2. Grammars & Parsing

Language & Logic

Dave Parker

University of Birmingham

2017/18

Recap

- Last lecture:

- Logic – what and why
  - connections to many other areas of CS
- Basic definitions
  - propositions, arguments, validity, soundness

- Please see Canvas for:
  - lecture videos & slides
  - announcements
  - practice quiz

Overview

- Natural vs. formal languages

- Syntax vs. semantics

- Grammars
  - definition, examples, parse trees

Language

- Natural languages vs. formal languages
  - natural language: evolved naturally through human use
    - English, French, Urdu, …
  - formal language: symbols + rules
    - Java, OCaml, propositional logic, …

- For both kinds of languages, we may be interested in:
  - syntax: rules defining allowable sentences/strings of words/symbols
  - semantics: the meaning of a (legal) sentence/string
  - grammar: set of rules defining the syntax
  - parsing: process of analysing a sentence/string according to a grammar

Grammars

- Formally, a grammar is a tuple $G = (V_t, V_n, P, S)$ where:
  - V_t is a set of terminal symbols (or terminals)
  - V_n is a set of non-terminal symbols (or non-terminals)
  - P is a set of production rules
  - S is the start symbol (from the set V_n)

- Production rules are of the form $V_n \rightarrow (V_t \cup V_n)^*$
  - i.e., a non-terminal (on the left-hand-side) and a sequence of terminals/non-terminals (on the right-hand side)

- To generate a word in the language defined by G
  - begin with string comprising just the start symbol S
  - repeatedly rewrite the current string using a production rule
  - until no non-terminals remain

Example grammar

- Grammar

  - Non-terminals V_n: N, D

  - Terminals V_t: 0, 1

  - Start symbol S: N

  - Production rules P:

    - $N \rightarrow D$

    - $N \rightarrow N\ D$

    - $D \rightarrow 0$

    - $D \rightarrow 1$

Generating a string:

N
N D
N D D
D D D
D D 1
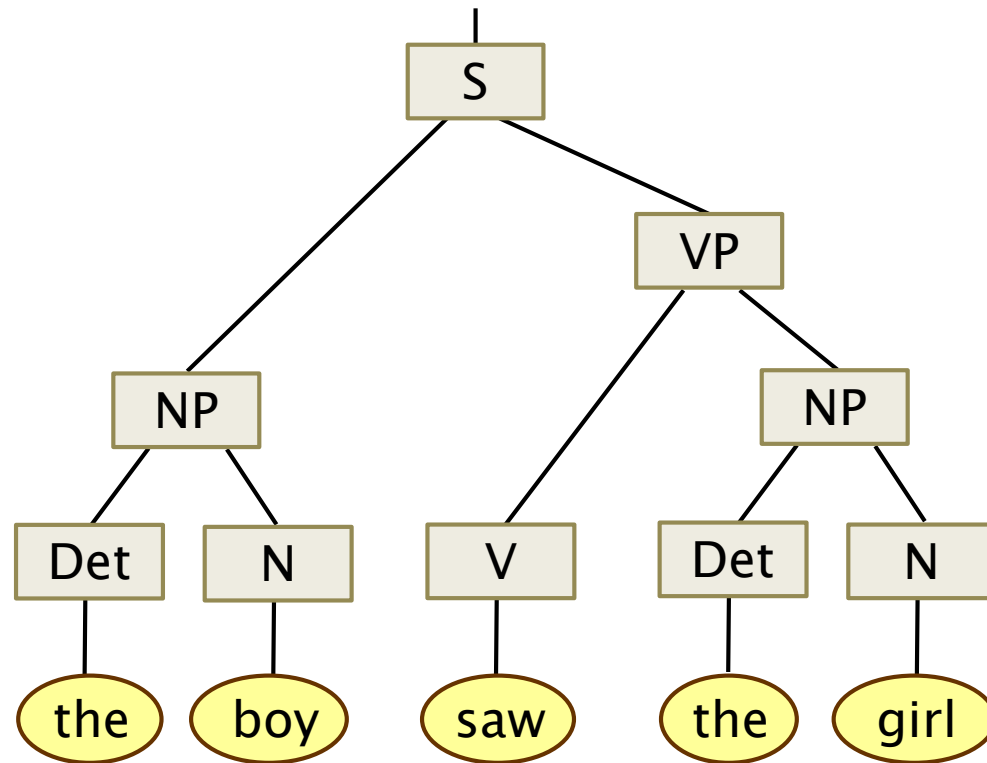D 0 1
1 0 1

Another grammar

- A grammar for a small fragment of English:

S → NP VP	V → [played]
NP → Det N	V → [saw]
NP → Det Adj N	V → [gave]
NP → NP PP	N → [dress]
VP → V NP	N → [telescope]
VP → V NP PP	N → [boy]
PP → P NP	N → [girl]
	P → [with]
	Det → [the]
	Adj → [yellow]

