Ling 566
Oct 4, 2016

Context-Free Grammar

Overview

- Formal definition of CFG

- Constituency, ambiguity, constituency tests

- Central claims of CFG

- Weaknesses of CFG

- Reading questions

What does a theory do?

- Monolingual

  - Model grammaticality/acceptability

  - Model relationships between sentences (internal structure)

- Multilingual

  - Model relationships between languages

  - Capture generalizations about possible languages

Summary

- Grammars as lists of sentences:

  - Runs afoul of creativity of language

- Grammars as finite-state machines:

  - No representation of structural ambiguity

  - Misses generalizations about structure

  - (Not formally powerful enough)

Chomsky Hierarchy

Type 0 Languages

Context-Sensitive Languages

Context-Free Languages

Regular Languages

Context-Free Grammar

- A quadruple: $< C, \Sigma, P, S >$

  - C: set of categories

  - Σ: set of terminals (vocabulary)

  - P: set of rewrite rules $\alpha \rightarrow \beta_1, \beta_2, \ldots, \beta_n$

  - S in C: start symbol

- For each rule $\alpha \rightarrow \beta_1, \beta_2, \ldots, \beta_n \in P$
  $\alpha \in C;\ \beta_i \in C \cup \Sigma;\ 1 \leq i \leq n$

A Toy Grammar

RULES

S $\longrightarrow$ NP VP

NP $\longrightarrow$ (D) A* N PP*

VP $\longrightarrow$ V (NP) (PP)

