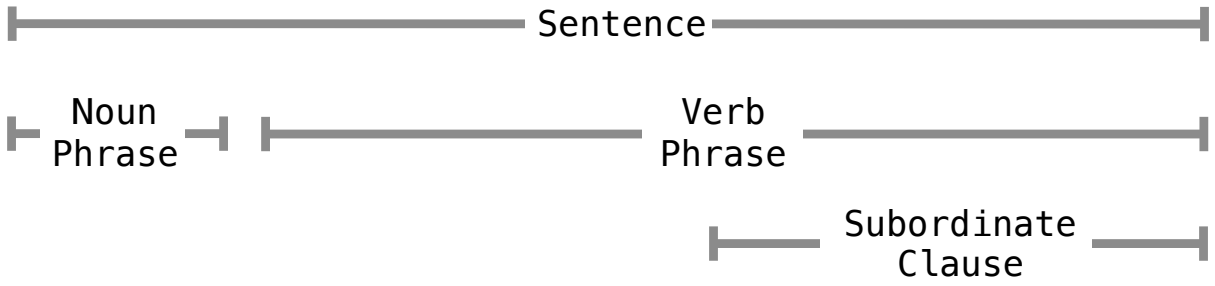
61A Lecture 37

Announcements

Ambiguity

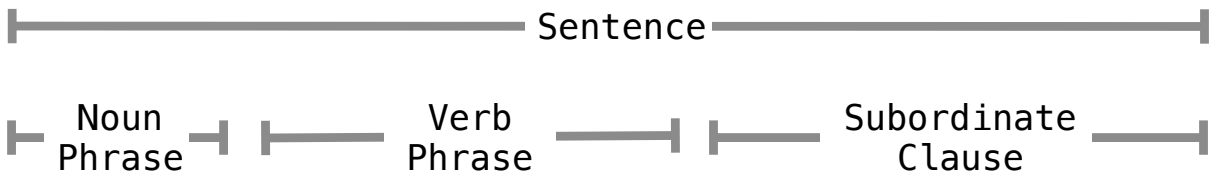
Syntactic Ambiguity in English



Programs must be written for people to read[1]

[1]Preface of **Structure and Interpretation of Computer Programs**
by Harold Abelson and Gerald Sussman with Julie Sussman

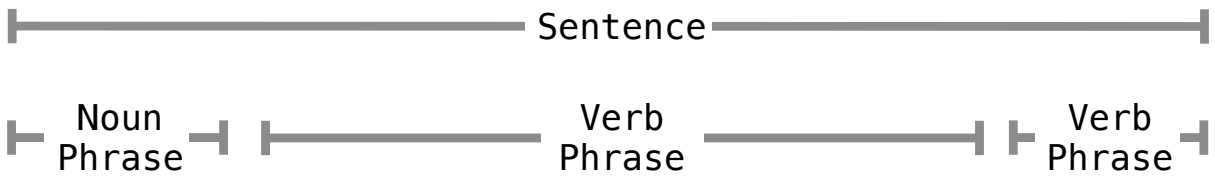
Syntactic Ambiguity in English



Programs must be written for people to read[1]

[1]Preface of **Structure and Interpretation of Computer Programs**
by Harold Abelson and Gerald Sussman with Julie Sussman

Syntactic Ambiguity in English



Programs must be written for people to read[1]

[1]Preface of **Structure and Interpretation of Computer Programs**
by Harold Abelson and Gerald Sussman with Julie Sussman