- Sentences
  - 1. "the boy saw the girl"
  - 2. "the boy played with the yellow telescope"

Parse trees

- Parse tree
  - tree representing the syntactic structure of a sentence/string

A note on notation

- We often abbreviate the presentation of a grammar
  - by merging rules for each non-terminal using a | ("pipe") symbol

S → NP VP
NP → Det N | Det Adj N | NP PP
VP → V NP | V NP PP
PP → P NP
V → [played] | [saw] | [gave]
N → [dress] | [telescope] | [boy] | [girl]
P → [with]
Det → [the]
Adj → [yellow]

Another parsing example

- Draw a parse tree for this sentence
  - "the boy saw the girl with the telescope"

    S → NP VP
    NP → Det N | Det Adj N | NP PP
    VP → V NP | V NP PP
    PP → P NP
    V → [played] | [saw] | [gave]
    N → [dress] | [telescope] | [boy] | [girl]
    P → [with]
    Det → [the]
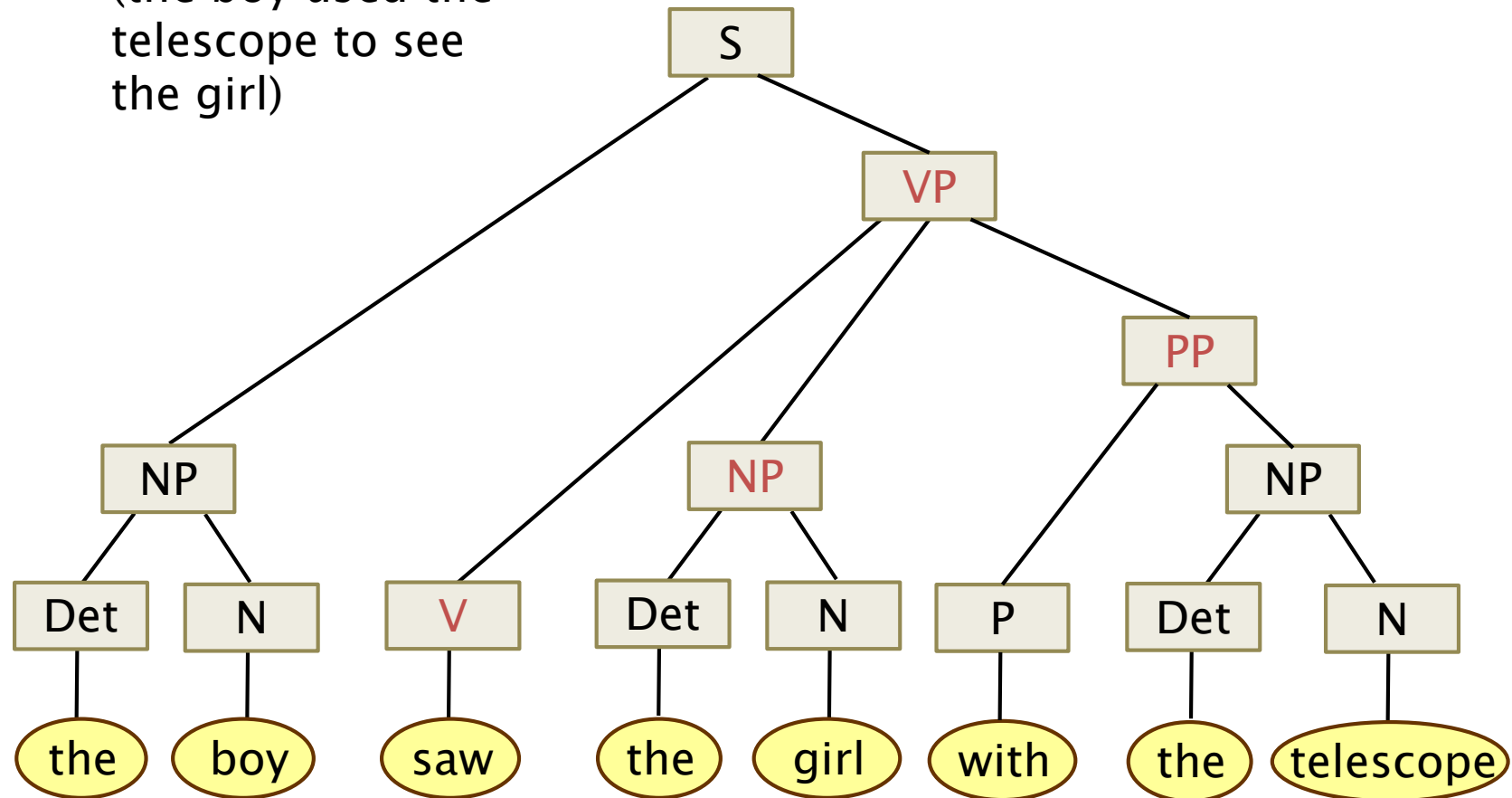   Adj → [yellow]