PP $\longrightarrow$ P NP

LEXICON

D: the, some

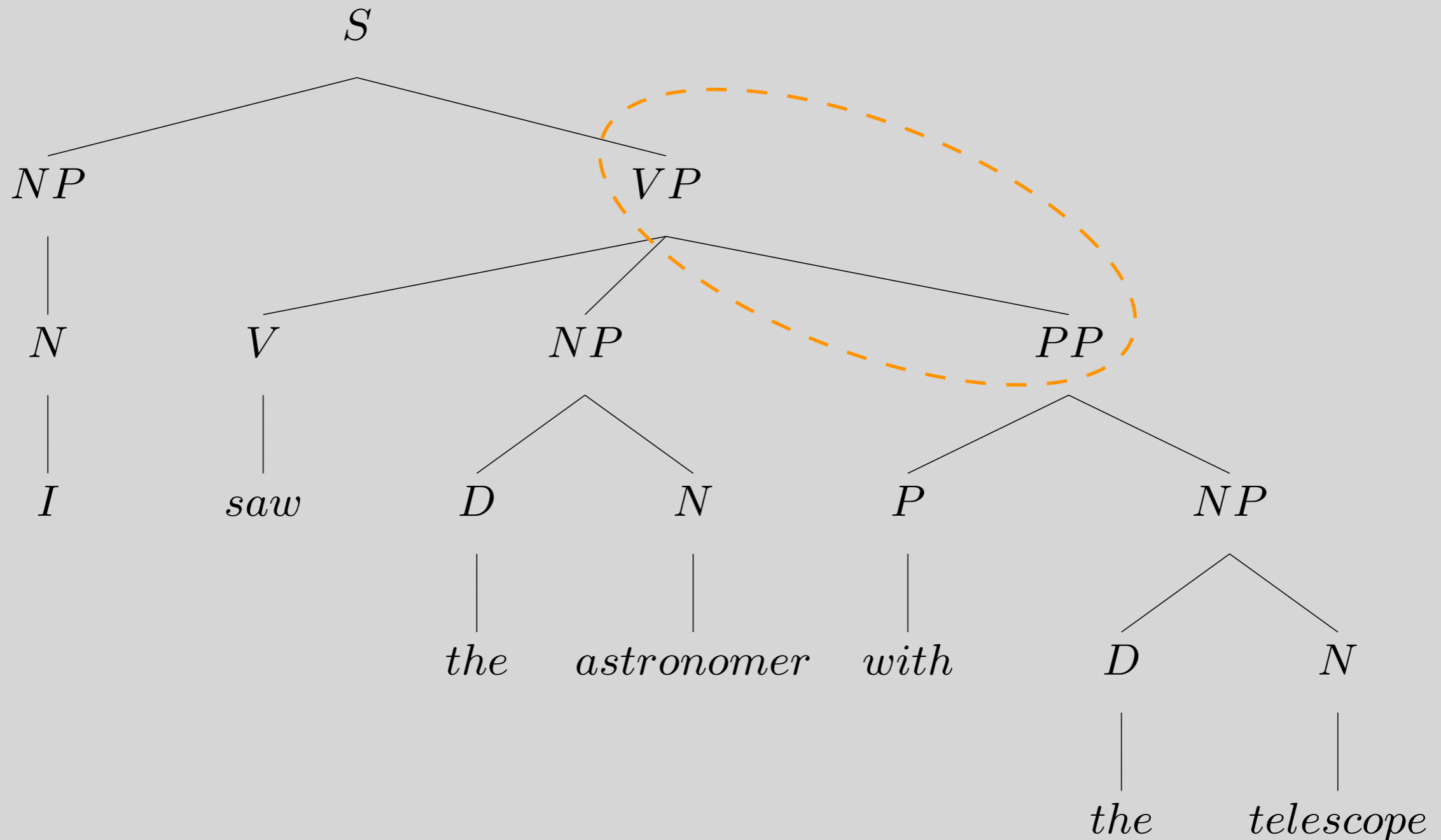A: big, brown, old

N: birds, fleas, dog, hunter, I
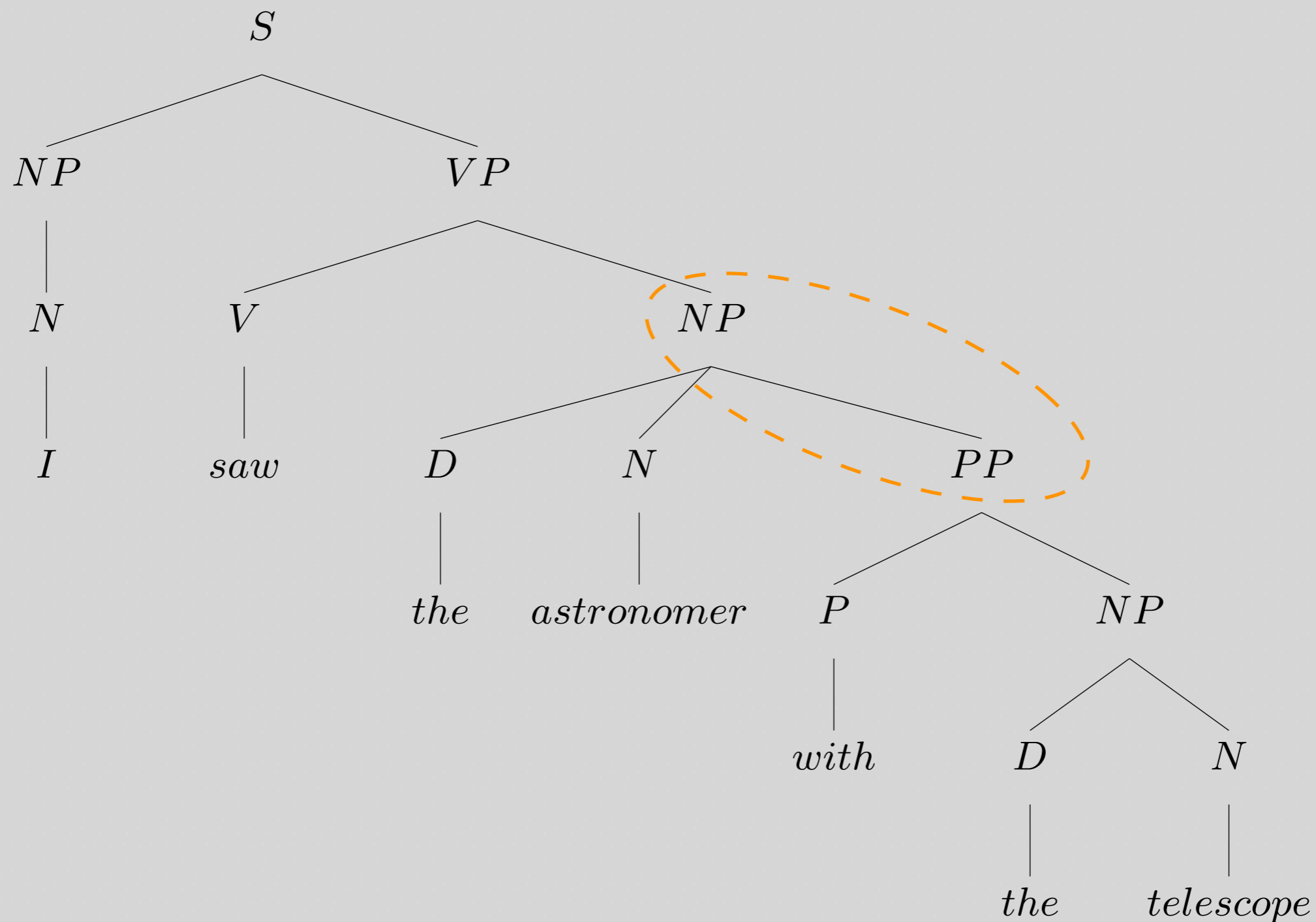
V: attack, ate, watched

P: for, beside, with

Structural Ambiguity

I saw the astronomer with the telescope.

