Structural Ambiguity

How different parse trees may be produced from the same sentence or phrase

- Attachment ambiguity: where a constituent may attach to the rest of the tree
  - I saw a man with a telescope
- Coordination ambiguity: How to group the arguments of a conjunction
  - Spicy rice and apples
- *Disambiguation* relies on applying additional knowledge
  - Of language, e.g., what verbs and nouns or prepositions go together
  - Of the real world
  - Of the context, such as prior sentences or conversations

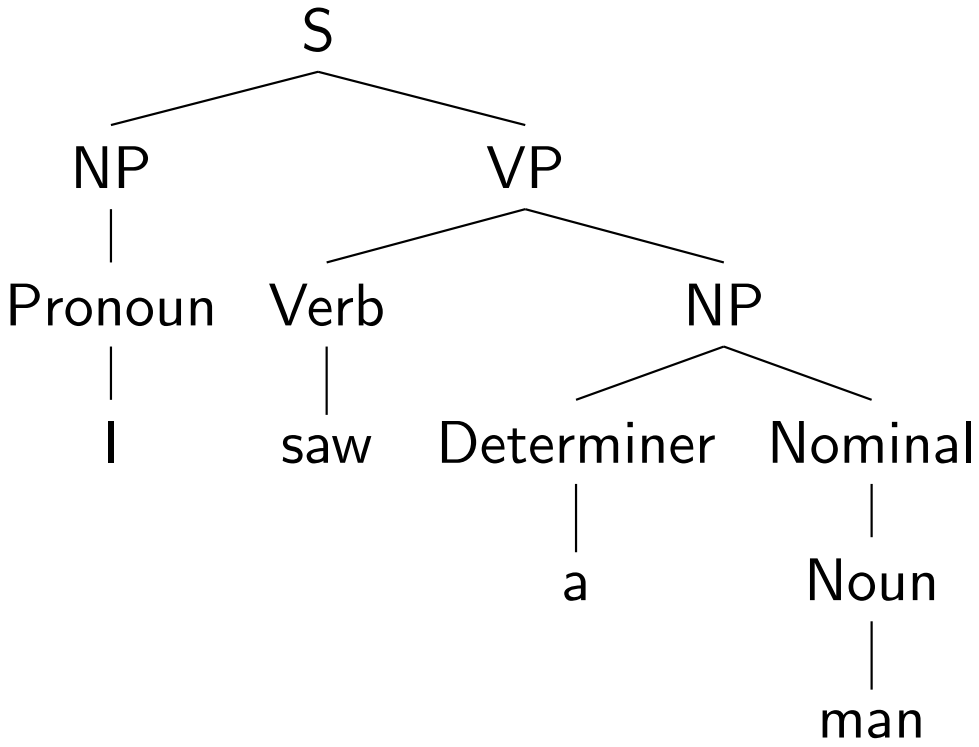
Jurafsky's Miniature Grammar, $\mathscr{L}_1$

Omitting the lexicon

S	$\longrightarrow$	NP VP
S	$\longrightarrow$	Auxiliary-Verb NP VP
S	$\longrightarrow$	VP
NP	$\longrightarrow$	Pronoun
NP	$\longrightarrow$	Proper-Noun
NP	$\longrightarrow$	Determiner Nominal
Nominal	$\longrightarrow$	Noun
Nominal	$\longrightarrow$	Nominal Noun
Nominal	$\longrightarrow$	Nominal PP
VP	$\longrightarrow$	Verb
VP	$\longrightarrow$	Verb NP
VP	$\longrightarrow$	Verb NP PP
VP	$\longrightarrow$	Verb PP
VP	$\longrightarrow$	VP PP
PP	$\longrightarrow$	Preposition NP

