

Introduction to Parsing

Parsing
ISCL-BA-06

Çağrı Çöltekin
`ccoltekin@sfs.uni-tuebingen.de`

University of Tübingen
Seminar für Sprachwissenschaft

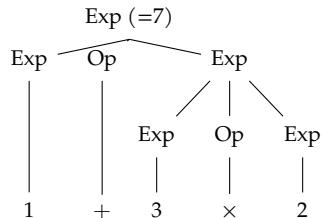
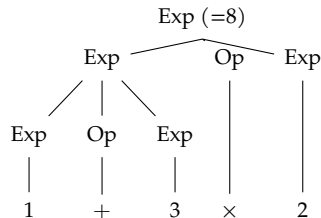
Winter Semester 2020/21

What is parsing?

- Parsing is the task of assigning a structure to a given sentence
- It is related to recognition: typically we follow the steps taken during derivation to obtain the structure
- From a different perspective, parsing is the inverse of the generation task
- Note: we focus on context-free parsing – the structures we build/recover are trees

Why do we need parsing?

- The formal approach to languages as sets emphasizes recognition
 - a string is whether in the language or not
- Parsing is in general a step for semantics
 - we cannot assign semantics without structure



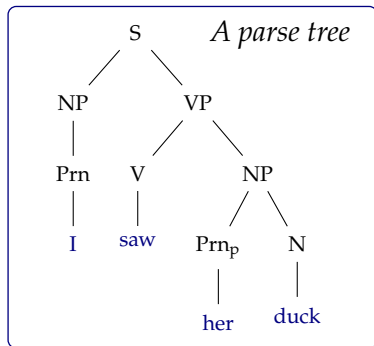
Overview

- Representation context-free analyses and parse trees
- Ambiguity
- Top-down parsing
- Bottom-up parsing
- General overview of the parsing methods
- Representing parsing methods: parse forests
- Parsing and semantics

Overview

- Representation context-free analyses and parse trees
- Ambiguity
- Top-down parsing
- Bottom-up parsing
- General overview of the parsing methods
- Representing parsing methods: parse forests
- Parsing and semantics

Different ways to represent a context-free parse

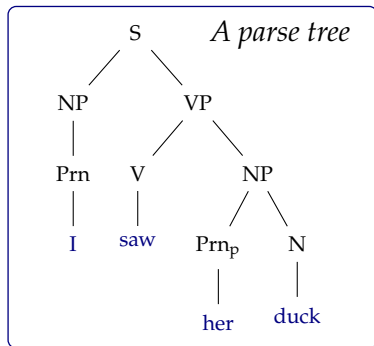


A history of derivations

Sentential form	derivation
S	(start)
NP VP	S \Rightarrow NP VP
Prn VP	NP \Rightarrow Prn
I VP	Prn \Rightarrow I
I V NP	VP \Rightarrow V NP
I saw NP	V \Rightarrow saw
I saw Prn _p N	NP \Rightarrow Prn _p N
I saw her N	Prn _p \Rightarrow her
I saw her duck	N \Rightarrow duck

(Labeled) brackets: $\left[{}_S \left[{}_{NP} \left[{}_{Prn} I \right] \right] \left[{}_{VP} \left[{}_V \text{ saw } \right] \left[{}_{NP} \left[{}_{Prn_p} \text{ her } \right] \left[{}_N \text{ duck } \right] \right] \right] \right]$

Different ways to represent a context-free parse



A history of derivations

Sentential form	derivation
S	(start)
NP VP	S → NP VP
Prn VP	NP → Prn
I VP	Prn → I
I V NP	VP → V NP
I saw NP	V → saw
I saw Prn _p N	NP → Prn _p N
I saw her N	Prn _p → her
I saw her duck	N → duck

(Labeled) brackets: $\left[{}_S \left[{}_{NP} \left[{}_{Prn} I \right] \right] \left[{}_{VP} \left[{}_V \text{ saw } \right] \left[{}_{NP} \left[{}_{Prn_p} \text{ her } \right] \left[{}_N \text{ duck } \right] \right] \right] \right]$

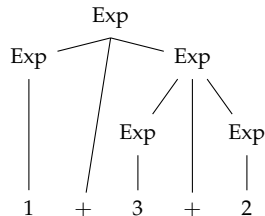
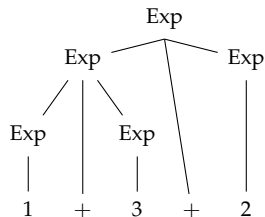
Relation between different representations

