Syntax: Introduction

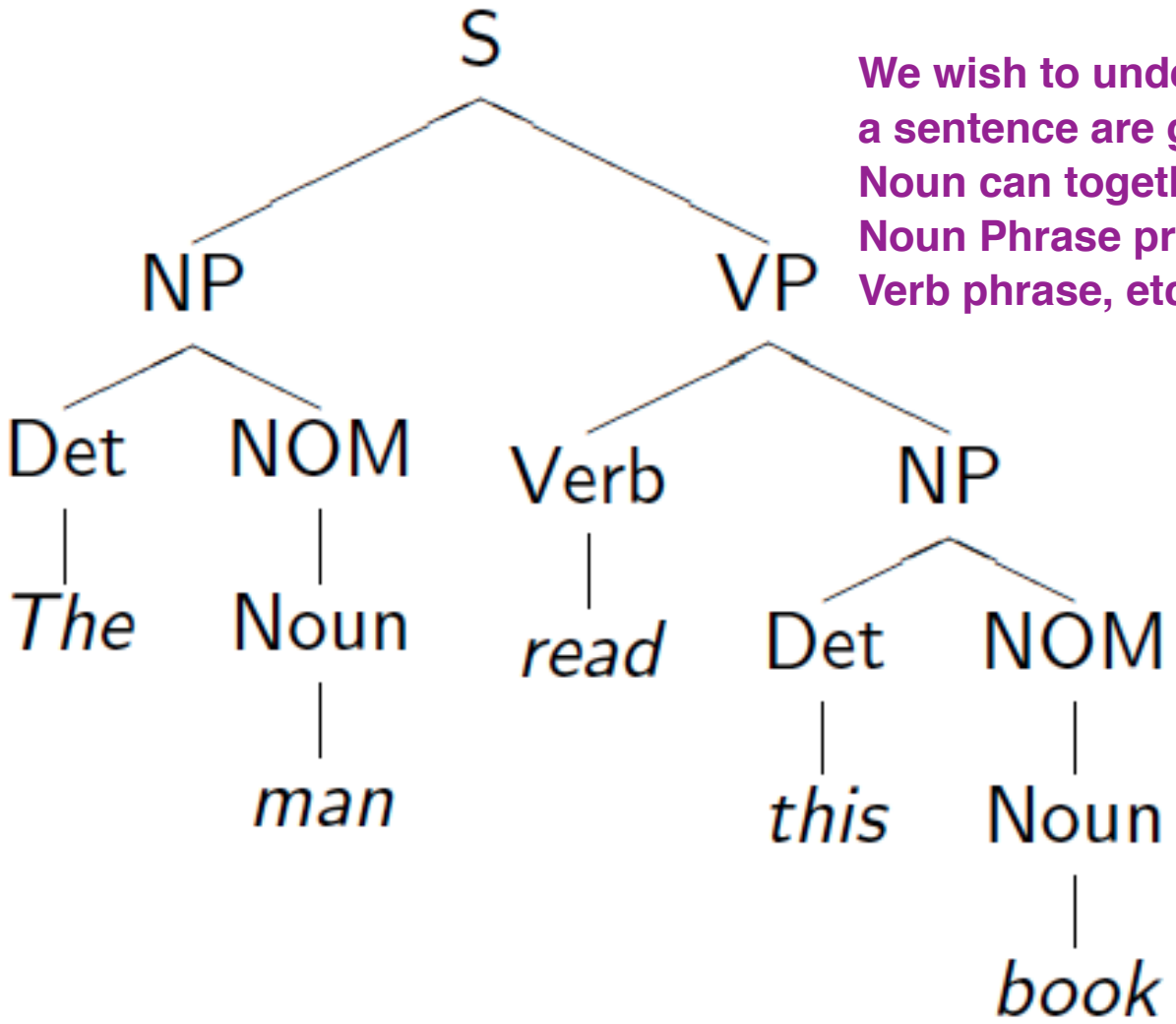
Till now, we have discussed:
Language modeling — understanding ordering of words
POS tagging — understanding roles of words

Now we will discuss more complex notions, such as grammatical relations among words

What is Syntax?

- Refers to the way words are arranged together, and the relationship between then.

