An Extended Grammar

$\bar{N}$	$\Rightarrow$	NN	
$\bar{N}$	$\Rightarrow$	NN	$\bar{N}$
$\bar{N}$	$\Rightarrow$	JJ	$\bar{N}$
$\bar{N}$	$\Rightarrow$	$\bar{N}$	$\bar{N}$
NP	$\Rightarrow$	DT	$\bar{N}$
PP	$\Rightarrow$	IN	NP
$\bar{N}$	$\Rightarrow$	$\bar{N}$	PP

NN	$\Rightarrow$	box
NN	$\Rightarrow$	car
NN	$\Rightarrow$	mechanic
NN	$\Rightarrow$	pigeon
DT	$\Rightarrow$	the
DT	$\Rightarrow$	a

JJ	$\Rightarrow$	fast
JJ	$\Rightarrow$	metal
JJ	$\Rightarrow$	idealistic
JJ	$\Rightarrow$	clay
IN	$\Rightarrow$	in
IN	$\Rightarrow$	under
IN	$\Rightarrow$	of
IN	$\Rightarrow$	on
IN	$\Rightarrow$	with
IN	$\Rightarrow$	as

Generates:
in a box, under the box, the fast car mechanic under the pigeon in the box, . . .

Verbs, Verb Phrases, and Sentences

- Basic Verb Types

Vi = Intransitive verb	e.g., sleeps, walks, laughs
Vt = Transitive verb	e.g., sees, saw, likes
Vd = Ditransitive verb	e.g., gave

- Basic VP Rules

$$\mathrm{VP} \rightarrow \mathrm{Vi}$$
$$\mathrm{VP} \rightarrow \mathrm{Vt} \quad \mathrm{NP}$$
$$\mathrm{VP} \rightarrow \mathrm{Vd} \quad \mathrm{NP} \quad \mathrm{NP}$$

- Basic S Rule

$$\mathrm{S} \rightarrow \mathrm{NP} \quad \mathrm{VP}$$

Examples of VP:

sleeps, walks, likes the mechanic, gave the mechanic the fast car, gave the fast car mechanic the pigeon in the box, . . .

Examples of S:

the man sleeps, the dog walks, the dog likes the mechanic, the dog in the box gave the mechanic the fast car,. . .

PPs Modifying Verb Phrases

A new rule:

VP $\rightarrow$ VP PP

New examples of VP:

sleeps in the car, walks like the mechanic, gave the mechanic the fast car on Tuesday, ...

Complementizers, and SBARs

- Complementizers
  COMP = complementizer   e.g., that

- SBAR
  SBAR $\rightarrow$ COMP S

Examples:
that the man sleeps, that the mechanic saw the dog . . .

More Verbs

- New Verb Types

V[5]	e.g., said, reported
V[6]	e.g., told, informed
V[7]	e.g., bet

- New VP Rules

$$\begin{array}{lllll} VP & \rightarrow & V[5] & SBAR & & \\ VP & \rightarrow & V[6] & NP & SBAR & \\ VP & \rightarrow & V[7] & NP & NP & SBAR \end{array}$$

Examples of New VPs:

said that the man sleeps
told the dog that the mechanic likes the pigeon
bet the pigeon $50 that the mechanic owns a fast car

Coordination

- A New Part-of-Speech:
  CC = Coordinator   e.g., and, or, but

- New Rules

NP	$\rightarrow$	NP	CC	NP
$\bar{\text{N}}$	$\rightarrow$	$\bar{\text{N}}$	CC	$\bar{\text{N}}$
VP	$\rightarrow$	VP	CC	VP
S	$\rightarrow$	S	CC	S
SBAR	$\rightarrow$	SBAR	CC	SBAR