Attachment Ambiguity: Setting the Stage

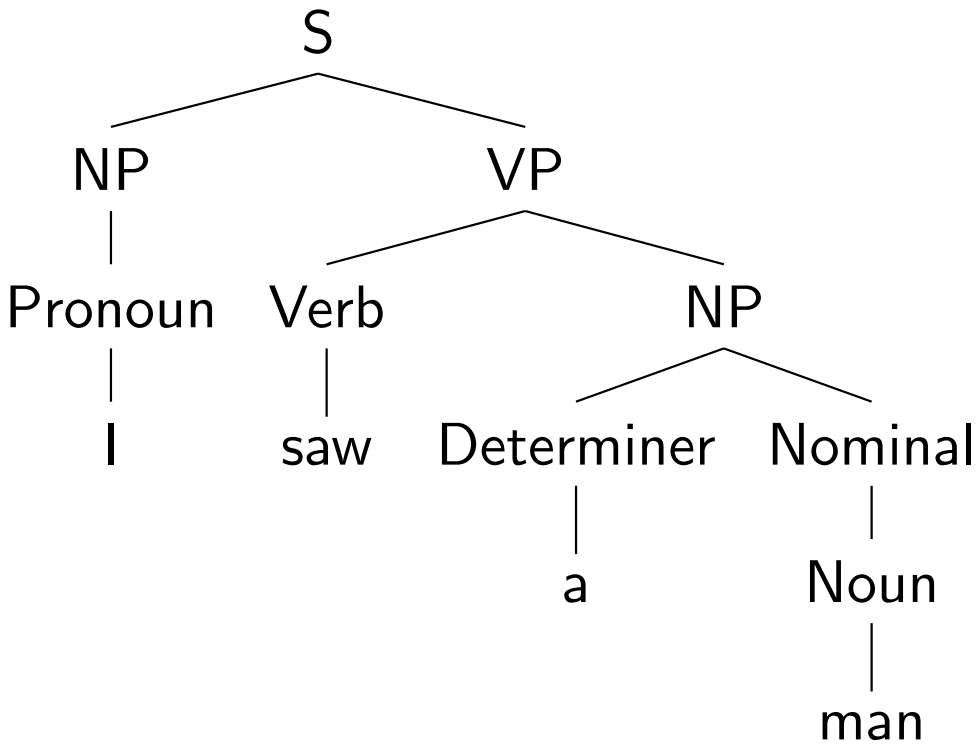
I saw a man



Attachment Ambiguity: Example

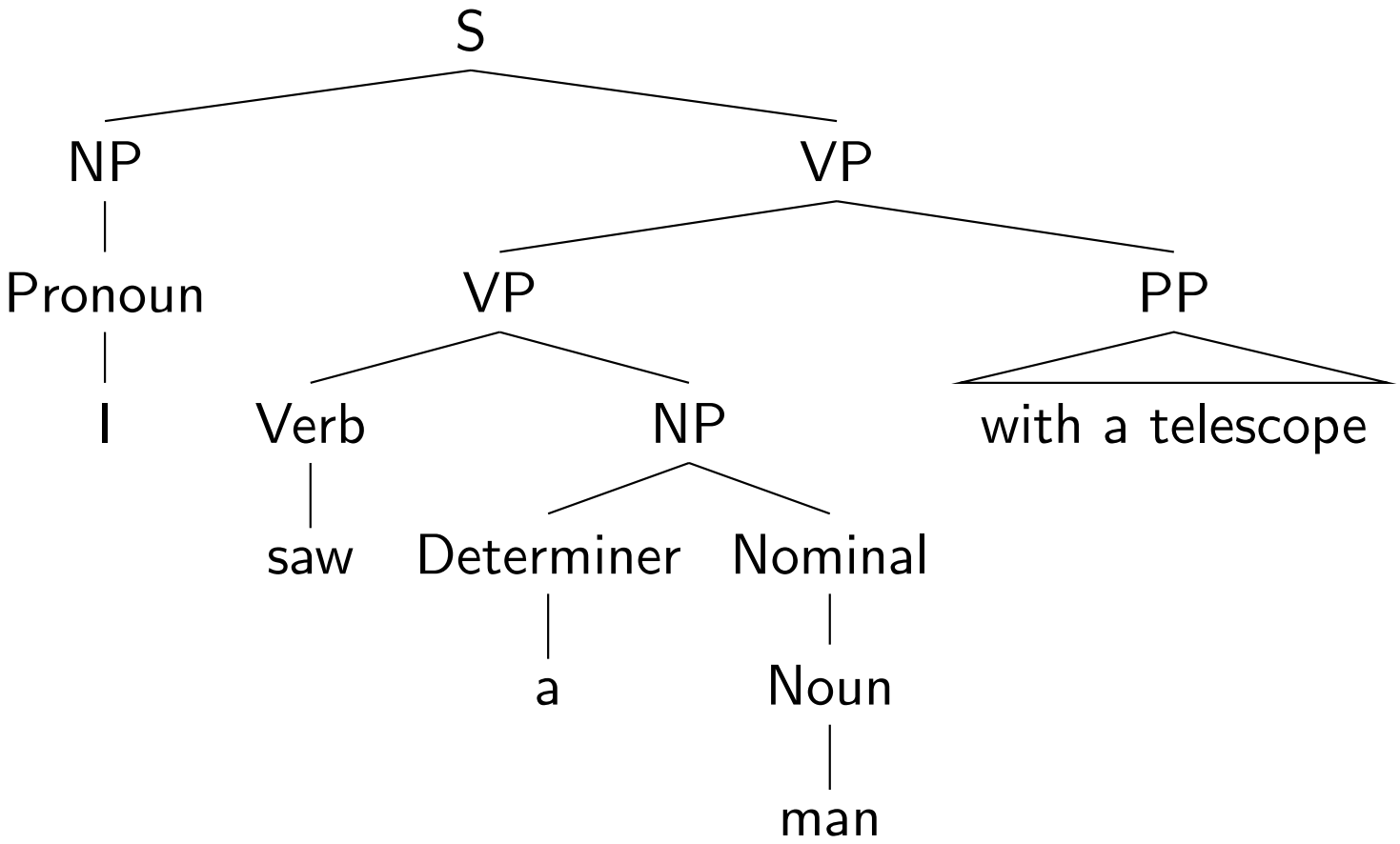
I saw a man with a telescope

Modify the following tree for the above sentence