Syntax Tree: Example



Sentence: The man read this book

We wish to understand how different words in a sentence are grouped, e.g., a Det and a Noun can together give a Noun Phrase, a Noun Phrase preceded by a verb can give a Verb phrase, etc.

Our final goal:
How to construct a syntax tree for a given sentence?

Defining the notions: Constituency

Constituent

A group of words acts as a single unit - phrases, clauses etc.

Defining the notions: Constituency

Constituent

A group of words acts as a single unit - phrases, clauses etc.

Part of Speech - "Substitution Test"

The {sad, intelligent, green, fat, ...} one is in the corner.

Different words having the same POS (here, adjective) can be used as a substitute for each other.

Defining the notions: Constituency

Constituent

A group of words acts as a single unit - phrases, clauses etc.

Part of Speech - "Substitution Test"

The {sad, intelligent, green, fat, ...} one is in the corner.

Constituency: Noun Phrase

- *Kermit the frog*
- *they*
- *December twenty-sixth*
- *the reason he is running for president*

Similarly, different word-sequences (phrases) can be used as a substitute for each other. E.g., noun phrases

Constituent Phrases

How are constituent phrases usually named?

Usually named based on the word that heads the constituent:

the man from Amherst	is a Noun Phrase (NP) because the head man is a noun
extremely clever	is an Adjective Phrase (AP) because the head clever is an adjective
down the river	is a Prepositional Phrase (PP) because the head down is a preposition
killed the rabbit	is a Verb Phrase (VP) because the head killed is a verb

Constituent Phrases

Usually named based on the word that heads the constituent:

the man from Amherst	is a Noun Phrase (NP) because the head man is a noun
extremely clever	is an Adjective Phrase (AP) because the head clever is an adjective
down the river	is a Prepositional Phrase (PP) because the head down is a preposition
killed the rabbit	is a Verb Phrase (VP) because the head killed is a verb

Words can also act as phrases

Joe grew potatoes

Joe and *potatoes* are both nouns and noun phrases

Constituent Phrases

Usually named based on the word that heads the constituent:

the man from Amherst	is a Noun Phrase (NP) because the head man is a noun
extremely clever	is an Adjective Phrase (AP) because the head clever is an adjective
down the river	is a Prepositional Phrase (PP) because the head down is a preposition
killed the rabbit	is a Verb Phrase (VP) because the head killed is a verb

Words can also act as phrases

Joe grew potatoes

Joe and *potatoes* are both nouns and noun phrases

Compare with: *The man from Amherst grew beautiful russet potatoes*.

Joe appears in a place that a larger noun phrase could have been.

Evidence that constituency exists

They appear in similar environments

Evidence that constituency exists

They appear in similar environments

Kermit the frog comes on stage
They come to Massachusetts every summer
December twenty-sixth comes after Christmas
The reason he is running for president comes out only now.

But not each individual word in the consituent

**The comes our...* **is comes out...* **for comes out...*

Evidence that constituency exists

They appear in similar environments

Kermit the frog comes on stage
They come to Massachusetts every summer
December twenty-sixth comes after Christmas
The reason he is running for president comes out only now.
But not each individual word in the consituent
**The comes our...* **is comes out...* **for comes out...*