Syntactic Ambiguity in English

pro•gram (noun)
  a series of coded software instructions

pro•gram (verb)
  provide a computer with coded instructions

Programs must be written for people to read

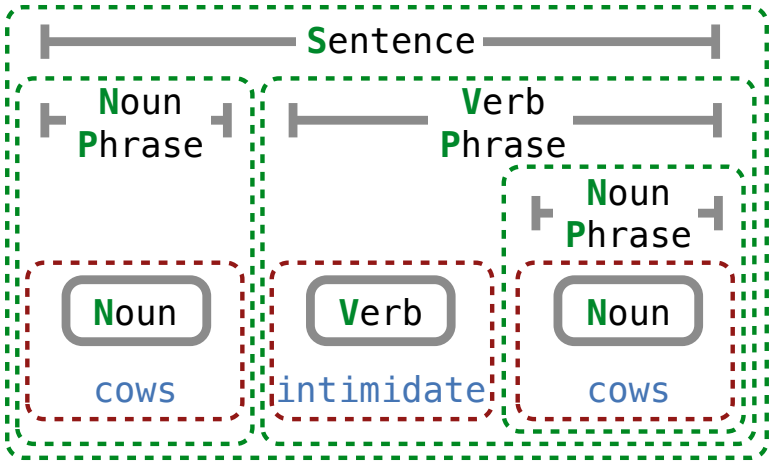
must (verb)
  be obliged to

must (noun)
  dampness or mold

Definitions from the New Oxford American Dictionary

Syntax Trees

Representing Syntactic Structure



A **Tree** represents a phrase:

- **tag** -- What kind of phrase (e.g., **S**, **NP**, **VP**)
- **branches** -- Sequence of **Tree** or **Leaf** components

A **Leaf** represents a single word:

- **tag** -- What kind of word (e.g., **N**, **V**)
- **word** -- The word

```
cows = Leaf('N', 'cows')
intimidate = Leaf('V', 'intimidate')
S, NP, VP = 'S', 'NP', 'VP'
Tree(S, [Tree(NP, [cows]),
         Tree(VP, [intimidate,
                   Tree(NP, [cows])])])
```

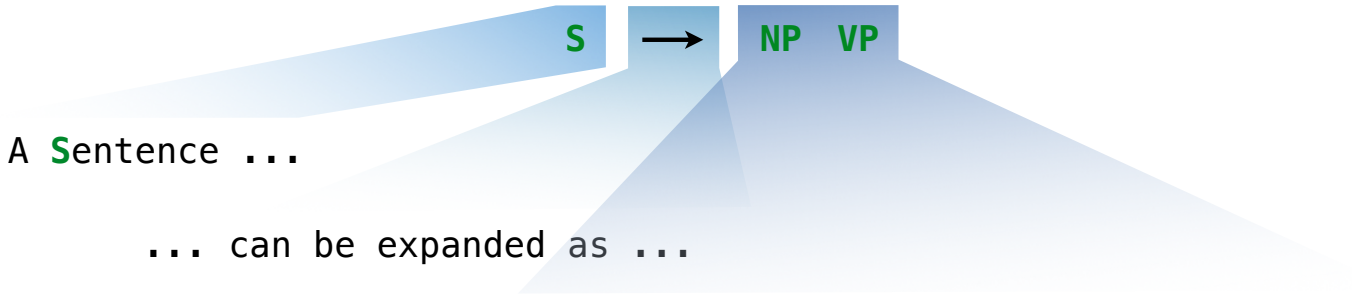
(Demo)

See 37/tree.py & print_tree.py

Grammars

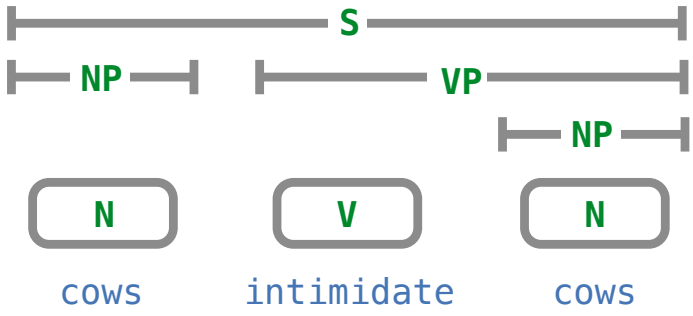
Context-Free Grammar Rules

A grammar rule describes how a tag can be expanded as a sequence of tags or words