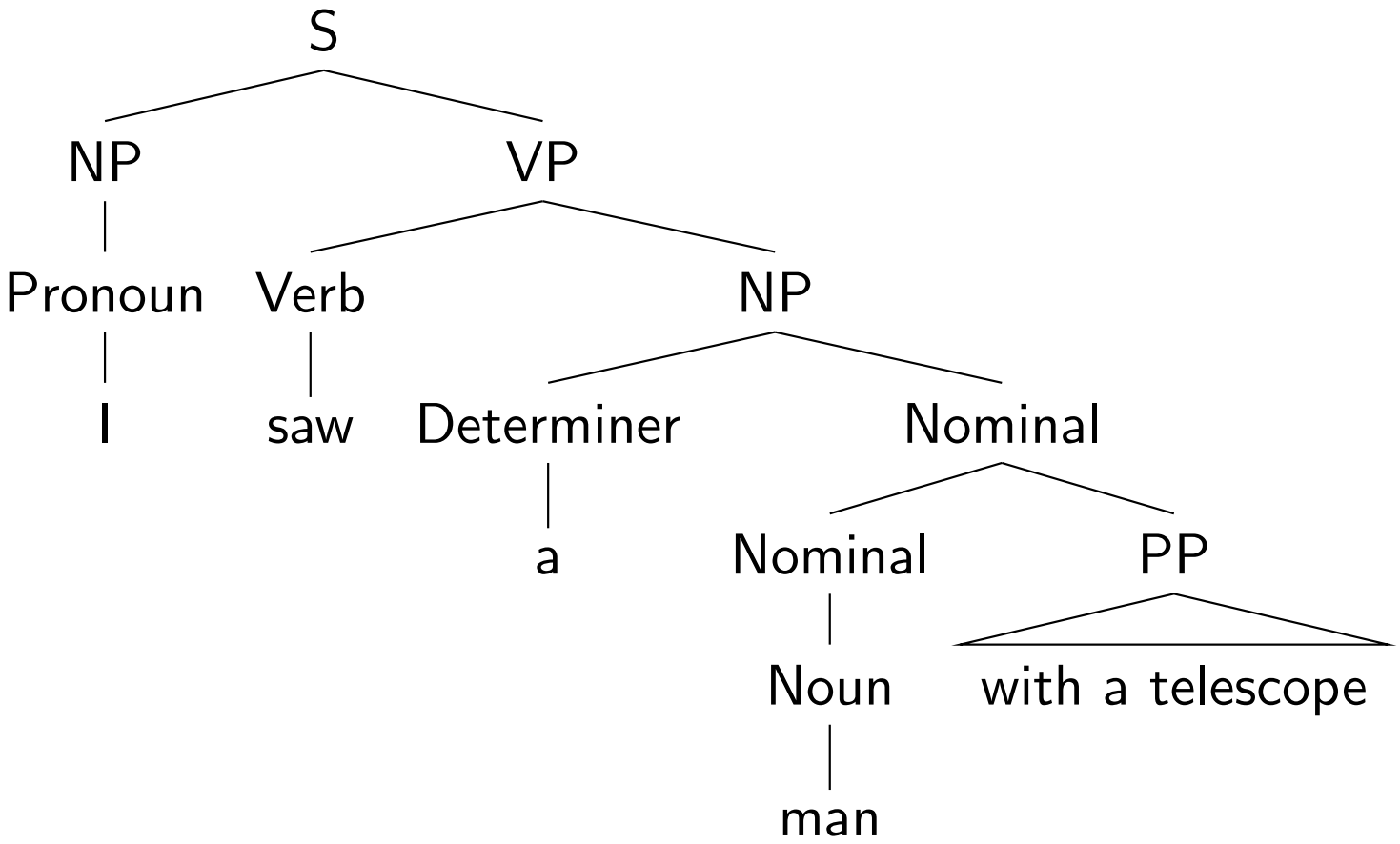
Attachment Ambiguity: 1

I saw a man with a telescope



Attachment Ambiguity: 2

I saw a man with a telescope



Simple Coordination Productions

Add these to the earlier grammar

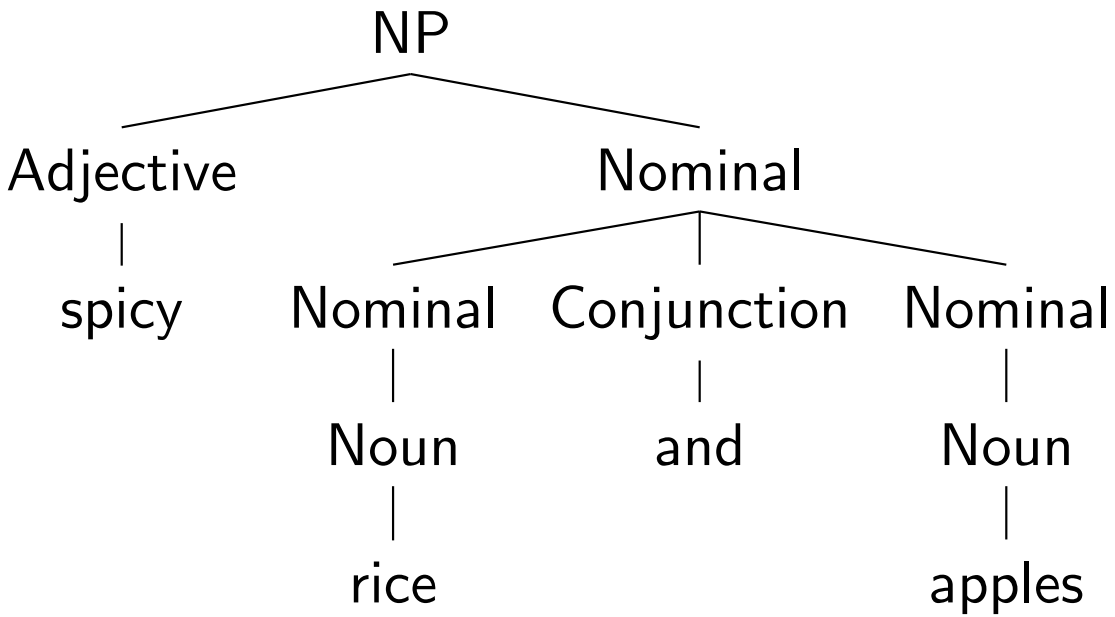
$$
\begin{aligned}
\text{NP} &\longrightarrow \text{NP Conjunction NP} \\
\text{Nominal} &\longrightarrow \text{Nominal Conjunction Nominal} \\
\text{VP} &\longrightarrow \text{VP Conjunction VP} \\
\text{PP} &\longrightarrow \text{PP Conjunction PP}
\end{aligned}
$$

