Part-of-Speech Tagging and Constituency Grammars

Natalie Parde
UIC CS 421

What is part-of-speech (POS) tagging?

The process of automatically assigning grammatical word classes to individual tokens in text.

verb determiner

Give me a **break**!

pronoun noun

verb noun

Did the window **break**?

determiner verb

POS Tagging

What are parts of speech?

- Traditional (broad) categories:
  - noun
  - verb
  - adjective
  - adverb
  - preposition
  - article
  - interjection
  - pronoun
  - conjunction
- Sometimes also referred to as **lexical categories**, **word classes**, **morphological classes**, or **lexical tags**

Parts of Speech

Noun

- People, places, or things
- Doctor, mountain, cellphone….

Verb

- Actions or states
- Eat, sleep, be….

Adjective

- Descriptive attributes
- Purple, triangular, windy….

Adverb

- Modifies other words by answering *how*, *in what way*, *when*, *where*, and *to what extent* questions
- Gently, quite, quickly….

Parts of Speech

Pronoun

- Refers to nouns mentioned elsewhere
- he, she, you….

Preposition

- Describes relationship between noun/pronoun and other word in clause
- on, above, to….

Article

- Indicates specificity
- a, an, the….

Interjection

- Exclamations
- oh, yikes, ah….

Conjunction

- Coordinates words in the same clause or connects multiple clauses/sentences
- and, but, if….

Why is POS tagging useful?

- First step of many downstream NLP tasks!
  - Speech synthesis
  - Constituency parsing
  - Dependency parsing
  - Information extraction
  - Machine translation

Natalie Parde - UIC CS 421

Open and Closed Classes

Each POS type falls into one of two larger classes:

- Open
- Closed

Open class:

- New members can be created at any time
- In English:
  - Nouns, verbs, adjectives, and adverbs
- Many (but not all!) languages have these four classes

Closed class:

- A small, fixed membership …new members cannot be created spontaneously
- Usually function words
- In English:
  - Prepositions and auxiliaries (may, can, been, etc.)

Open and Closed Classes

- Broader POS classes often have smaller subclasses
  - Noun:
    - Proper (Illinois)
    - Common (state)
  - Verb:
    - Main (tweet)
    - Modal (had)
- Some subclasses of a part of speech might be open, while others are closed