Structure 1: PP under VP

Structure 1: PP under NP

Constituents

- How do constituents help us? (What's the point?)

- What aspect of the grammar determines which words will be modeled as a constituent?

- How do we tell which words to group together into a constituent?

- What does the model claim or predict by grouping words together into a constituent?

Constituency Tests

- Recurrent Patterns

  The quick brown fox with the bushy tail jumped over *the lazy brown dog with one ear.*

- Coordination

  The quick brown fox with the bushy tail and *the lazy brown dog with one ear* are friends.

- Sentence-initial position

  The election of 2000, everyone will remember for a long time.

- Cleft sentences

  It was *a book about syntax* *they were reading.*

General Types of Constituency Tests

- Distributional

- Intonational

- Semantic

- Psycholinguistic

... but they don't always agree.

Central claims implicit in CFG formalism:

1. Parts of sentences (larger than single words) are linguistically significant units, i.e. phrases play a role in determining meaning, pronunciation, and/or the acceptability of sentences.

2. Phrases are contiguous portions of a sentence (no discontinuous constituents).

3. Two phrases are either disjoint or one fully contains the other (no partially overlapping constituents).

4. What a phrase can consist of depends only on what kind of a phrase it is (that is, the label on its top node), not on what appears around it.

- Claims 1-3 characterize what is called 'phrase structure grammar'

- Claim 4 (that the internal structure of a phrase depends only on what type of phrase it is, not on where it appears) is what makes it 'context-free'.

- There is another kind of phrase structure grammar called 'context-sensitive grammar' (CSG) that gives up 4. That is, it allows the applicability of a grammar rule to depend on what is in the neighboring environment. So rules can have the form A→X, in the context of Y_Z.

Possible Counterexamples

- To Claim 2 (no discontinuous constituents):

A technician arrived who could solve the problem.

- To Claim 3 (no overlapping constituents):

I read what was written about me.

- To Claim 4 (context independence):

- *He arrives this morning.*
- **He arrive this morning.*
- **They arrives this morning.*

A Trivial CFG

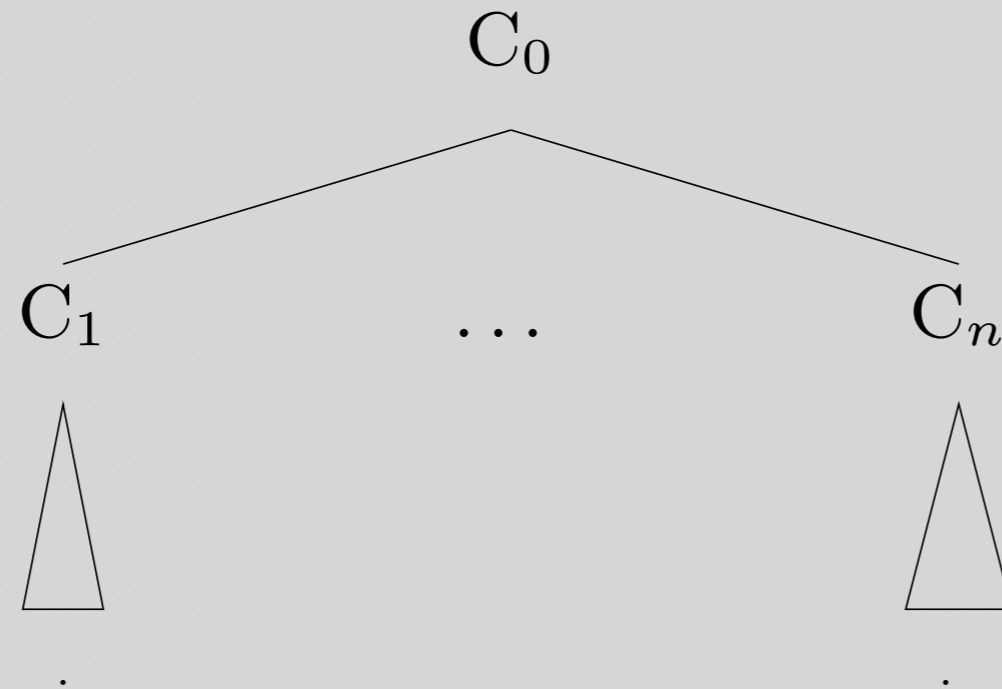S $\rightarrow$ NP VP

NP $\rightarrow$ D N

VP $\rightarrow$ V NP

D: *the*

V: *chased*

N: *dog, cat*

Trees and Rules

C_0

C_1 ... C_n

is a well-formed nonlexical tree if (and only if)

C_1 , ... , C_n are well-formed trees, and

$C_0 \rightarrow C_1 \ldots Cn$ is a grammar rule.