Also, for adjectives include

$$\text{NP} \longrightarrow \text{Adjective Nominal}$$

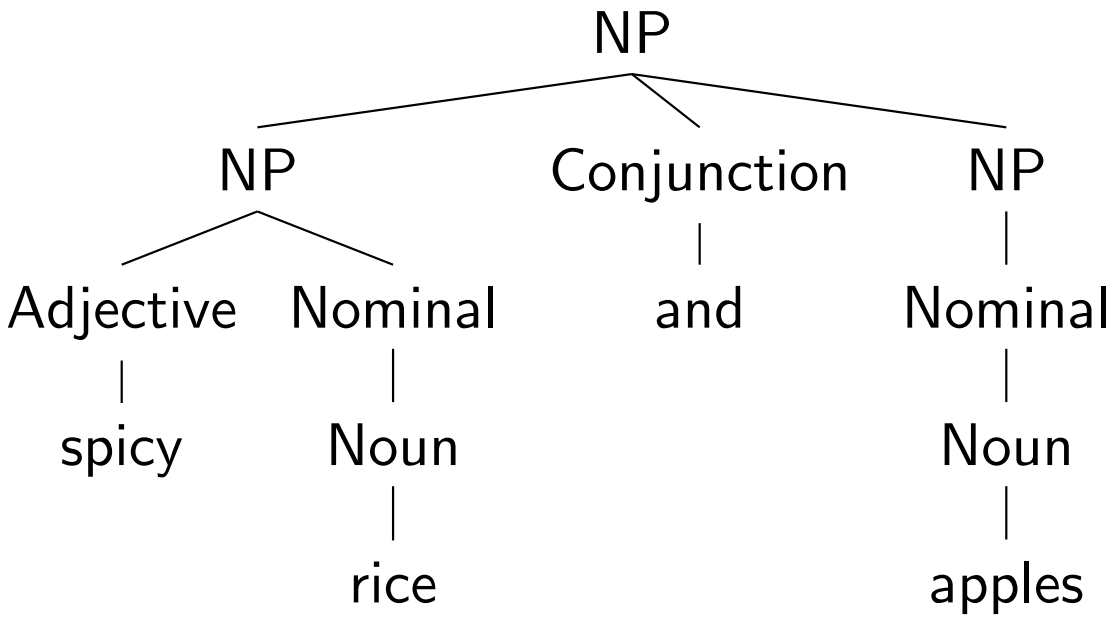
Coordination Ambiguity: 1

Spicy rice and apples



Coordination Ambiguity: 2

Spicy rice and apples



Sentences in Practice

A. A. Milne, Winnie the Pooh

Eeyore's take on writing

"This writing business. Pencils and what-not. Over-rated, if you ask me. Silly stuff. Nothing in it."