Grammar

S → NP VP
NP → N
N → cows
VP → V NP
V → intimidate

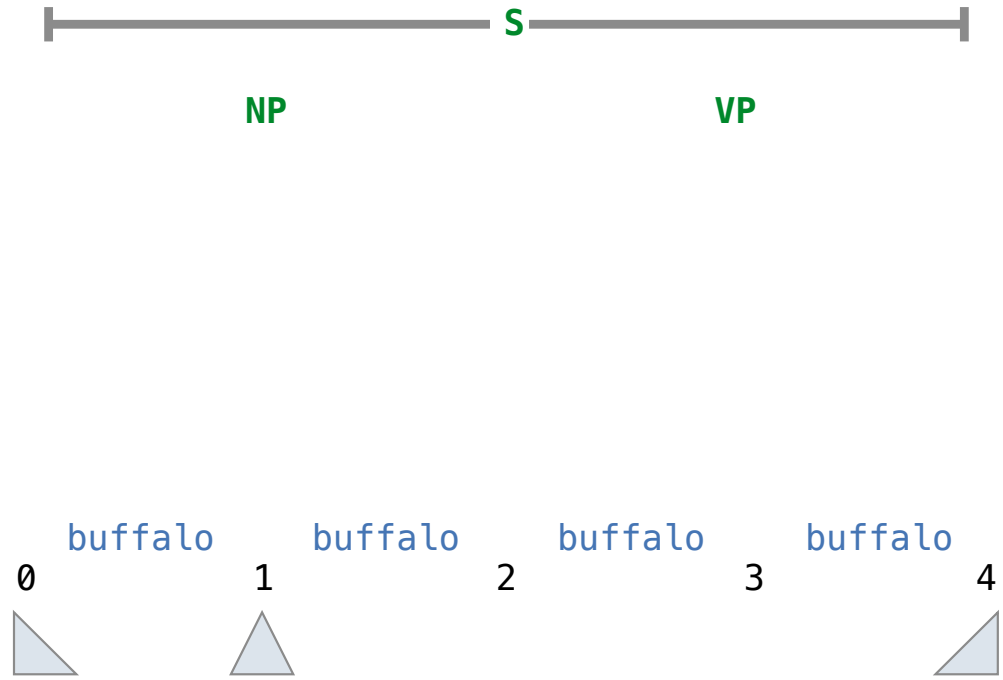


(Demo)