Can be placed in a number of different locations

Evidence that constituency exists

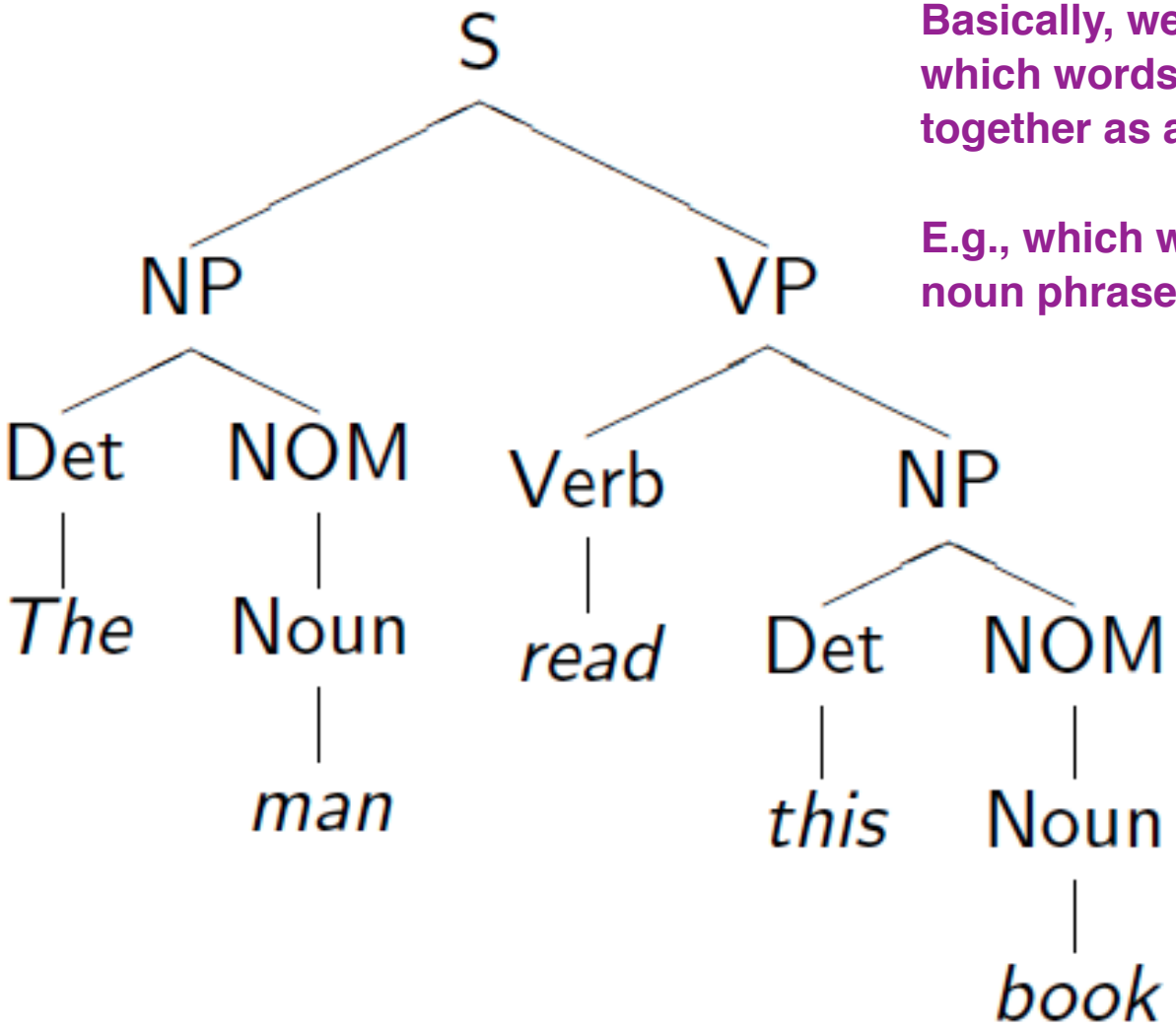
They appear in similar environments

Kermit the frog comes on stage
They come to Massachusetts every summer
December twenty-sixth comes after Christmas
The reason he is running for president comes out only now.
But not each individual word in the consituent
**The comes our...* **is comes out...* **for comes out...*

Can be placed in a number of different locations

Consituent = Prepositional phrase: *On December twenty-sixth*
On December twenty-sixth I'd like to fly to Florida.
I'd like to fly on December twenty-sixth to Florida.
I'd like to fly to Florida on December twenty-sixth.
But not split apart
**On December I'd like to fly twenty-sixth to Florida.*
**On I'd like to fly December twenty-sixth to Florida.*

Modeling Constituency: what tool do we need?



Basically, we want tools to understand which words (in a given text) come together as a unit.

E.g., which words together make up a noun phrase?