- The parse tree and the bracket representation is equivalent
 - parse trees are easier to read by humans
 - brackets are easier for computers
 - brackets are the typical representation for treebanks
- A parse tree (or bracket representation) can be obtained with a different order of production rules

Grammars and ambiguity

$\text{Exp} \rightarrow n$
 $\text{Exp} \rightarrow \text{Exp} + \text{Exp}$
 (terminal symbol 'n' stands for any number)

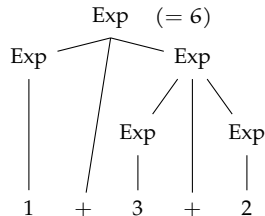
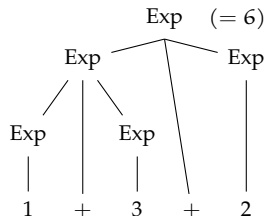
- If a grammar is ambiguous, some sentences produce multiple analyses
- If the resulting analysis lead to the same semantics, the ambiguity is *spurious*



Grammars and ambiguity

$\text{Exp} \rightarrow n$
 $\text{Exp} \rightarrow \text{Exp} + \text{Exp}$
 (terminal symbol 'n' stands for any number)

- If a grammar is ambiguous, some sentences produce multiple analyses
- If the resulting analysis lead to the same semantics, the ambiguity is *spurious*

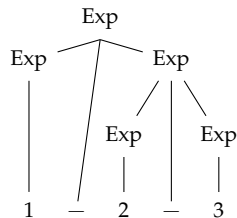
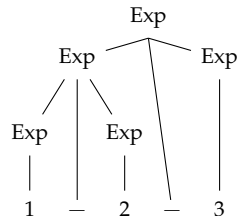


Grammars and ambiguity

$$\begin{aligned}\text{Exp} &\rightarrow n \\ \text{Exp} &\rightarrow \text{Exp} - \text{Exp}\end{aligned}$$

(terminal symbol 'n' stands for any number)

- Is this ambiguity spurious?

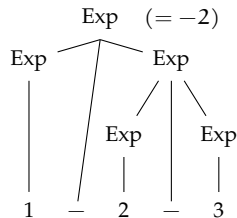
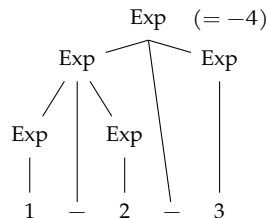


Grammars and ambiguity

$$\begin{aligned}\text{Exp} &\rightarrow n \\ \text{Exp} &\rightarrow \text{Exp} - \text{Exp}\end{aligned}$$

(terminal symbol 'n' stands for any number)

- Is this ambiguity spurious?
- If different structures yield different semantics, the ambiguity is *essential*



Languages and ambiguity

- A language is ambiguous if there is no unambiguous grammar that can produce it
- For example, the language $a^n b^n c^m \cup a^p b^q c^q$ is ambiguous
 - The strings of the form $a^k b^k c^k$ could be generated by either part of the language definition
- Note: do not confuse ambiguity with different derivations leading to same analysis
 - Ambiguity results in different structures
 - Multiple derivations with the same structure is related to the mechanism used for obtaining the derivations

Ambiguity can be removed from a grammar

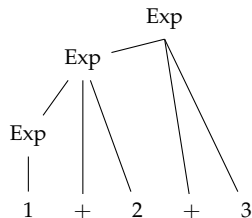
if the language is not ambiguous

$\text{Exp} \rightarrow n$
 $\text{Exp} \rightarrow \text{Exp} + n$
(terminal symbol 'n' stands for any number)

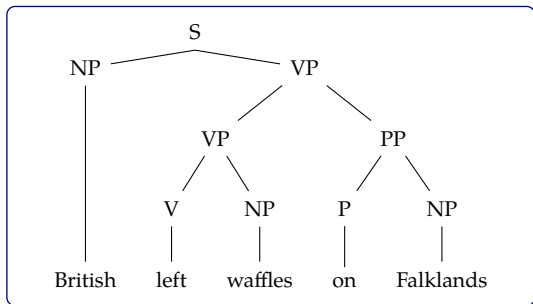
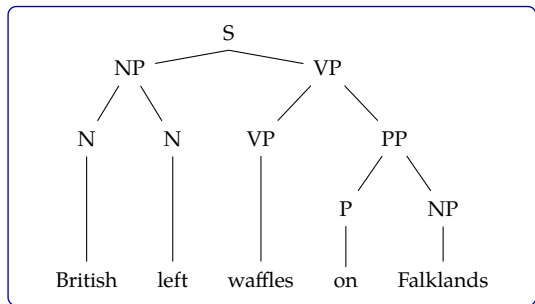
- This one does not have the ambiguity of