Parsing

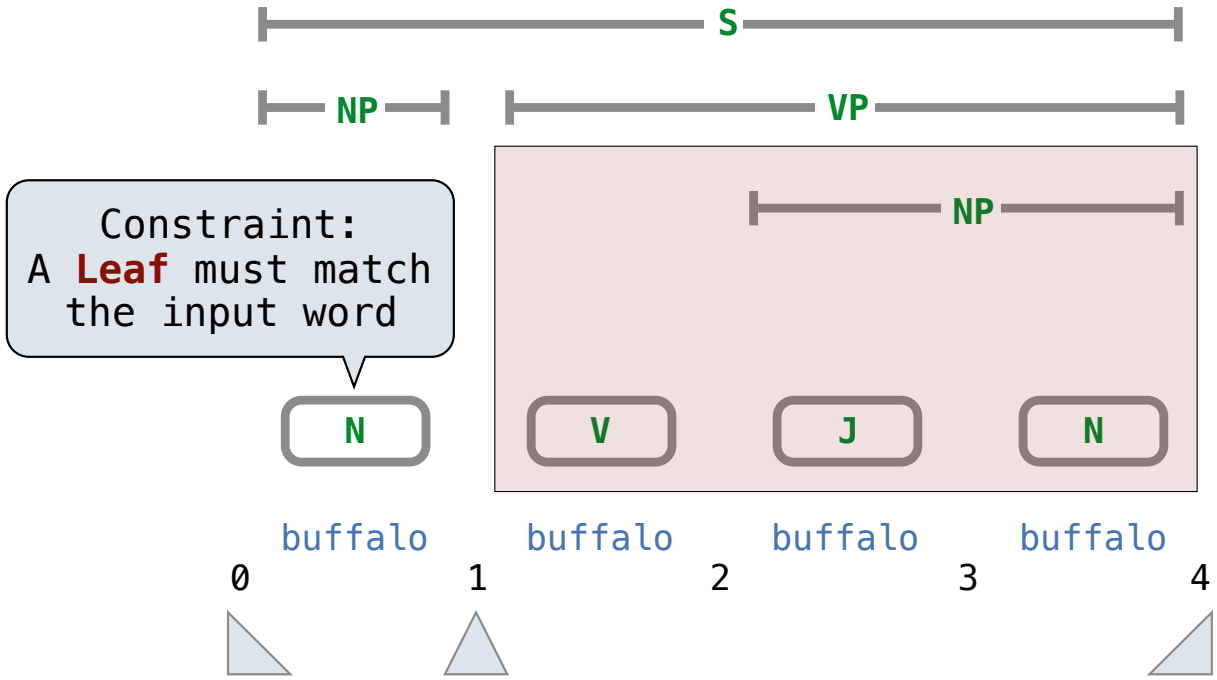
Exhaustive Parsing

Expand all tags recursively, but constrain words to match input



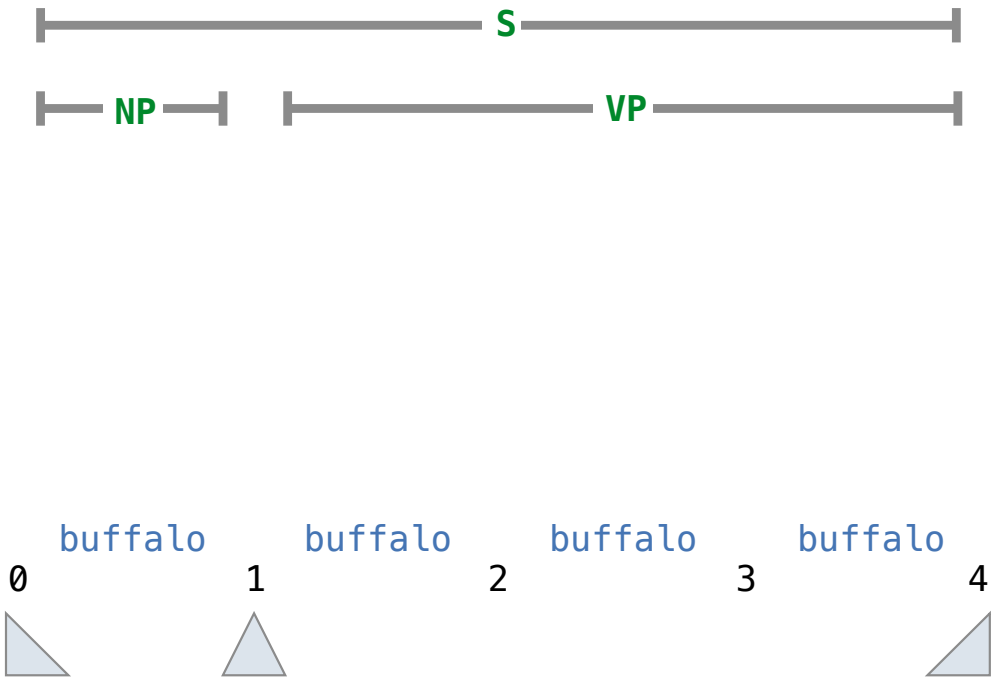
Exhaustive Parsing

Expand all tags recursively, but constrain words to match input



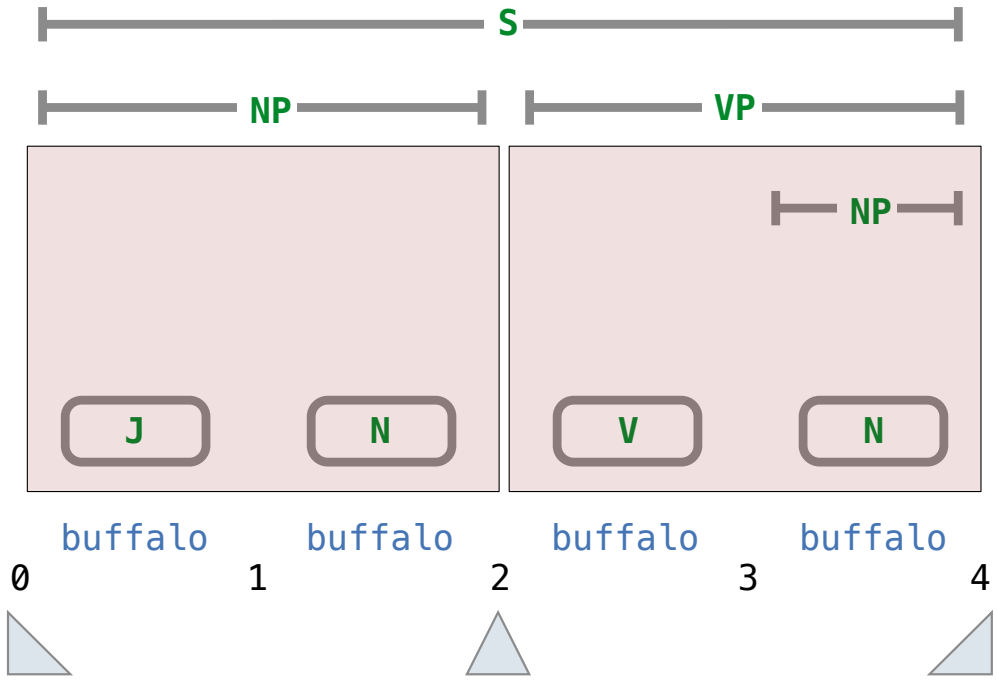
Exhaustive Parsing

Expand all tags recursively, but constrain words to match input



Exhaustive Parsing