- Natural languages usually exhibit ambiguity…

The boy saw the girl with the telescope
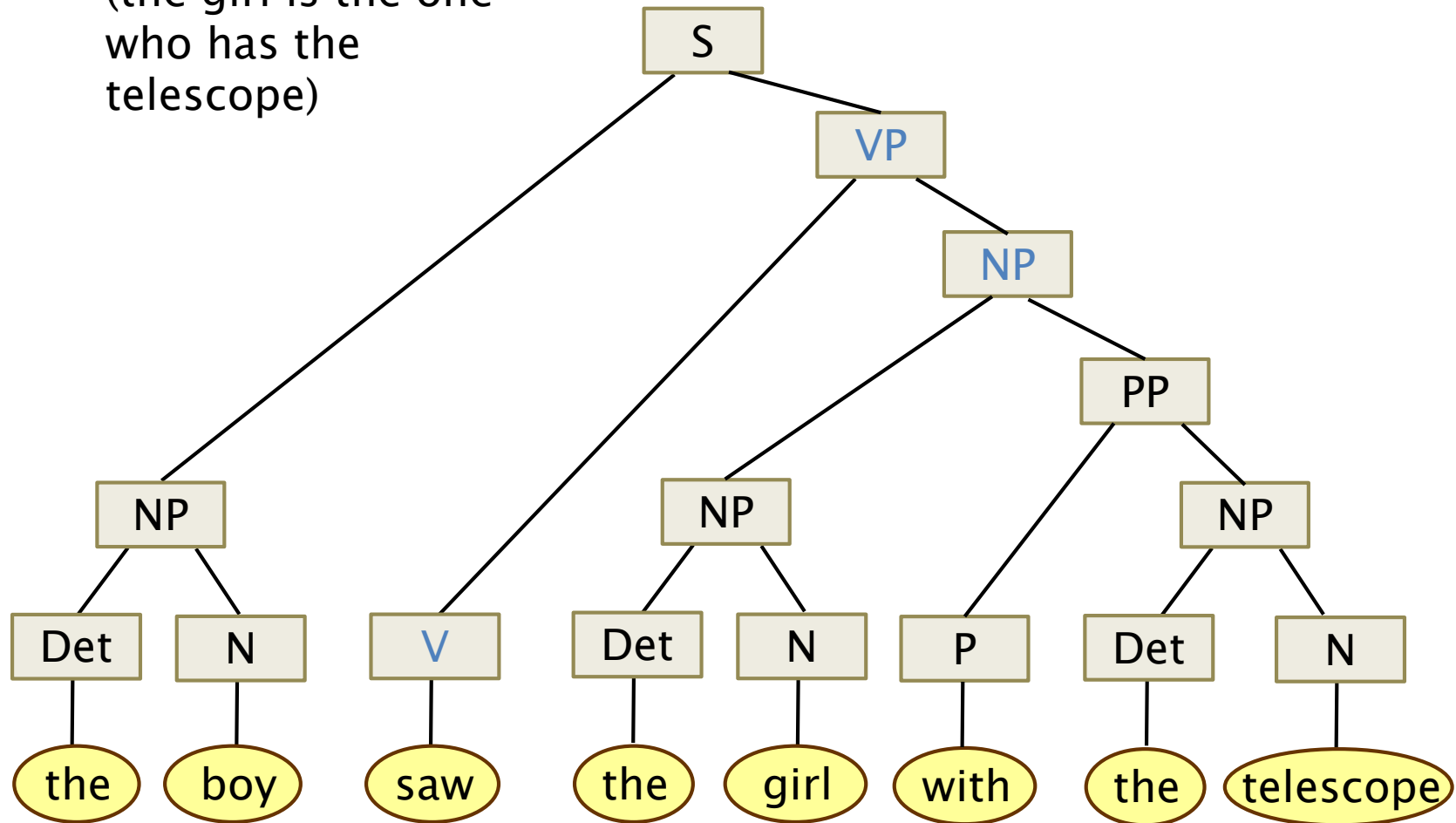
- Parse tree using production rule VP → V NP PP

  – (the boy used the telescope to see the girl)

The boy saw the girl with the telescope

- Parse tree using production rule VP → V NP (and NP → NP PP)
  - (the girl is the one who has the telescope)

Overview

- Natural vs. formal languages

- Syntax vs. semantics

- Grammars
  - terminals, non-terminals, rules + start symbol
  - generation of a string/sentence by a grammar
  - rewritings, parse trees
  - ambiguity

Next week

- Mon 4pm: lecture (propositional logic)

- Tue 11am / Thur 10am: class (exercises/discussion)
  - on Tuesday, if your surname is in the range A–J (by default)
  - on Thursday, if your surname is in the range K–Z (by default)

- Reminder
  - take a look at the practice quiz(zes)…