$\text{Exp} \rightarrow n$
 $\text{Exp} \rightarrow \text{Exp} + \text{Exp}$

- Both grammars define the same language



Natural languages are ambiguous



- The grammars we define have to distinguish between two different structures

Top-down parsing

general idea

- Start from S , find a sequence of derivations that yield the sentence
- This is simply the same as the generation procedure we discussed earlier
- Attempt to generate all strings from the parse grammar, but allow productions that only leads to the input string

Top-down: demonstration

the cat bites a dog

$S \rightarrow NP VP$

$NP \rightarrow Det N$

$VP \rightarrow V NP$

$VP \rightarrow V$

$Det \rightarrow a$

$Det \rightarrow the$

$N \rightarrow cat$

$N \rightarrow dog$

$V \rightarrow bites$

Top-down: demonstration

S

the cat bites a dog

S → NP VP

NP → Det N

VP → V NP

VP → V

Det → a

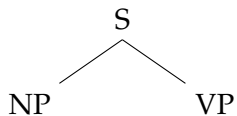
Det → the

N → cat

N → dog

V → bites

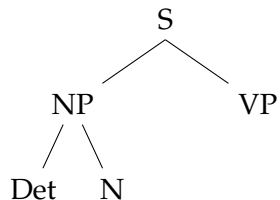
Top-down: demonstration



the cat bites a dog

S	→	NP VP
NP	→	Det N
VP	→	V NP
VP	→	V
Det	→	a
Det	→	the
N	→	cat
N	→	dog
V	→	bites

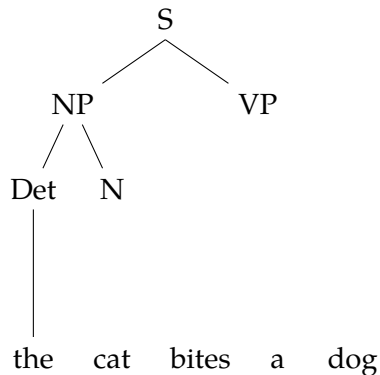
Top-down: demonstration



the cat bites a dog

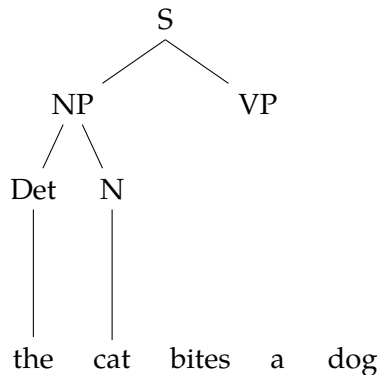
S	→	NP VP
NP	→	Det N
VP	→	V NP
VP	→	V
Det	→	a
Det	→	the
N	→	cat
N	→	dog
V	→	bites

Top-down: demonstration



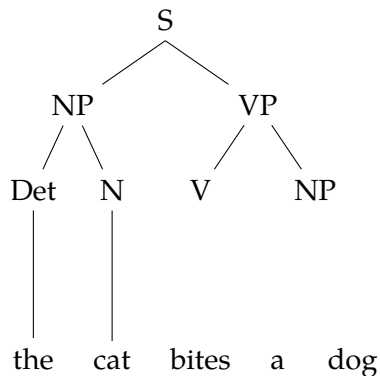
S	→	NP VP
NP	→	Det N
VP	→	V NP
VP	→	V
Det	→	a
Det	→	the
N	→	cat
N	→	dog
V	→	bites

Top-down: demonstration



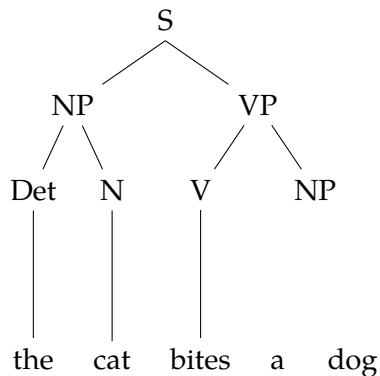
S → NP VP
NP → Det N
VP → V NP
VP → V
Det → a
Det → the
N → cat
N → dog
V → bites

Top-down: demonstration



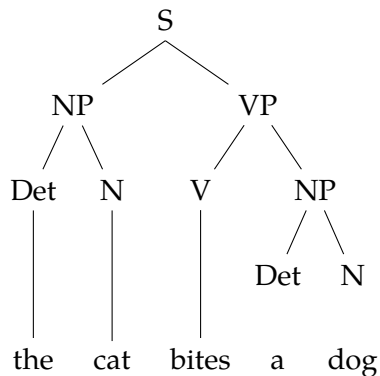
S → NP VP
NP → Det N
VP → V NP
VP → V
Det → a
Det → the
N → cat
N → dog
V → bites