Expand all tags recursively, but constrain words to match input



(Demo)

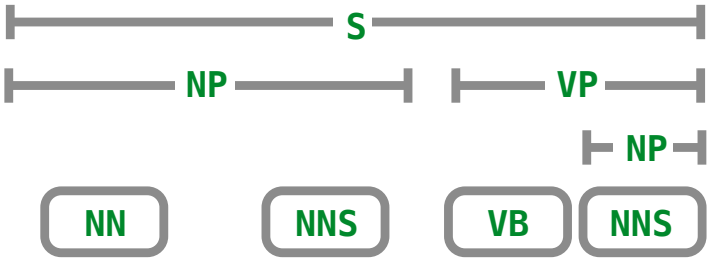
See 37/parse.py & parse_complete.py

Learning

(Demo)

Scoring a Tree Using Relative Frequencies

Not all syntactic structures are equally common

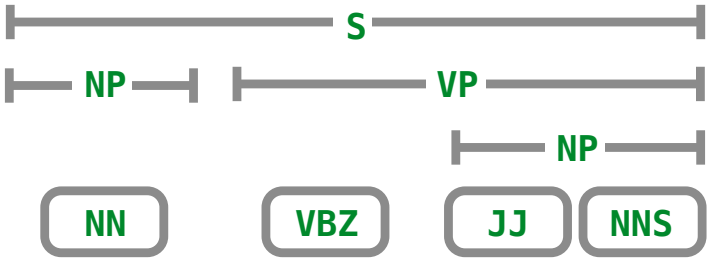


Rule frequency per 100,000 tags

S ⟶ NP VP	25372	NN ⟶	teacher	5
NP ⟶ NN NNS	1335	NNS ⟶	strikes	25
VP ⟶ VB NP	6679	VB ⟶	idle	26
NP ⟶ NNS	4282	NNS ⟶	kids	32