Open Class

Nouns

Proper
- *IBM*
- *Italy*

Common
- *cat / cats*
- *snow*

Verbs

Main
- *see*
- *registered*

Modal
- *can*
- *had*

Adjectives
old older oldest

Adverbs
slowly

… more

Closed Class

Determiners
the some

Conjunctions
and or

Pronouns
he its

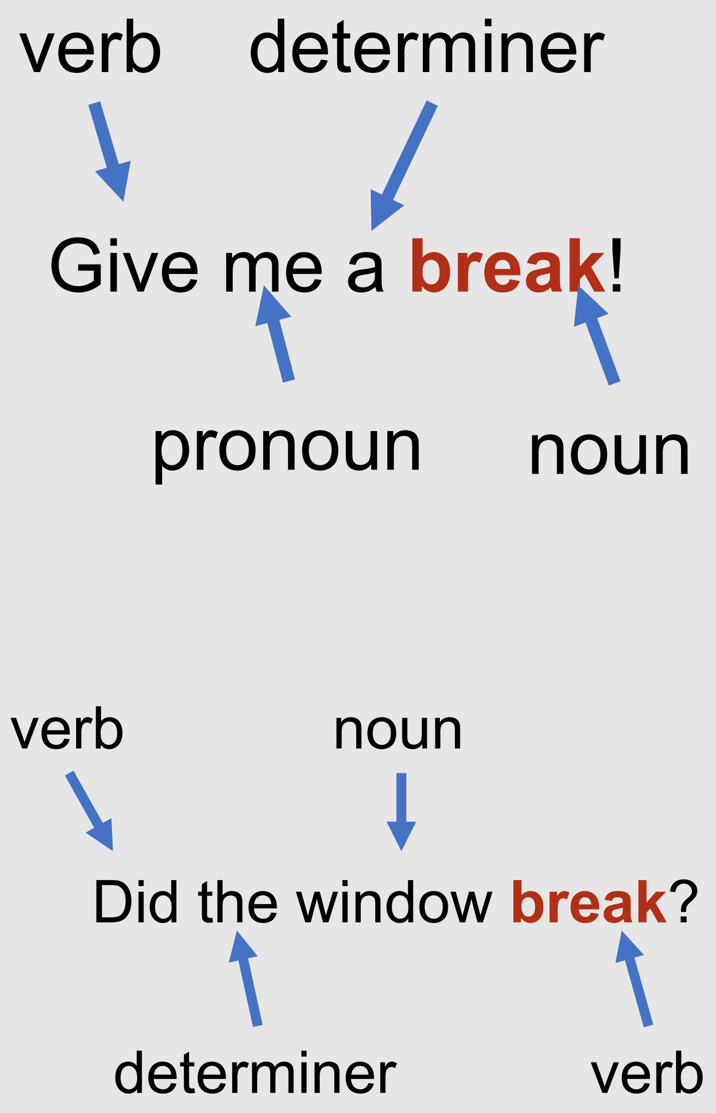
Prepositions
to with

Interjections
Ow Eh

… more

POS Tagging

- Can be very challenging!
- Words often have more than one valid part of speech tag
  - Today's faculty meeting went really **well**! = adverb
  - Do you think the undergrads are **well**? = adjective
  - **Well**, did you see the latest response to your email? = interjection
  - Jurafsky and Martin's book is a **well** of information. = noun
  - Laughter began to **well** up inside her at, as always, a highly inconvenient time. = verb



POS Tagging

- Goal: Determine the *best* POS tag for a particular instance of a word.

POS Tagsets

In order to determine which POS tag to assign to a word, we first need to decide which **tagset** we will use

Tagset: A finite set of POS tags, where each tag defines a distinct grammatical role

Can range from very coarse to very fine

Penn Treebank Tagset

- **Most common POS tagset**
- 36 POS tags + 12 other tags (punctuation and currency)
- Used when developing the Penn Treebank, a **corpus** created at the University of Pennsylvania containing more than 4.5 million words of American English
- Link to documentation: https://catalog.ldc.upenn.edu/docs/LDC95T7/cl93.html

Penn Treebank Tagset

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

What do some of these distinctions mean?

cities

Chicago

Chicagos

city

CC	Coordinating Conjunction	**NNS**	**Noun, plural**	TO	to
CD	Cardinal Number	**NNP**	**Proper noun, singular**	UH	Interjection
DT	Determiner	**NNPS**	**Proper noun, plural**	VB	Verb, base form
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	**Noun, singular or mass**	SYM	Symbol	WRB	Wh-adverb

What do some of these distinctions mean?

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

eat

ate

eating

eaten

eat

eats

should

What do some of these distinctions mean?

CC	Coordinating Conjunction	NNS	Noun, plural	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	PRP	Personal pronoun	VBN	Verb, past participle
JJ	**Adjective**	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	**Adjective, comparative**	RB	Adverb	VBZ	Verb, 3rd person singular present
JJS	**Adjective, superlative**	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass	SYM	Symbol	WRB	Wh-adverb

weird

weirder

weirdest

What do some of these distinctions mean?

CC	Coordinating Conjunction	NNS	Noun, plural	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	calmly	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	**Adverb**	VBZ	Verb, 3rd person singular present
JJS	Adjective, calmer ve	**RBR**	**Adverb, comparative**	WDT	Wh-determiner
LS	List item marker	**RBS**	**Adverb, superlative**	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singul calmest	SYM	Symbol	WRB	Wh-adverb

As a general (but not perfect!) rule….

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

Closed Class

As a general (but not perfect!) rule….