Top-down: demonstration



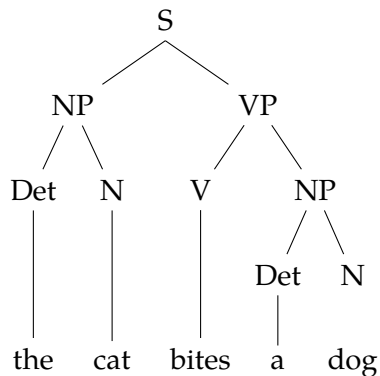
S → NP VP
NP → Det N
VP → V NP
VP → V
Det → a
Det → the
N → cat
N → dog
V → bites

Top-down: demonstration



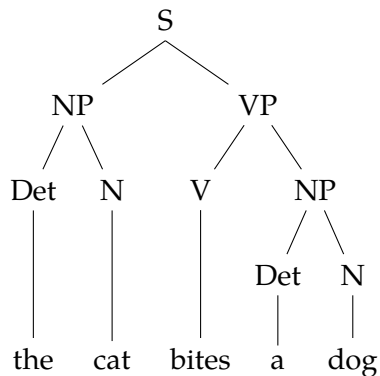
S → NP VP
NP → Det N
VP → V NP
VP → V
Det → a
Det → the
N → cat
N → dog
V → bites

Top-down: demonstration



S → NP VP
 NP → Det N
 VP → V NP
 VP → V
 Det → a
 Det → the
 N → cat
 N → dog
 V → bites

Top-down: demonstration



S \rightarrow NP VP
 NP \rightarrow Det N
 VP \rightarrow V NP
 VP \rightarrow V
 Det \rightarrow a
 Det \rightarrow the
 N \rightarrow cat
 N \rightarrow dog
 V \rightarrow bites

From demonstration to parsing

- There may be multiple production applicable
- We need an automatic mechanism to select the correct productions
- We have two actions:
 - predict generate a hypothesis based on the grammar
 - match when a terminal is produced, check if it matches with the terminal in the expected position
 - if matched, continue
 - otherwise, backtrack
- if we eliminate all non terminals, and the complete input string is matched, then parsing successful

Top-down parsing: another demonstration

the grammar

$S \rightarrow NP VP$

$NP \rightarrow Det N$

$VP \rightarrow V NP$

$VP \rightarrow V$

$Det \rightarrow a$

$Det \rightarrow the$

$N \rightarrow cat$

$N \rightarrow dog$

$V \rightarrow bites$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a } \mathbf{X}$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a } \times$
	Det N VP	$\text{Det} \Rightarrow \text{the } \checkmark$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a} \times$
	Det N VP	$\text{Det} \Rightarrow \text{the} \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog} \times$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a} \times$
	Det N VP	$\text{Det} \Rightarrow \text{the} \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog} \times$
the cat	N VP	$\text{N} \Rightarrow \text{cat} \checkmark$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a} \times$
	Det N VP	$\text{Det} \Rightarrow \text{the} \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog} \times$
the cat	N VP	$\text{N} \Rightarrow \text{cat} \checkmark$
the cat	VP	$\text{VP} \Rightarrow \text{V}$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a } \times$
	Det N VP	$\text{Det} \Rightarrow \text{the } \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog } \times$
the cat	N VP	$\text{N} \Rightarrow \text{cat } \checkmark$
the cat	VP	$\text{VP} \Rightarrow \text{V}$
the cat bites	V	$\text{V} \Rightarrow \text{bites } \checkmark$

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a} \times$
	Det N VP	$\text{Det} \Rightarrow \text{the} \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog} \times$
the cat	N VP	$\text{N} \Rightarrow \text{cat} \checkmark$
the cat	VP	$\text{VP} \Rightarrow \text{V}$
the cat bites	V	$\text{V} \Rightarrow \text{bites} \checkmark$
the cat bites		(not at the end) \times

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

parse: *the cat bites a dog*

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a } \times$
	Det N VP	$\text{Det} \Rightarrow \text{the } \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog } \times$
the cat	N VP	$\text{N} \Rightarrow \text{cat } \checkmark$
the cat	VP	$\text{VP} \Rightarrow \text{V}$
the cat bites	V	$\text{V} \Rightarrow \text{bites } \checkmark$
the cat bites		(not at the end) \times
the cat	V NP	$\text{VP} \Rightarrow \text{V NP}$