Sources of Ambiguity

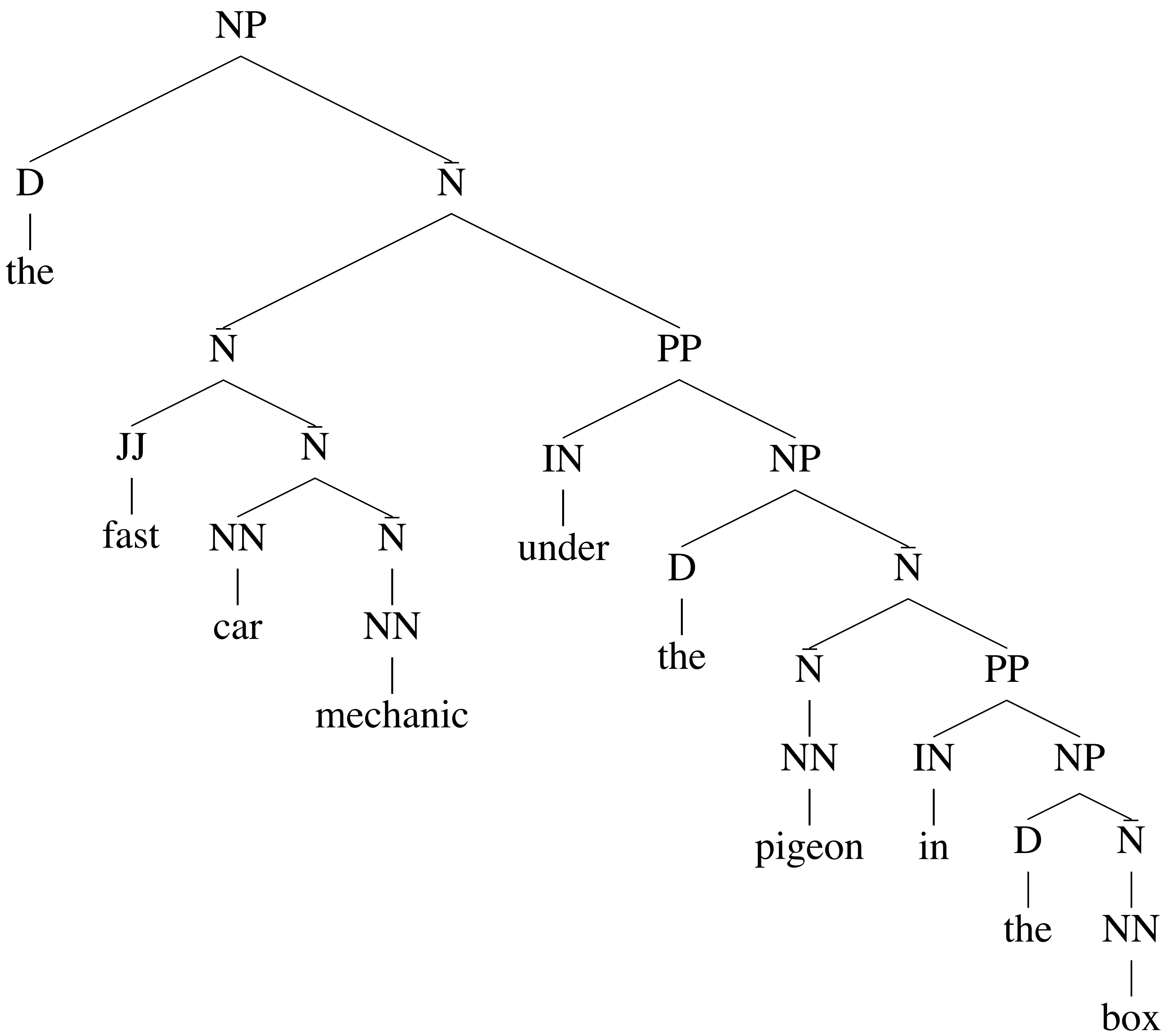
- Part-of-Speech ambiguity

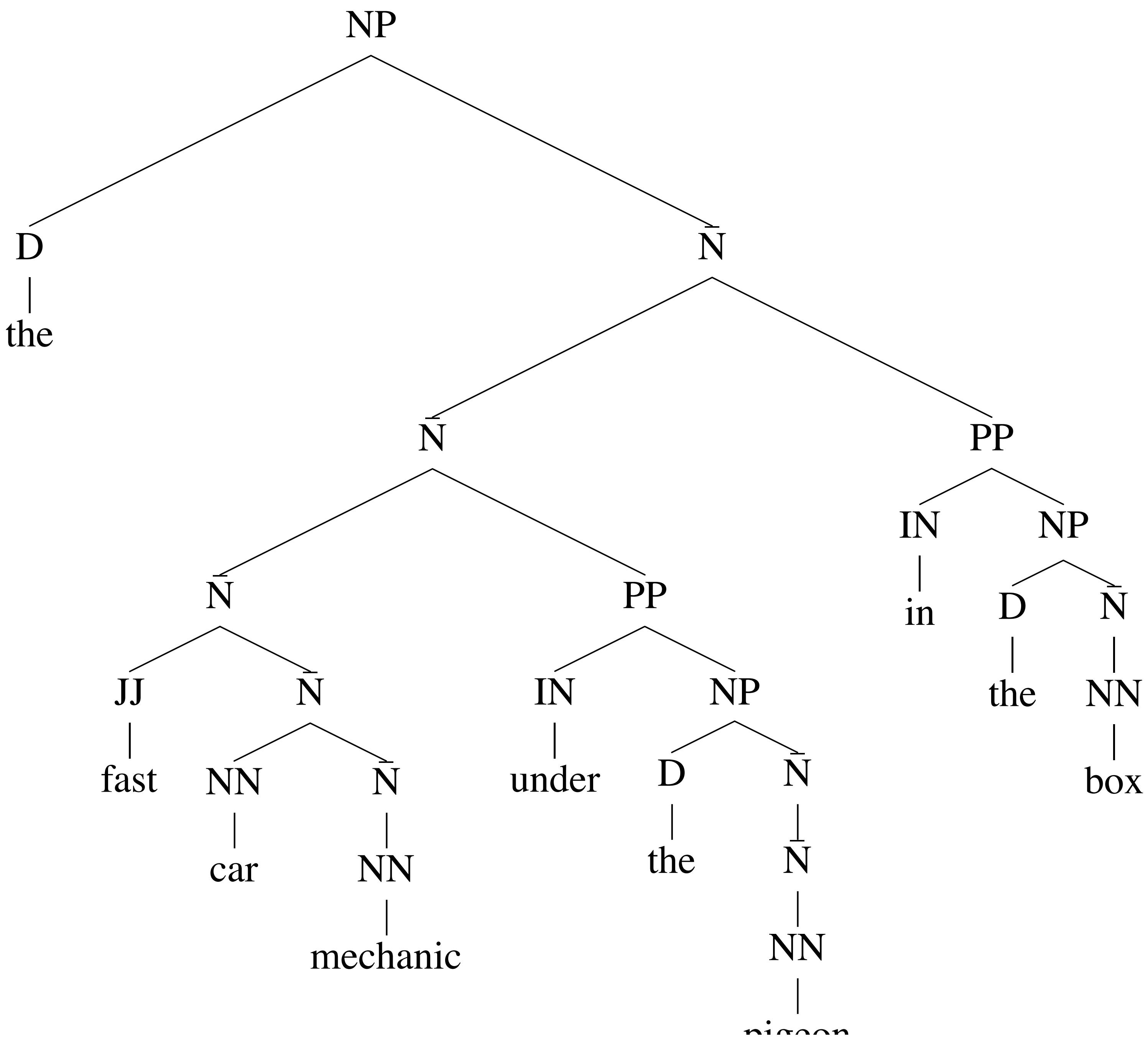
  NNS $\rightarrow$ walks

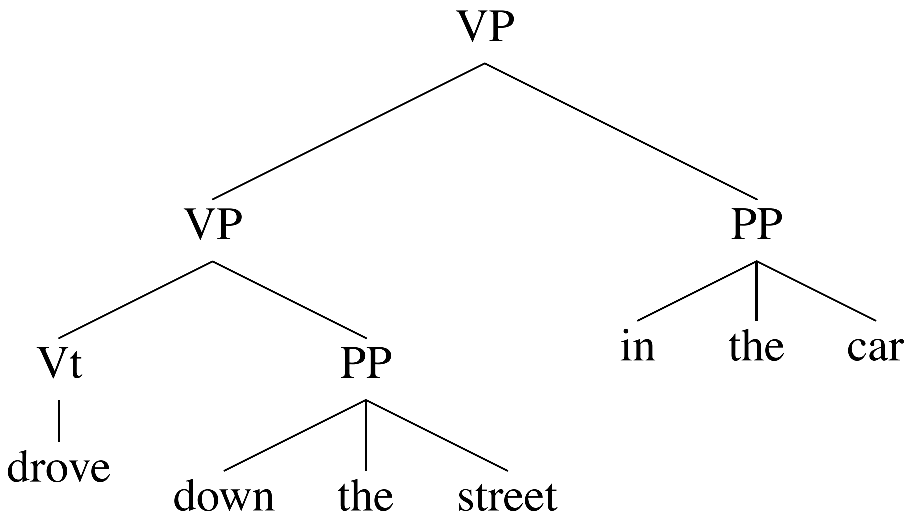