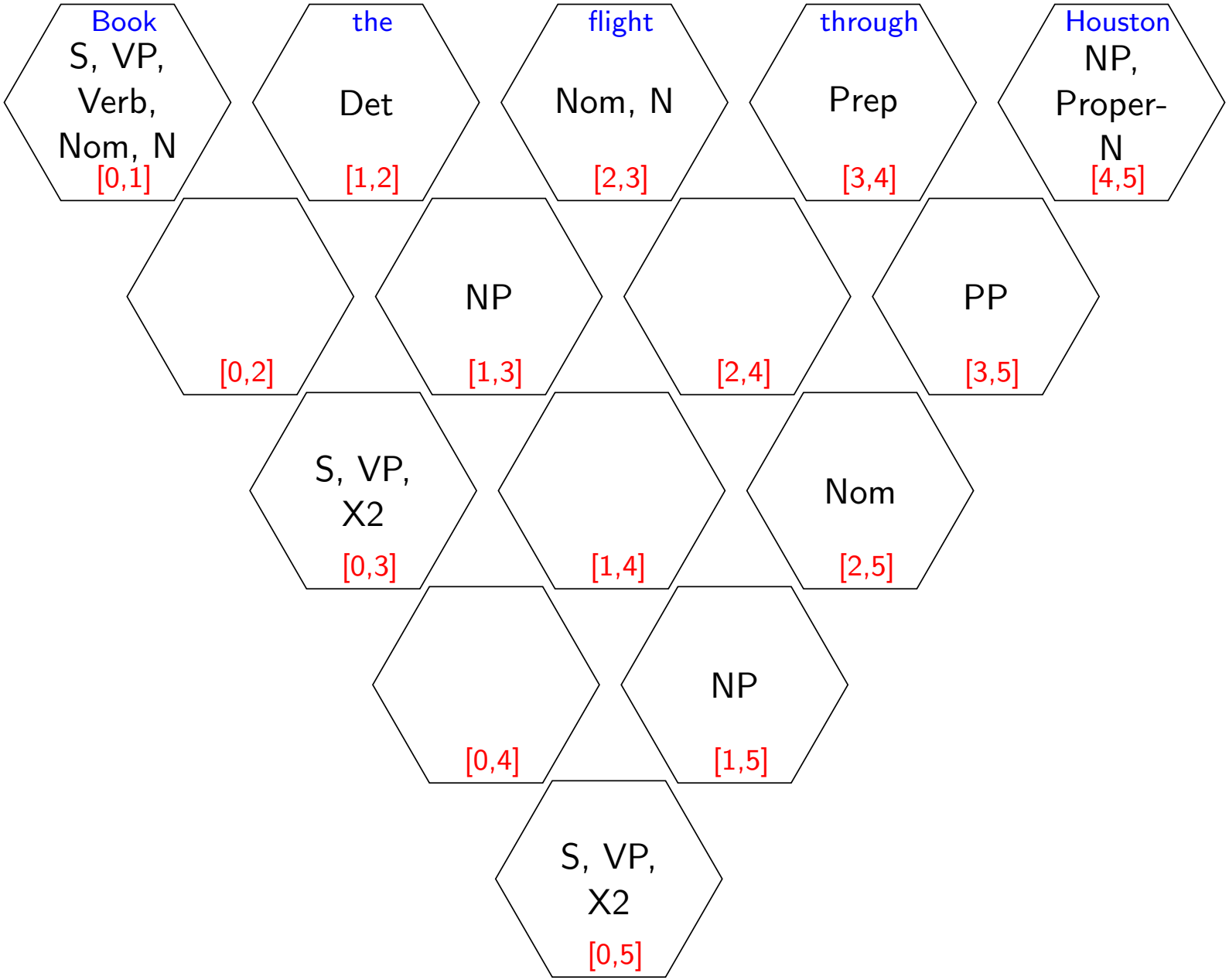
- Five sentences
- Do you identify verbs in them?
- What grammar would generate these sentences?

Parsing with a Context-Free Grammar

Cocke-Kasami-Younger (CKY) algorithm

- Apply dynamic programming
  - Build up solutions incrementally
  - Reusing them in larger solutions
- Convert to Chomsky Normal Form
- Each constituent is based on
  - A single terminal
  - Two nonterminals (constituents)
- Compute and store all possible constituents for each cell in a matrix
  - Allow duplicates to accommodate ambiguity
  - Store provenance of each value
- When we arrive at a cell the cells it relies upon are already computed
- The nonterminal in the final cell represents the constituent for the entire input (if any)
- Reconstruct parse tree from the provenance

Example of a CKY Parse



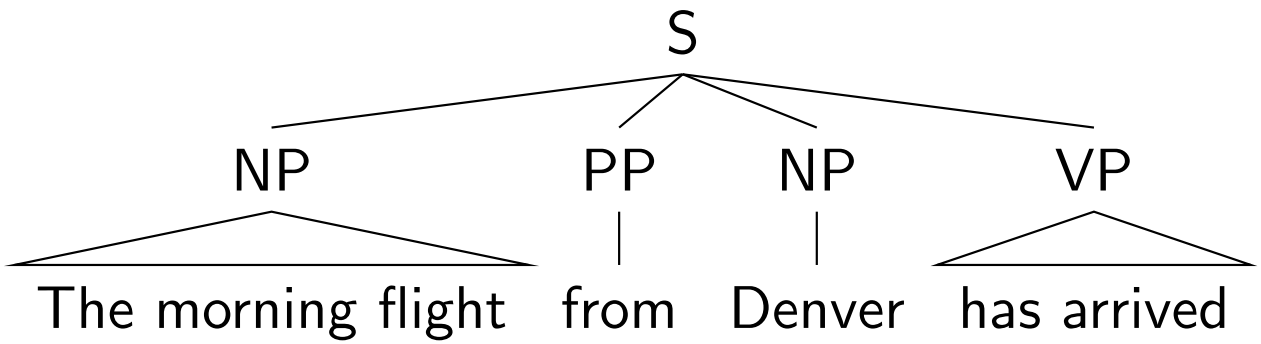
Improving CKY for Practical Use

- Generalize to arbitrary grammars (not just Chomsky Normal Form)
  - Ensures parses produced reflect grammarians' intuitions
- In statistical parsing, accommodate probabilities to
  - Select likelier parses
  - Avoid exponentially many parses