Open Class

CC	Coordinating Conjunction	**NNS**	**Noun, plural**	TO	to
CD	Cardinal Number	**NNP**	**Proper noun, singular**	UH	Interjection
DT	Determiner	**NNPS**	**Proper noun, plural**	**VB**	**Verb, base form**
EX	Existential *there*	PDT	Predeterminer	**VBD**	**Verb, past tense**
FW	**Foreign word**	POS	Possessive	**VBG**	**Verb, gerund or present participle**
IN	Preposition or subordinating conjunction	PRP	Personal pronoun	**VBN**	**Verb, past participle**
JJ	**Adjective**	PRP$	Possessive pronoun	**VBP**	**Verb, non-3rd person singular present**
JJR	**Adjective, comparative**	**RB**	**Adverb**	**VBZ**	**Verb, 3rd person singular present**
JJS	**Adjective, superlative**	**RBR**	**Adverb, comparative**	WDT	Wh-determiner
LS	List item marker	**RBS**	**Adverb, superlative**	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	**Noun, singular or mass**	SYM	Symbol	WRB	Wh-adverb

Other Popular POS Tagsets

Brown Corpus

- ~1 million words of American English text
- 82 (!) POS tags

C5 Tagset

- 61 POS tags

C7 Tagset

- 146 (!!) POS tags

So …how can we assign POS tags?

Time	flies	like	an	arrow;	fruit	flies	like	a	banana

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN									

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ								

CC	Coordinating Conjunction	**NNS**	Noun, plural ?	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present ?
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN							

CC	Coordinating Conjunction	NNS	Noun, plural ¿	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ¿ ¿
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ¿	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ¿
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ¿	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT						

CC	Coordinating Conjunction	NNS	Noun, plural ?	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ?
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN				

CC	Coordinating Conjunction	**NNS**	Noun, plural ?	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form ? ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present ?
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS			

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ??
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS	VBZ		

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ??	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ???
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS	VBZ	DT	

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner :)	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ??	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ???
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

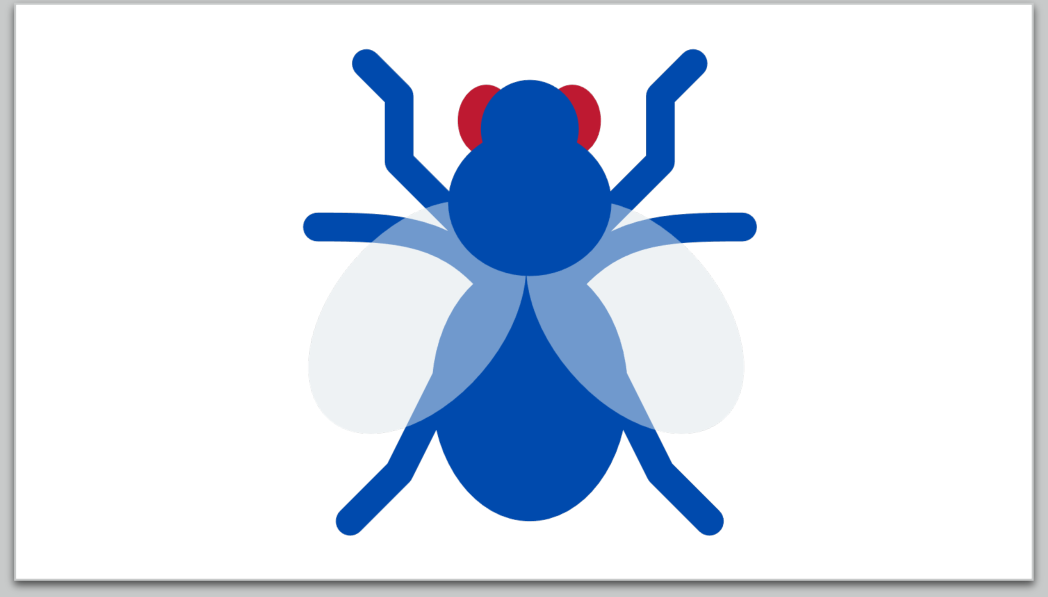
So …how can we assign POS tags?