Bottom-up Tree Construction

D: *the*

V: *chased*

N: *dog, cat*

```
  D           V           N           N
  |           |           |           |
 the        chased       dog         cat
```
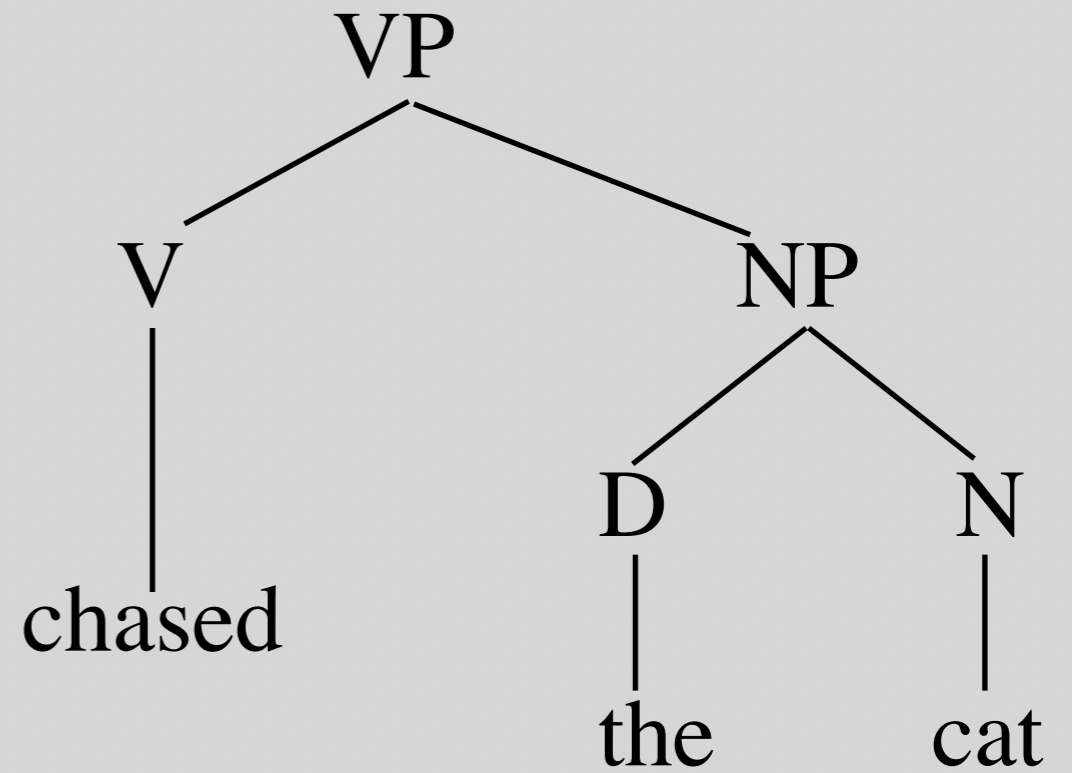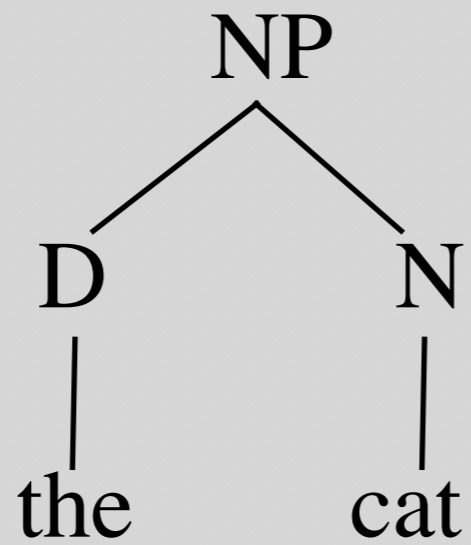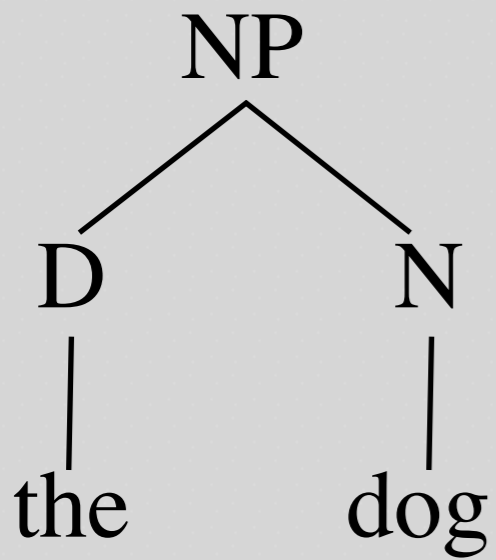
NP $\longrightarrow$ D N   VP $\longrightarrow$ V NP

```
        NP                    NP                        VP
       /  \                  /  \                      /  \
      D    N                D    N                    V    NP
      |    |                |    |                    |    / \
     the  dog              the  cat               chased  D   N
                                                          |   |
                                                         the  cat
```