  Vi $\rightarrow$ walks

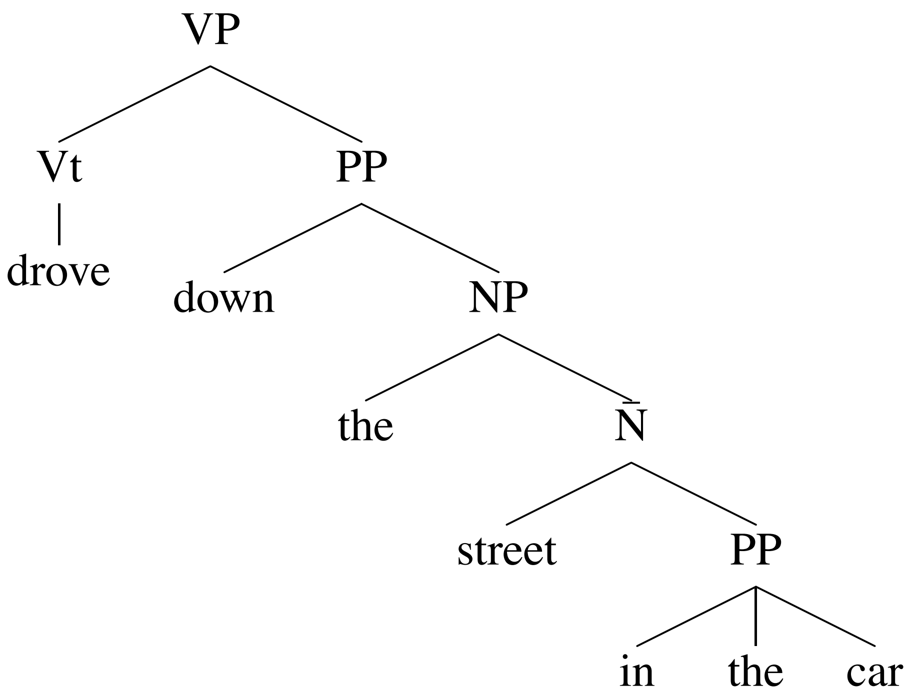
- Prepositional Phrase Attachment

  the fast car mechanic under the pigeon in the box


NP
D
N̄
the
N̄
PP
JJ
N̄
IN
NP
fast
NN
N̄
under
D
N̄
car
NN
the
N̄
PP
mechanic
NN
IN
NP
pigeon
in
D
N̄
the
NN
box


NP
D
the
N̄
N̄
PP
N̄
PP
IN
NP
in
D
N̄
JJ
N̄
IN
NP
the
NN
fast
NN
N̄
under
D
N̄
box
car
NN
the
N
mechanic
NN
pigeon