Scoring a Tree Using Relative Frequencies

Not all syntactic structures are equally common



Rule frequency per 100,000 tags

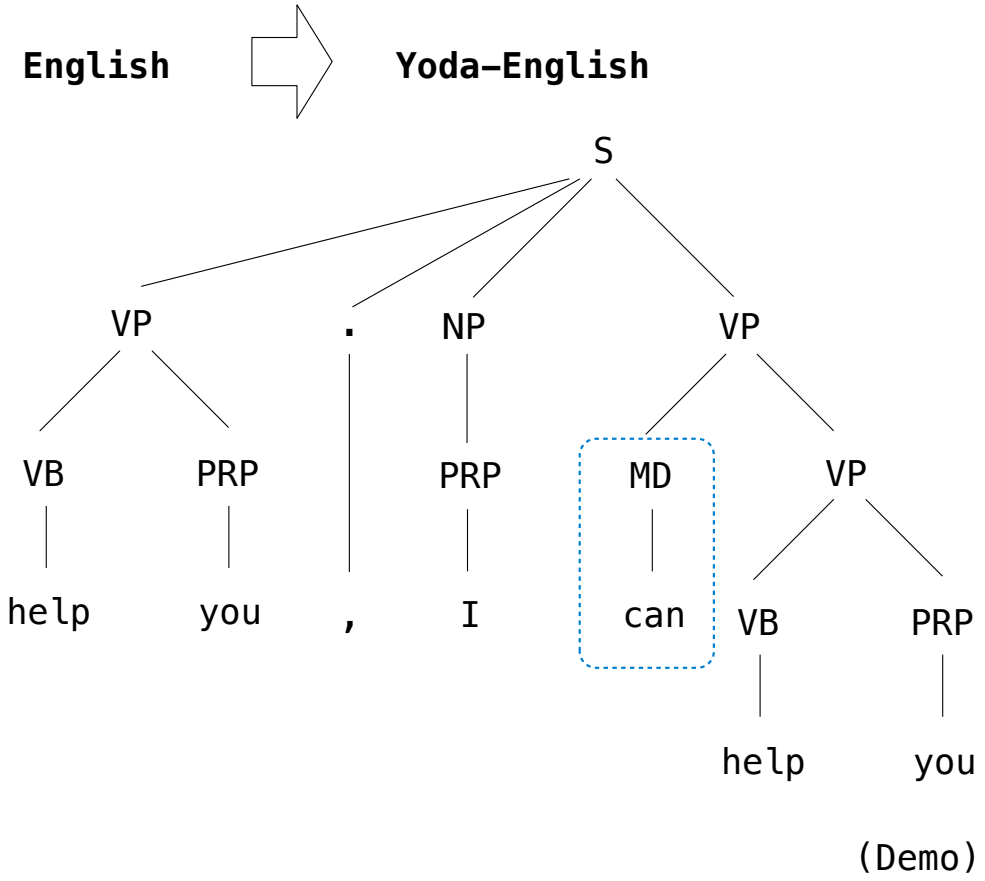
S ⟶ NP VP	25372		NN ⟶	teacher	5		
NP ⟶ NN	~~1335~~	4358	VBZ ⟶	strikes	~~25~~	19	
VP ⟶ VBZ NP	~~6679~~	3160	JJ ⟶	idle	~~26~~	18	
NP ⟶ JJ NNS	~~4282~~	2526	NNS ⟶	kids	32		

(Demo)

See 37/max_parse.py

Translation

Syntactic Reordering



(Demo)

See 37/yoda.py