$$S \longrightarrow NP \ VP$$

```
                          S
                 _____/ _____
                /                   \
              NP                     VP
            __/ \__              ___/  \___
           /       \            /          \
          D         N          V            NP
          |         |          |          _/  \_
          |         |          |         /      \
         the       dog      chased      D        N
                                        |        |
                                        |        |
                                       the      cat
```
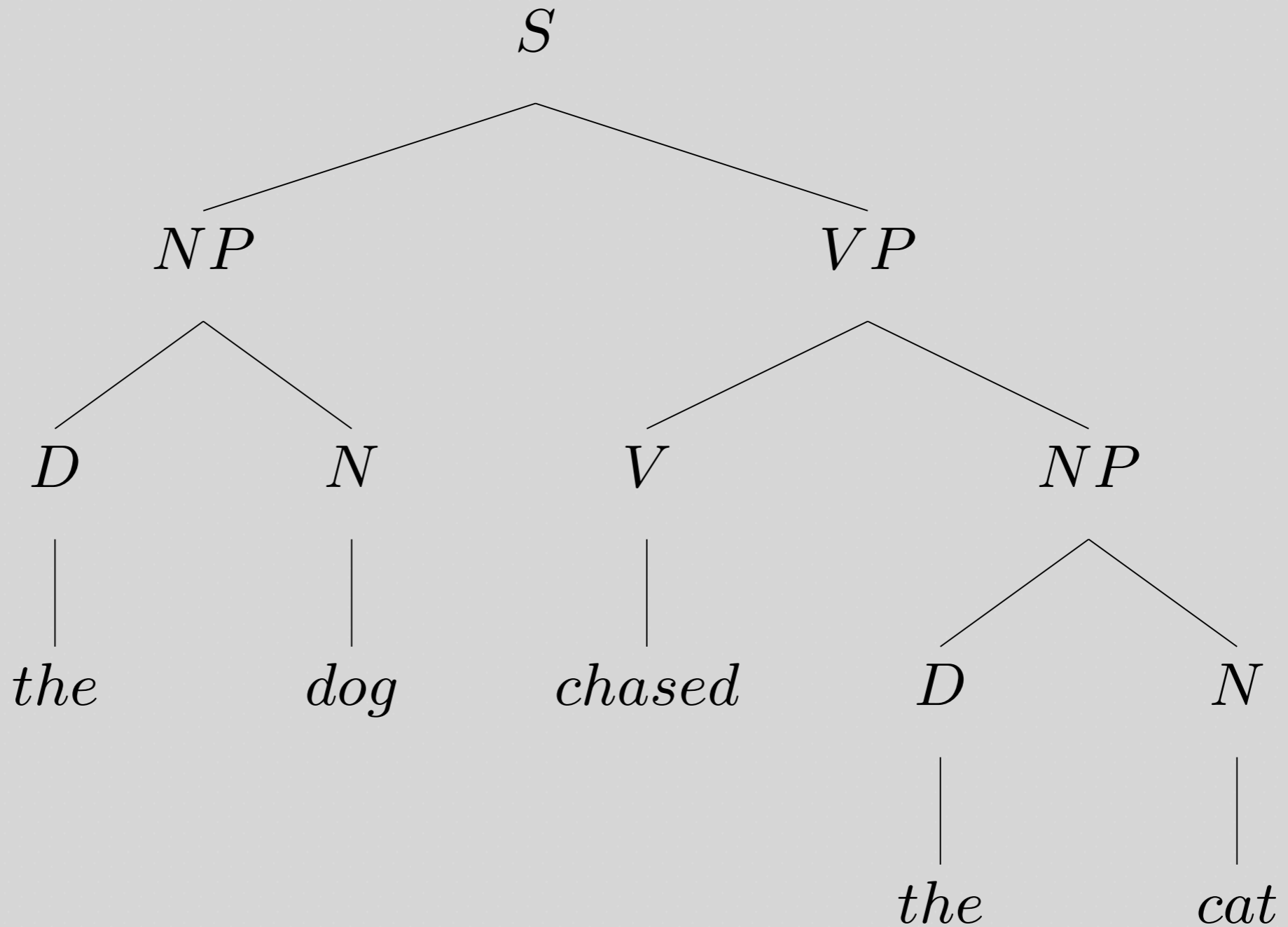
Top-down Tree Construction