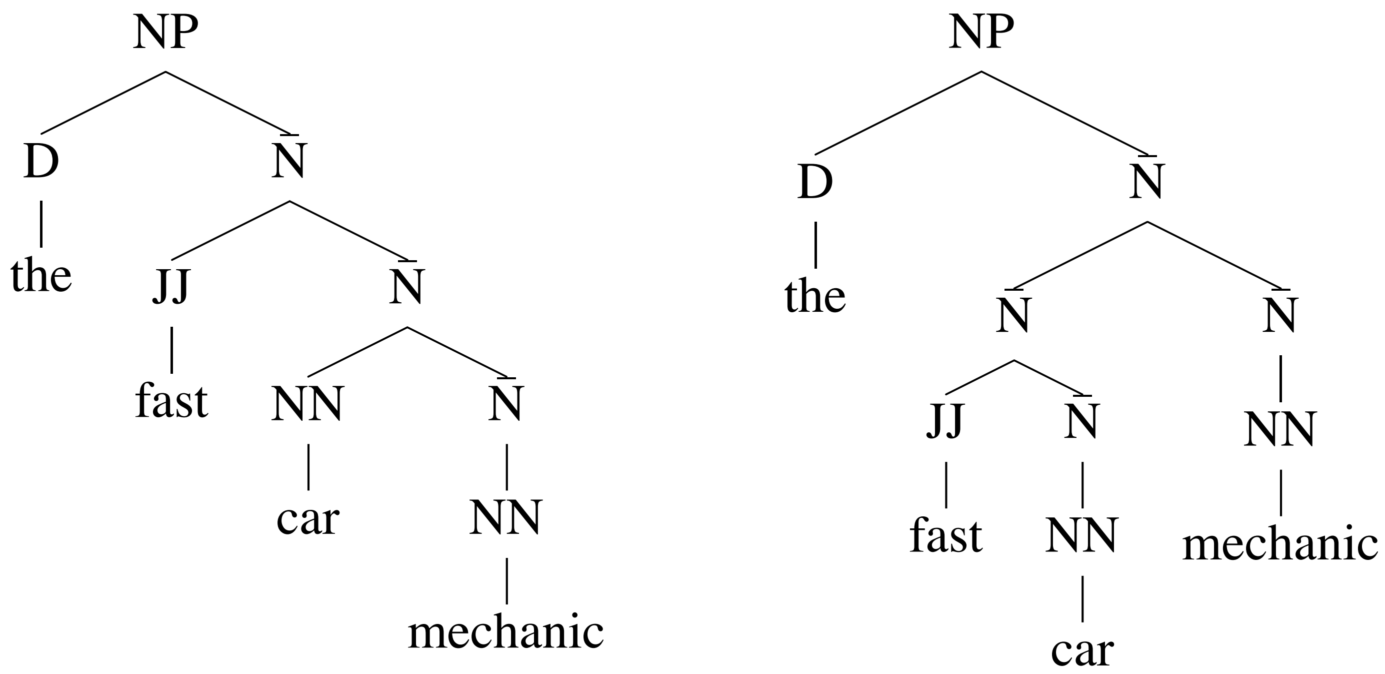

VP
VP
PP
Vt
PP
in the car
drove
down the street


VP
Vt
PP
drove
down
NP
the
N̄
street
PP
in the car

Two analyses for: John was believed to have been shot by Bill

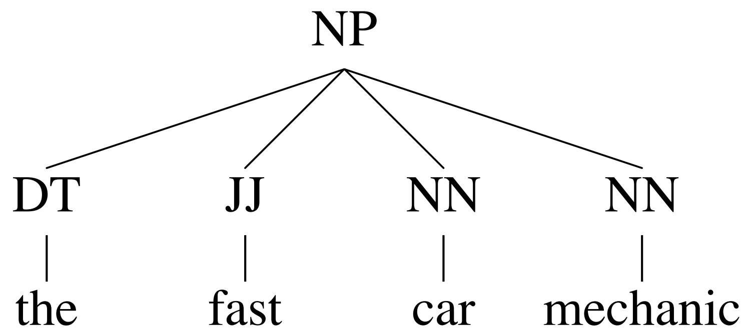
Sources of Ambiguity: Noun Premodifiers

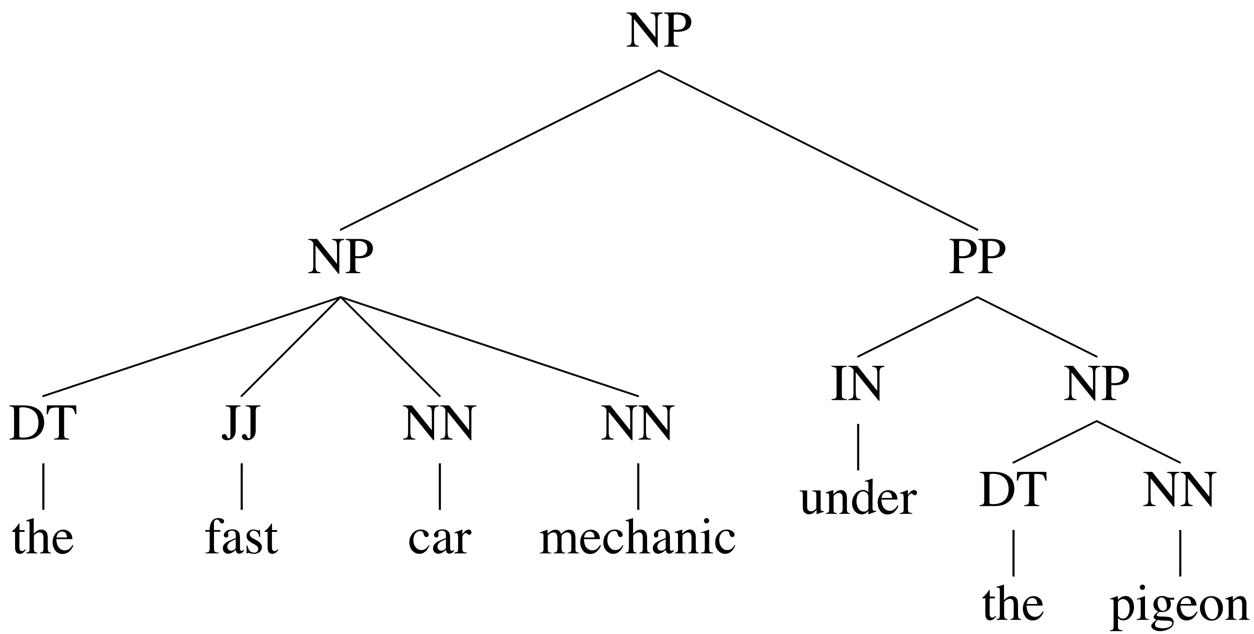
- Noun premodifiers:



A Funny Thing about the Penn Treebank

Leaves NP premodifier structure flat, or underspecified:





A Probabilistic Context-Free Grammar

S	$\Rightarrow$	NP	VP	1.0
VP	$\Rightarrow$	Vi		0.4
VP	$\Rightarrow$	Vt	NP	0.4
VP	$\Rightarrow$	VP	PP	0.2
NP	$\Rightarrow$	DT	NN	0.3
NP	$\Rightarrow$	NP	PP	0.7
PP	$\Rightarrow$	P	NP	1.0

Vi	$\Rightarrow$	sleeps	1.0
Vt	$\Rightarrow$	saw	1.0
NN	$\Rightarrow$	man	0.7
NN	$\Rightarrow$	woman	0.2
NN	$\Rightarrow$	telescope	0.1
DT	$\Rightarrow$	the	1.0
IN	$\Rightarrow$	with	0.5
IN	$\Rightarrow$	in	0.5

- Probability of a tree with rules $\alpha_i \rightarrow \beta_i$ is $\prod_i P(\alpha_i \rightarrow \beta_i | \alpha_i)$

DERIVATION	RULES USED	PROBABILITY