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

parse: *the cat bites a dog*

matched	goal	production
	S	$S \Rightarrow \text{NP VP}$
	NP VP	$\text{NP} \Rightarrow \text{Det VP}$
	Det N VP	$\text{Det} \Rightarrow \text{a } \times$
	Det N VP	$\text{Det} \Rightarrow \text{the } \checkmark$
the	N VP	$\text{N} \Rightarrow \text{dog } \times$
the cat	N VP	$\text{N} \Rightarrow \text{cat } \checkmark$
the cat	VP	$\text{VP} \Rightarrow \text{V}$
the cat bites	V	$\text{V} \Rightarrow \text{bites } \checkmark$
the cat bites		(not at the end) \times
the cat	V NP	$\text{VP} \Rightarrow \text{V NP}$
the cat bites	NP	$\text{V} \Rightarrow \text{bites } \checkmark$

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

matched	goal	production
	S	S ⇒ NP VP
	NP VP	NP ⇒ Det VP
	Det N VP	Det ⇒ a ✗
	Det N VP	Det ⇒ the ✓
the	N VP	N ⇒ dog ✗
the cat	N VP	N ⇒ cat ✓
the cat	VP	VP ⇒ V
the cat bites	V	V ⇒ bites ✓
the cat bites		(not at the end) ✗
the cat	V NP	VP ⇒ V NP
the cat bites	NP	V ⇒ bites ✓
the cat bites	Det N	NP ⇒ Det N

parse: *the cat bites a dog*

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

parse: *the cat bites a dog*

matched	goal	production
	S	S ⇒ NP VP
	NP VP	NP ⇒ Det VP
	Det N VP	Det ⇒ a ✗
	Det N VP	Det ⇒ the ✓
the	N VP	N ⇒ dog ✗
the cat	N VP	N ⇒ cat ✓
the cat	VP	VP ⇒ V
the cat bites	V	V ⇒ bites ✓
the cat bites		(not at the end) ✗
the cat	V NP	VP ⇒ V NP
the cat bites	NP	V ⇒ bites ✓
the cat bites	Det N	NP ⇒ Det N
the cat bites a	N	Det ⇒ a ✓

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

parse: *the cat bites a dog*

matched	goal	production
	S	S ⇒ NP VP
	NP VP	NP ⇒ Det VP
	Det N VP	Det ⇒ a ✗
	Det N VP	Det ⇒ the ✓
the	N VP	N ⇒ dog ✗
the cat	N VP	N ⇒ cat ✓
the cat	VP	VP ⇒ V
the cat bites	V	V ⇒ bites ✓
the cat bites		(not at the end) ✗
the cat	V NP	VP ⇒ V NP
the cat bites	NP	V ⇒ bites ✓
the cat bites	Det N	NP ⇒ Det N
the cat bites a	N	Det ⇒ a ✓
the cat bites a dog		Det ⇒ dog ✓

Top-down parsing: another demonstration

the grammar	
S	→ NP VP
NP	→ Det N
VP	→ V NP
VP	→ V
Det	→ a
Det	→ the
N	→ cat
N	→ dog
V	→ bites

parse: *the cat bites a dog*

matched	goal	production
	S	S ⇒ NP VP
	NP VP	NP ⇒ Det VP
	Det N VP	Det ⇒ a ✗
	Det N VP	Det ⇒ the ✓
the	N VP	N ⇒ dog ✗
the cat	N VP	N ⇒ cat ✓
the cat	VP	VP ⇒ V
the cat bites	V	V ⇒ bites ✓
the cat bites		(not at the end) ✗
the cat	V NP	VP ⇒ V NP
the cat bites	NP	V ⇒ bites ✓
the cat bites	Det N	NP ⇒ Det N
the cat bites a	N	Det ⇒ a ✓
the cat bites a dog		Det ⇒ dog ✓

Note that the valid productions yield the parse tree.