S ⟶ NP VP   NP ⟶ D N   VP ⟶ V NP

```
        S                    NP                    VP
       / \                  /  \                  /  \
     NP   VP               D    N                V    NP
                         (twice)
```

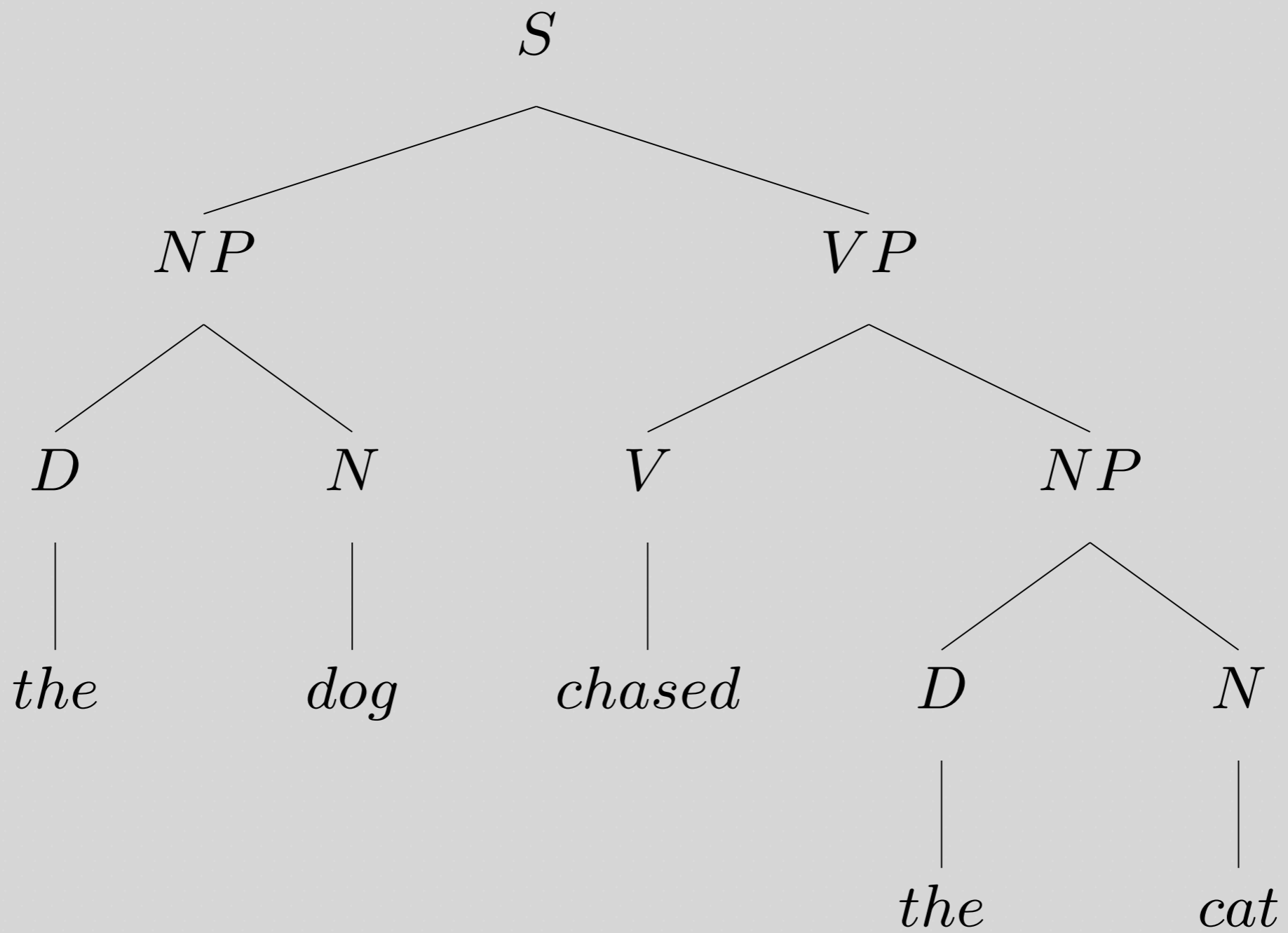D   V   N   N
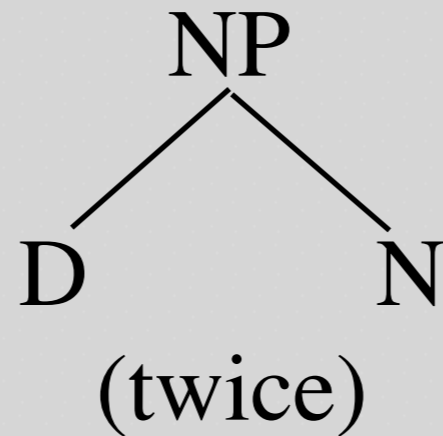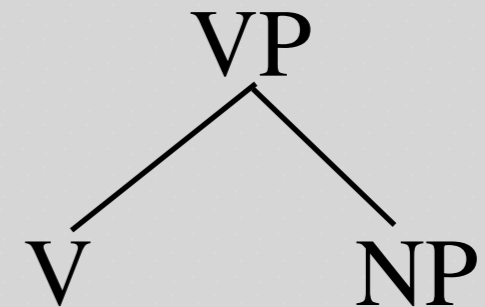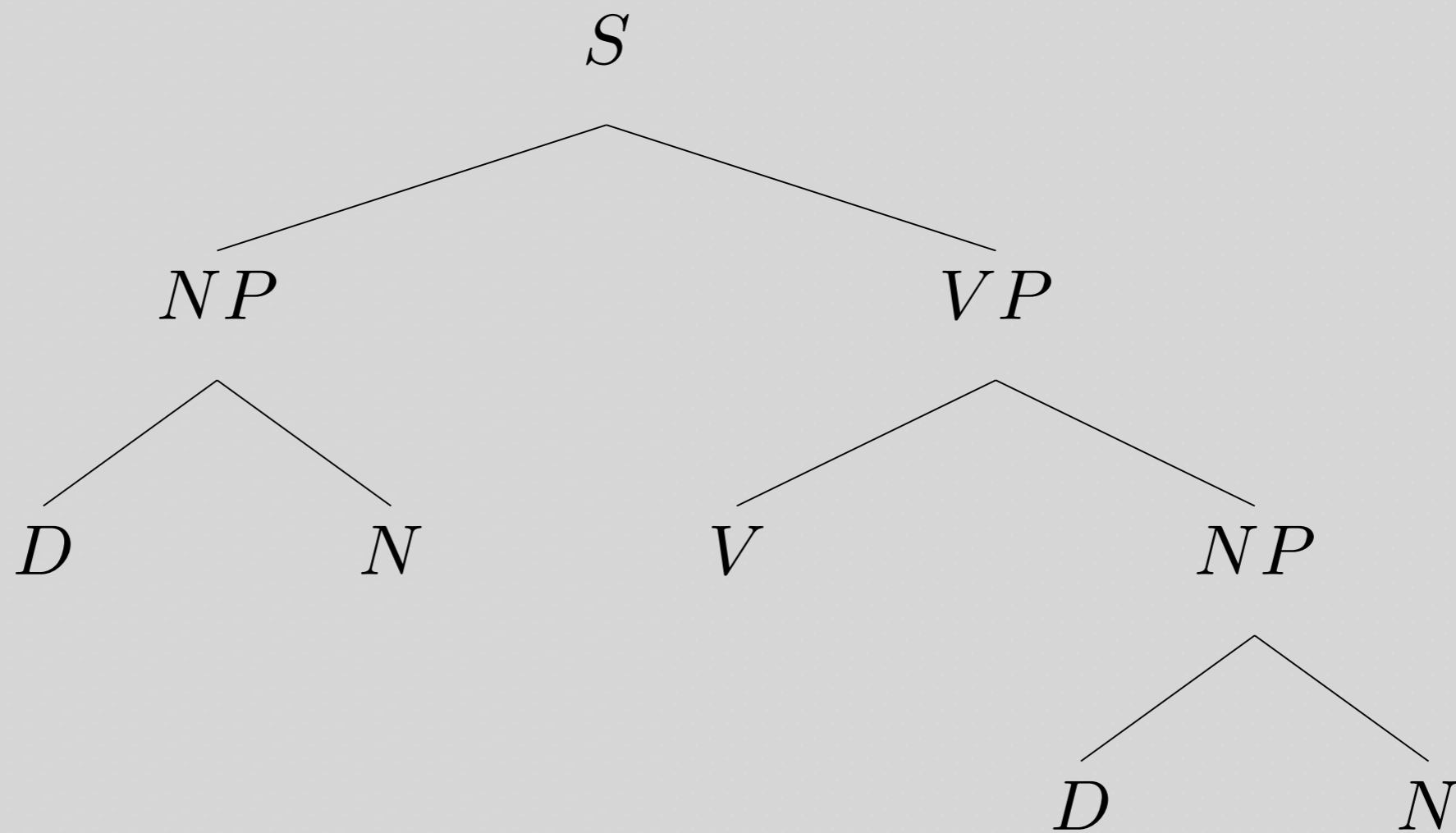
the   chased   dog   cat