S		

DERIVATION	RULES USED	PROBABILITY
S	S $\rightarrow$ NP VP	1.0
NP VP		

DERIVATION	RULES USED	PROBABILITY
S	S $\rightarrow$ NP VP	1.0
NP VP	NP $\rightarrow$ DT N	0.3
DT N VP		

DERIVATION	RULES USED	PROBABILITY
S	S $\longrightarrow$ NP VP	1.0
NP VP	NP $\longrightarrow$ DT N	0.3
DT N VP	DT $\longrightarrow$ the	1.0
the N VP		

DERIVATION	RULES USED	PROBABILITY
S	S $\rightarrow$ NP VP	1.0
NP VP	NP $\rightarrow$ DT N	0.3
DT N VP	DT $\rightarrow$ the	1.0
the N VP	N $\rightarrow$ dog	0.1
the dog VP		

DERIVATION	RULES USED	PROBABILITY
S	S $\rightarrow$ NP VP	1.0
NP VP	NP $\rightarrow$ DT N	0.3
DT N VP	DT $\rightarrow$ the	1.0
the N VP	N $\rightarrow$ dog	0.1
the dog VP	VP $\rightarrow$ VB	0.4
the dog VB		

DERIVATION	RULES USED	PROBABILITY
S	S $\rightarrow$ NP VP	1.0
NP VP	NP $\rightarrow$ DT N	0.3
DT N VP	DT $\rightarrow$ the	1.0
the N VP	N $\rightarrow$ dog	0.1
the dog VP	VP $\rightarrow$ VB	0.4
the dog VB	VB $\rightarrow$ laughs	0.5
the dog laughs		

TOTAL PROBABILITY = $1.0 \times 0.3 \times 1.0 \times 0.1 \times 0.4 \times 0.5$