Top-down parsing: problems and possible solutions

- Trial-and-error procedure leads to exponential time parsing

Top-down parsing: problems and possible solutions

- Trial-and-error procedure leads to exponential time parsing
- But lots of repeated work: dynamic programming may help avoid it

Top-down parsing: problems and possible solutions

- Trial-and-error procedure leads to exponential time parsing
- But lots of repeated work: dynamic programming may help avoid it
- What happens if we had a rule like

$$\text{NP} \rightarrow \text{NP PP}$$

Top-down parsing: problems and possible solutions

- Trial-and-error procedure leads to exponential time parsing
- But lots of repeated work: dynamic programming may help avoid it
- What happens if we had a rule like

$$\text{NP} \rightarrow \text{NP PP}$$

Top-down parsing: problems and possible solutions

- Trial-and-error procedure leads to exponential time parsing
- But lots of repeated work: dynamic programming may help avoid it
- What happens if we had a rule like

$$\text{NP} \rightarrow \text{NP PP}$$

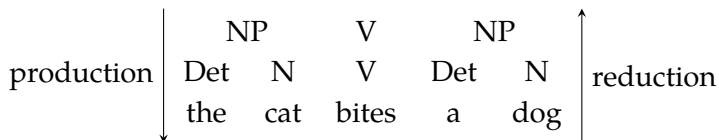
some rules may cause infinite loops

- Notice that if we knew which terminals are possible as the initial part of a non-terminal symbol, we can eliminate the unsuccessful matches earlier

Bottom-up parsing

general idea

- Start from from the input symbol, and try to *reduce* the input to start symbol
- We need to match parts of the sentential form (starting from the input) to the RHS of the grammar rules
- While top-down process relies on *productions* the bottom-up process relies on *reductions*



Bottom-up: demonstration

the cat bites a dog

$S \rightarrow NP VP$

$NP \rightarrow Det N$

$VP \rightarrow V NP$

$VP \rightarrow V$

$Det \rightarrow a$

$Det \rightarrow the$

$N \rightarrow cat$

$N \rightarrow dog$

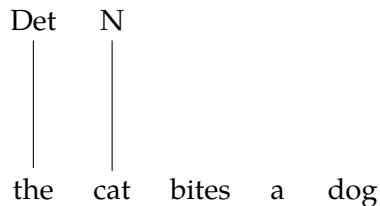
$V \rightarrow bites$

Bottom-up: demonstration

Det
|
the cat bites a dog

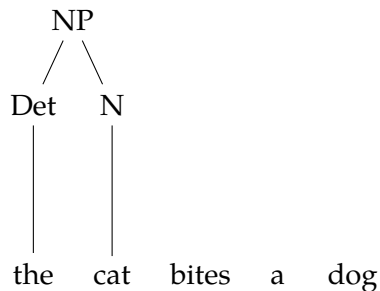
S → NP VP
NP → Det N
VP → V NP
VP → V
Det → a
Det → the
N → cat
N → dog
V → bites

Bottom-up: demonstration



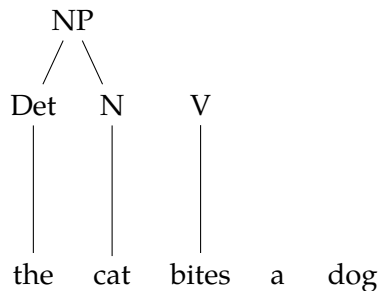
S \rightarrow NP VP
NP \rightarrow Det N
VP \rightarrow V NP
VP \rightarrow V
Det \rightarrow a
Det \rightarrow the
N \rightarrow cat
N \rightarrow dog
V \rightarrow bites

Bottom-up: demonstration



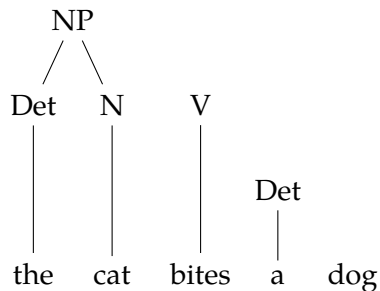
S \rightarrow NP VP
NP \rightarrow Det N
VP \rightarrow V NP
VP \rightarrow V
Det \rightarrow a
Det \rightarrow the
N \rightarrow cat
N \rightarrow dog
V \rightarrow bites

Bottom-up: demonstration



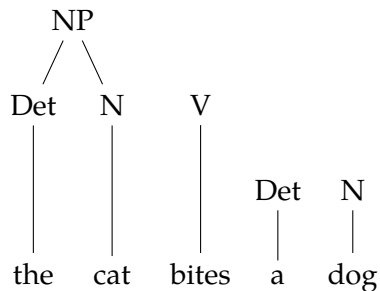
S \rightarrow NP VP
NP \rightarrow Det N
VP \rightarrow V NP
VP \rightarrow V
Det \rightarrow a
Det \rightarrow the
N \rightarrow cat
N \rightarrow dog
V \rightarrow bites