Partial or Shallow Parsing

Applicable when we don't need a complete parse to produce a valuable product

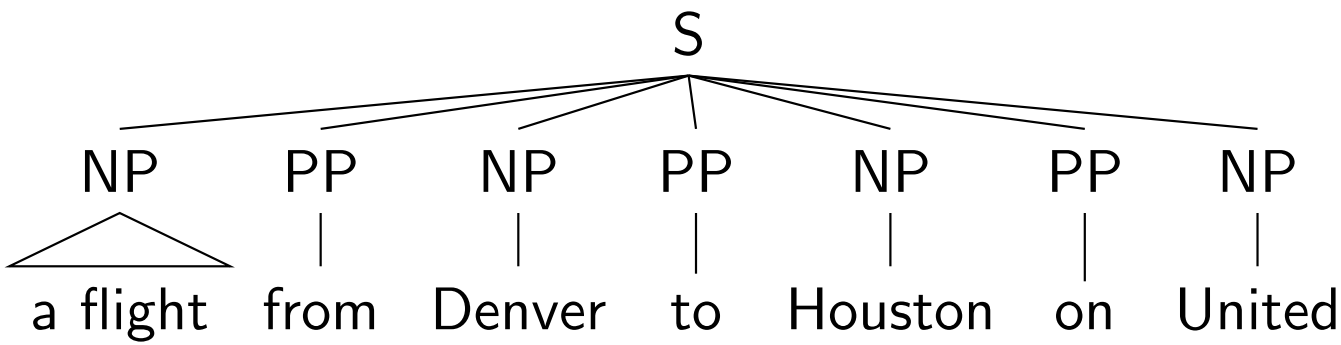
- Produce flat trees
  - Avoid decisions about nesting and ambiguity that a full parser must contend with
- *Chunking*: Identify constituents for nonoverlapping segments
- Exclude hierarchical structure
  - $[_{\text{Pro}}$ I$]$ $[_{\text{V}}$ saw$]$ $[_{\text{NP}}$ a man$]$ $[_{\text{PP}}$ with a telescope$]$



Identifying Base Phrases

Alternative to chunking

- A base phrase (some variation in definitions)
  - Doesn't (recursively) contain constituents of the same type
  - Includes the headword and any prehead modifiers (or any post-head material)
  - Excludes post-head modifiers (to avoid attachment ambiguity)
    - Can be difficult to use as a result since boundaries are less clear
    - Can yield outcomes where an NP or PP may contain nothing other than its head



Machine Learning for Chunking

An application of sequence learning

- Introduce $2n+1$ tags (given n chunk types)
  - B_k: Beginning of chunk type k
  - I_k: Inside of chunk type k
  - O: Outside of all chunk types
  - No need for end of a chunk since the beginning of the next (or end of sentence) indicates its end
- Example of IOB chunking

I	saw	a	man	with	a	telescope
B_{NP}	B_{VP}	B_{NP}	I_{NP}	B_{PP}	I_{PP}	I_{PP}
$[_{NP}$I]	$[_{VP}$saw]	$[_{NP}$a	man]	$[_{PP}$with	a	telescope]

- Training data: from existing treebanks
  - Identify head words of a constituent
  - Include head and prehead words within the constituent
  - Exclude post-head words

Evaluation Metrics for Chunking

- Correct chunk: whose tag (label) and segment are correct
- Metrics adopted from information retrieval

$$\text{Precision}, P = \frac{\text{Number of correct chunks identified}}{\text{Number of chunks identified}}$$

$$\text{Recall}, R = \frac{\text{Number of correct chunks identified}}{\text{Number of (correct) chunks existing}}$$

$$\text{F-measure}, F_\beta = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

$$\text{F}_1, F_1 = \frac{2PR}{P+R}$$

- F-measure trades off precision and recall
  - F_1 gives equal importance to precision and recall