Weaknesses of CFG (atomic node labels)

- It doesn't tell us what constitutes a linguistically natural rule

$$VP \rightarrow P \ NP$$
$$NP \rightarrow VP \ S$$

- Rules get very cumbersome once we try to deal with things like agreement and transitivity.

- It has been argued that certain languages (notably Swiss German and Bambara) contain constructions that are provably beyond the descriptive capacity of CFG.

Agreement & Transitivity

S	→	NP-SG VP-SG	VP-SG	→	IV-SG
S	→	NP-PL VP-PL	VP-PL	→	IV-PL
NP-SG	→	(D) NOM-SG	VP-SG	→	TV-SG NP
NP-PL	→	(D) NOM-PL	VP-PL	→	TV-PL NP
NOM-SG	→	NOM-SG PP	VP-SG	→	DTV-SG NP NP
NOM-PL	→	NOM-PL PP	VP-PL	→	DTV-PL NP NP
NOM-SG	→	N-SG	VP-SG	→	CCV-SG S
NOM-PL	→	N-PL	VP-PL	→	CCV-PL S
NP	→	NP-SG	VP-SG	→	VP-SG PP
NP	→	NP-PL	VP-PL	→	VP-PL PP
	…			…	

Shieber 1985

- Swiss German example:

... mer d'chind   em Hans   es huus   lönd hälfe aastriiche
... we   the children-ACC Hans-DAT the hous-ACC let   help paint
... we let the children help Hans paint the house

- Cross-serial dependency:

  - *let* governs case on *children*

  - *help* governs case on *Hans*

  - *paint* governs case on *house*

Shieber 1985

- Define a new language $f(\text{SG})$:

$f(\text{d'chind})$	$=$	a	$f(\text{Jan säit das mer})$	$=$	w
$f(\text{em Hans})$	$=$	b	$f(\text{es huus})$	$=$	x
$f(\text{lönde})$	$=$	c	$f(\text{aastriiche})$	$=$	y
$f(\text{hälfe})$	$=$	d	$f([\text{other}])$	$=$	z

- Let r be the regular language $wa^*b^*xc^*d^*y$

- $f(\text{SG}) \cap r = wa^m b^n x c^m d^n y$

- $wa^m b^n x c^m d^n y$ is not context free.

- But context free languages are closed under intersection.

- $\therefore$ $f(\text{SG})$ (and by extension Swiss German) must not be context free.

Strongly/weakly CF

- A language is *weakly* context-free if the set of strings in the language can be generated by a CFG.

- A language is *strongly* context-free if the CFG furthermore assigns the correct structures to the strings.

- Shieber's argument is that SW is not *weakly* context-free and *a fortiori* not *strongly* context-free.

- Bresnan et al (1983) had already argued that Dutch is *strongly* not context-free, but the

On the other hand....

- It's a simple formalism that can generate infinite languages and assign linguistically plausible structures to them.

- Linguistic constructions that are beyond the descriptive power of CFG are rare.

- It's computationally tractable and techniques for processing CFGs are well understood.

So.....

- CFG has been the starting point for most types of generative grammar.

- The theory we develop in this course is an extension of CFG.

Reading Questions

- Can signed languages be described with CFG?

- Can CFG be used to describe agglutinating languages?

- What is the context that CFGs are free of?

Reading Questions

- What does the superscript + mean in X -> X+ CONJ X?

- What's the difference between Kleene star and parentheses?

  - NP --> (D) A* N PP*

  - VP --> V (NP) (PP)

- If we can use CFG to write rules like PP -> P NP, what does it mean to say that we can't capture headedness?

Reading Questions

- Does HPSG use X-bar theory?

- What is NOM? How is it different from N?

- If N + PP is a NOM (distinct from NP), is there something similar for V + PP distinct from VP?

Reading Questions

- Wouldn't it be more efficient to use features, rather than NP-PL etc?

- How much info do we need to encode about words? What about idioms?

Reading Questions

- What does it mean for a grammar to be able to adequately describe a language (e.g., on page 36)? How would you go about demonstrating that a type of language belonged to a particular level of the Chomsky hierarchy?

- What does it mean to be Turing Complete?

- How do HPSG and Transformational Grammar in terms of the languages they can describe?

- Why model structure and grammaticality with the same system?

Reading Questions

- Does HPSG try to model what's in the wetware?

- Humans seem to need very little computational power to store and utilize vast amounts of information. How do we use a human "data structure" in our computer programs?

- Is it reasonable to assume that NLs have a finite lexicon?

Reading Questions

- How does HPSG differ from other extensions to CFG (e.g. transforations)?

- What makes it better for computational applications?

- Are there other theories that can be modeled computationally?

- How well does HPSG work for non-English languages?

Reading Questions

- How do you create a CFG for a language? Manually? Automatically?

- How many rules do you end up with? How do you evaluate this?

Overview

- Formal definition of CFG

- Constituency, ambiguity, constituency tests

- Central claims of CFG

- Weaknesses of CFG

- Next time: Feature structures