Bottom-up: demonstration



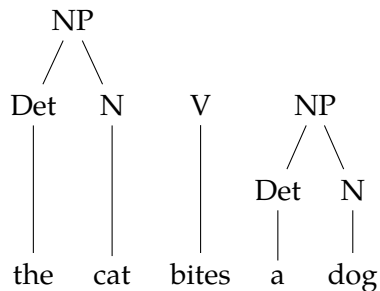
S	→	NP VP
NP	→	Det N
VP	→	V NP
VP	→	V
Det	→	a
Det	→	the
N	→	cat
N	→	dog
V	→	bites

Bottom-up: demonstration



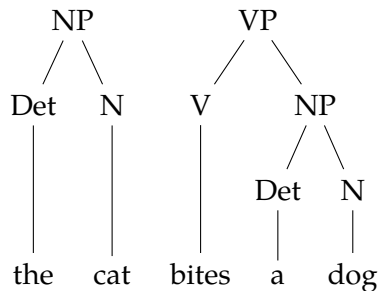
S \rightarrow NP VP
NP \rightarrow Det N
VP \rightarrow V NP
VP \rightarrow V
Det \rightarrow a
Det \rightarrow the
N \rightarrow cat
N \rightarrow dog
V \rightarrow bites

Bottom-up: demonstration



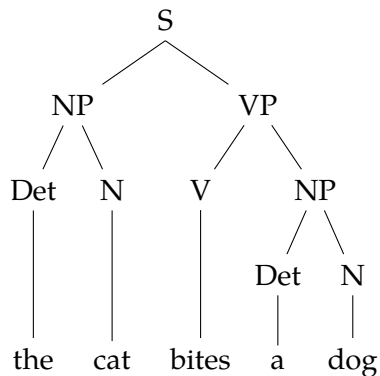
S \rightarrow NP VP
 NP \rightarrow Det N
 VP \rightarrow V NP
 VP \rightarrow V
 Det \rightarrow a
 Det \rightarrow the
 N \rightarrow cat
 N \rightarrow dog
 V \rightarrow bites

Bottom-up: demonstration



S \rightarrow NP VP
 NP \rightarrow Det N
 VP \rightarrow **V NP**
 VP \rightarrow V
 Det \rightarrow a
 Det \rightarrow the
 N \rightarrow cat
 N \rightarrow dog
 V \rightarrow bites

Bottom-up: demonstration



S → NP VP

NP → Det N

VP → V NP

VP → V

Det → a

Det → the

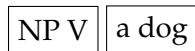
N → cat

N → dog

V → bites

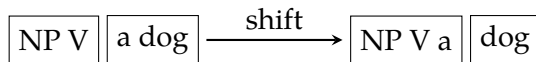
A (first) introduction to shift-reduce parsing

- We keep two data structures:
 - a stack for the (partially) reduced sentential form
 - an input queue that contains only terminal symbols

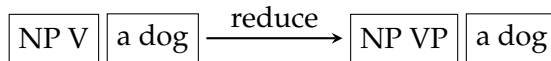


- We use two operations:

shift shifts a terminal to stack



reduce when top symbols on stack match a RHS, replace them with the LHS of the rule



Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
<hr/>		

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP
S	a dog	shift

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP
S	a dog	shift
S a	dog	Det \Rightarrow A

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP
S	a dog	shift
S a	dog	Det \Rightarrow A
S Det dog		N \Rightarrow dog

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP
S	a dog	shift
S a	dog	Det \Rightarrow A
S Det dog		N \Rightarrow dog
S Det N		NP \Rightarrow Det N

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP
S	a dog	shift
S a	dog	Det \Rightarrow A
S Det dog		N \Rightarrow dog
S Det N		NP \Rightarrow Det N
S NP		(stuck)