Modeling Constituency

Context-free grammar

The most common way of modeling constituency

Modeling Constituency

Context-free grammar

The most common way of modeling constituency

Consists of production Rules

These rules express the ways in which the symbols of the language can be grouped and ordered together

Modeling Constituency

Context-free grammar

The most common way of modeling constituency

Consists of production Rules

These rules express the ways in which the symbols of the language can be grouped and ordered together

Example

Noun phrase can be composed of either a ProperNoun or a determiner (Det) followed by a Nominal; a Nominal can be more than one nouns

Modeling Constituency

Context-free grammar

The most common way of modeling constituency

Consists of production Rules

These rules express the ways in which the symbols of the language can be grouped and ordered together

Example

Noun phrase can be composed of either a ProperNoun or a determiner (Det) followed by a Nominal; a Nominal can be more than one nouns

$NP \rightarrow$ Det Nominal

$NP \rightarrow$ ProperNoun

$Nominal \rightarrow$ Noun | Noun Nominal

CFG for Languages

CFG: $G = (T, N, S, R)$

- T: set of terminals
- N: set of non-terminals
  - For NLP, we distinguish out a set $P \subset N$ of pre-terminals, which always rewrite as terminals
- S : start symbol
- R: Rules/productions of the form $X \rightarrow \gamma$, $X \in N$ and $\gamma \in (T \cup N)*$

CFG for Languages

CFG: $G = (T, N, S, R)$

- T: set of terminals
- N: set of non-terminals
  - For NLP, we distinguish out a set $P \subset N$ of pre-terminals, which always rewrite as terminals
- S : start symbol
- R: Rules/productions of the form $X \rightarrow \gamma$, $X \in N$ and $\gamma \in (T \cup N)*$

Terminals and pre-terminals

Terminals mainly correspond to words in the language while pre-terminals mainly correspond to POS categories

CFG for Languages

Example

$NP \rightarrow$ Det Nominal

$NP \rightarrow$ ProperNoun

$Nominal \rightarrow$ Noun | Noun Nominal

"Noun", "ProperNoun", "Det" are pre-terminals.

But we cannot generate any string with these rules, since there is no terminal.

CFG for Languages

Example

$NP \rightarrow$ Det Nominal

$NP \rightarrow$ ProperNoun

$Nominal \rightarrow$ Noun | Noun Nominal

Now, these can be combined with other rules, that express facts about a lexicon.

CFG for Languages

Example

$NP \rightarrow$ Det Nominal

$NP \rightarrow$ ProperNoun

$Nominal \rightarrow$ Noun | Noun Nominal

Now, these can be combined with other rules, that express facts about a lexicon.

$Det \rightarrow$ a

$Det \rightarrow$ the

$Noun \rightarrow$ flight

Terminals: "a", "the", "flight"
Pre-terminals: "Det", "Noun", "ProperNoun"
Non-terminals: "NP", "Nominal"

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight
Generating ‘a flight’:

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight
Generating 'a flight':
$NP \rightarrow$ Det Nominal

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight
Generating 'a flight':
$NP \rightarrow$ Det Nominal
$\rightarrow$ Det Noun

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight
Generating 'a flight':
$NP \rightarrow$ Det Nominal
$\rightarrow$ Det Noun $\rightarrow$ a Noun

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight
Generating 'a flight':
$NP \rightarrow$ Det Nominal
$\rightarrow$ Det Noun $\rightarrow$ a Noun $\rightarrow$ a flight

CFG as a generator

$NP \rightarrow$ Det Nominal
$NP \rightarrow$ ProperNoun
$Nominal \rightarrow$ Noun | Noun Nominal
$Det \rightarrow$ a
$Det \rightarrow$ the
$Noun \rightarrow$ flight
Generating 'a flight':
$NP \rightarrow$ Det Nominal
$\rightarrow$ Det Noun $\rightarrow$ a Noun $\rightarrow$ a flight

- Thus a CFG can be used to randomly generate a series of strings

CFGs and Recursion

A CFG can contain recursive rules.
E.g., the rule for a Prepositional Phrase (PP)

Recursive Definition

- PP → Prep NP
- NP → Noun PP

CFGs and Recursion

Recursive Definition

- PP → Prep NP
- NP → Noun PP

start of a NP start of a NP

Example Sentence

[$_S$The mailman ate his [$_{NP}$ lunch [$_{PP}$ with his friend [$_{PP}$ from the cleaning staff [$_{PP}$ of the building [$_{PP}$ at the intersection [$_{PP}$ on the north end [$_{PP}$ of town]]]]]]].

start of a NP start of a NP start of a NP

CFGs and Grammaticality

A CFG defines a formal language = set of all sentences (string of words) that can be derived by the grammar

- Sentences in this set are said to be **grammatical**
- Sentences outside this set are said to be **ungrammatical**