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the			
Det	cat bites a dog	shift			
Det cat	bites a dog	N \Rightarrow cat			
NP	bites a dog	NP \Rightarrow Det N			
NP	bites a dog	shift			
NP bites	a dog	V \Rightarrow bites			
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the	NP V a	dog	Det \Rightarrow a
Det	cat bites a dog	shift			
Det cat	bites a dog	N \Rightarrow cat			
NP	bites a dog	NP \Rightarrow Det N			
NP	bites a dog	shift			
NP bites	a dog	V \Rightarrow bites			
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the	NP V a	dog	Det \Rightarrow a
Det	cat bites a dog	shift	NP V Det	dog	shift
Det cat	bites a dog	N \Rightarrow cat			
NP	bites a dog	NP \Rightarrow Det N			
NP	bites a dog	shift			
NP bites	a dog	V \Rightarrow bites			
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the	NP V a	dog	Det \Rightarrow a
Det	cat bites a dog	shift	NP V Det	dog	shift
Det cat	bites a dog	N \Rightarrow cat	NP V Det dog		N \Rightarrow dog
NP	bites a dog	NP \Rightarrow Det N			
NP	bites a dog	shift			
NP bites	a dog	V \Rightarrow bites			
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the	NP V a	dog	Det \Rightarrow a
Det	cat bites a dog	shift	NP V Det	dog	shift
Det cat	bites a dog	N \Rightarrow cat	NP V Det dog		N \Rightarrow dog
NP	bites a dog	NP \Rightarrow Det N	NP V Det N		NP \Rightarrow Det N
NP	bites a dog	shift			
NP bites	a dog	V \Rightarrow bites			
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the	NP V a	dog	Det \Rightarrow a
Det	cat bites a dog	shift	NP V Det	dog	shift
Det cat	bites a dog	N \Rightarrow cat	NP V Det dog		N \Rightarrow dog
NP	bites a dog	NP \Rightarrow Det N	NP V Det N		NP \Rightarrow Det N
NP	bites a dog	shift	NP V NP		VP \Rightarrow V NP
NP bites	a dog	V \Rightarrow bites			
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule	stack	input	rule
	the cat bites a dog	shift	NP V	a dog	shift
the	cat bites a dog	Det \Rightarrow the	NP V a	dog	Det \Rightarrow a
Det	cat bites a dog	shift	NP V Det	dog	shift
Det cat	bites a dog	N \Rightarrow cat	NP V Det dog		N \Rightarrow dog
NP	bites a dog	NP \Rightarrow Det N	NP V Det N		NP \Rightarrow Det N
NP	bites a dog	shift	NP V NP		VP \Rightarrow V NP
NP bites	a dog	V \Rightarrow bites	NP VP		S \Rightarrow NP VP
NP V	a dog	VP \Rightarrow V			
NP VP	a dog	S \Rightarrow NP VP			
S	a dog	shift			
S a	dog	Det \Rightarrow A			
S Det dog		N \Rightarrow dog			
S Det N		NP \Rightarrow Det N			
S NP		(stuck)			

Shift-reduce (bottom-up) parsing a demonstration

stack	input	rule
	the cat bites a dog	shift
the	cat bites a dog	Det \Rightarrow the
Det	cat bites a dog	shift
Det cat	bites a dog	N \Rightarrow cat
NP	bites a dog	NP \Rightarrow Det N
NP	bites a dog	shift
NP bites	a dog	V \Rightarrow bites
NP V	a dog	VP \Rightarrow V
NP VP	a dog	S \Rightarrow NP VP
S	a dog	shift
S a	dog	Det \Rightarrow A
S Det dog		N \Rightarrow dog
S Det N		NP \Rightarrow Det N
S NP		(stuck)

stack	input	rule
NP V	a dog	shift
NP V a	dog	Det \Rightarrow a
NP V Det	dog	shift
NP V Det dog		N \Rightarrow dog
NP V Det N		NP \Rightarrow Det N
NP V NP		VP \Rightarrow V NP
NP VP		S \Rightarrow NP VP
S		(done)

- All input reduced to S, accept
- Rules form the parse tree

Summary

- Parsing can be formulated as a top-down or bottom-up search (the search may also be depth-first or breadth first)
- Naive parsing algorithms are inefficient (exponential time complexity)
- There are some directions: dynamic programming, filtering
- Suggested reading for this part: Grune and Jacobs (2007, ch.3)

Summary

- Parsing can be formulated as a top-down or bottom-up search (the search may also be depth-first or breadth first)
- Naive parsing algorithms are inefficient (exponential time complexity)
- There are some directions: dynamic programming, filtering
- Suggested reading for this part: Grune and Jacobs (2007, ch.3)

Next:

- Bottom-up chart parsing: CKY algorithm
- Suggested reading: Grune and Jacobs (2007, section 4.2), Jurafsky and Martin (2009, draft 3rd ed, section 13.2)

Acknowledgments, references, additional reading material

- Please read Grune and Jacobs (2007) chapter 3, a big part part of the lecture follows this chapter



Grune, D. and C.J.H. Jacobs (2007). *Parsing Techniques: A Practical Guide*. second. Monographs in Computer Science. The first edition is available at http://dickgrune.com/Books/PTAPG_1st_Edition/BookBody.pdf. Springer New York. ISBN: 9780387689548.



Jurafsky, Daniel and James H. Martin (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. second. Pearson Prentice Hall. ISBN: 978-0-13-504196-3. URL: <http://web.stanford.edu/~jurafsky/slp3/>.