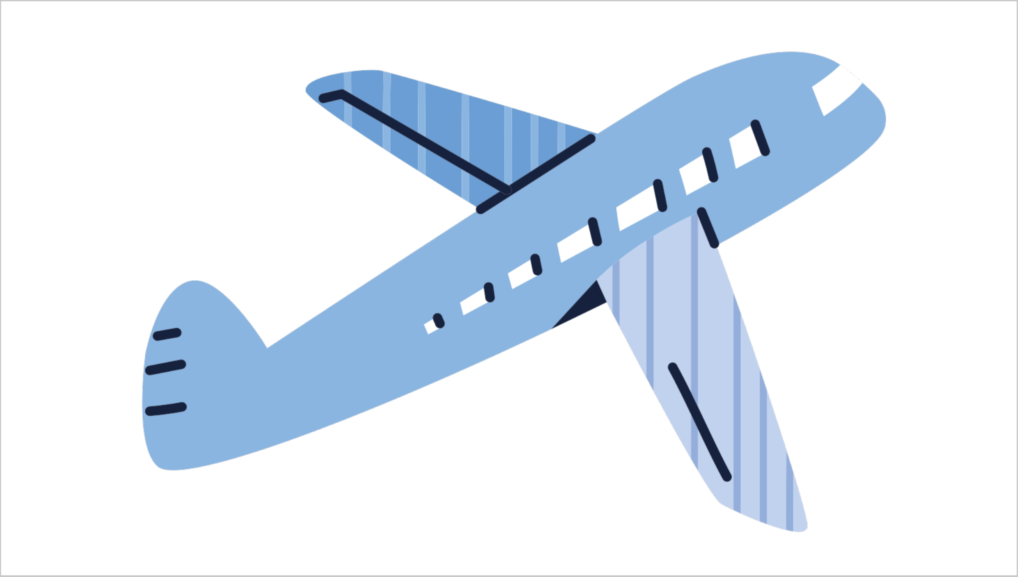
Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS	VBZ	DT	NN

CC	Coordinating Conjunction	NNS	Noun, plural ¿¿	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner :)	NNPS	Proper noun, plural	VB	Verb, base form ¿¿
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ¿¿	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ¿¿¿
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ¿	SYM	Symbol	WRB	Wh-adverb

Ambiguity is a big issue for POS taggers!

- Many words have multiple senses
    - **time** = noun, verb
    - **flies** = noun, verb
    - **like** = verb, preposition





Just how ambiguous is natural language?

- Brown Corpus: Approximately 11% of word types have multiple valid part of speech labels
- These tend to be very common words!
  - We think **that** the faculty meeting will only last two more hours. = IN
  - Was **that** the 32nd Piazza post today? = DT
  - You can't eat **that** many donuts every time the clock strikes midnight! = RB
- Overall, ~40% of word *tokens* are instances of ambiguous word *types*

Despite this, modern POS taggers still work quite well.

- Accuracy > 97%
- Simple baseline can achieve ~90%
  - Tag every word with its most frequent tag
  - Tag unknown words as nouns

How do POS taggers work?

- Several ways to predict POS tags:
  - Rule-based
  - Statistical
    - HMMs
    - Maximum Entropy Markov Models (MEMMs)
  - Transformation-based

Rule-Based POS Tagging

- Start with a dictionary, and assign all possible tags to the words in that dictionary
- Manually design rules to selectively remove invalid tags
- Keep the remaining correct tag for each word

Example Rule-Based Approach

- Start with a dictionary that specifies permissible tags for our small vocabulary:
  - she
    - PRP
  - promised
    - VBN, VBD
  - to
    - TO
  - back
    - VB, JJ, RB, NN
  - the
    - DT
  - bill
    - NN, VB

Example Rule-Based Approach

Assign every possible tag to each word in the sequence

she	promised	to	back	the	bill
PRP	VBN	TO	VB	DT	NN
	VBD		JJ		VB
			RB		
			NN		

Example Rule-Based Approach

Apply rules to eliminate invalid tags

Eliminate VBN if VBD is an option when VBN|VBD follows “<start> PRP”

she	promised	to	back	the	bill
PRP	~~VBN~~	TO	VB	DT	NN
	VBD		JJ		VB
			RB		
			NN		

Example Rule-Based Approach

Keep the remaining correct tag for each word

she	promised	to	back	the	bill
PRP	~~VBN~~	TO	VB	DT	NN
	VBD		~~JJ~~		~~VB~~
			~~RB~~		
			~~NN~~		

Rule-based POS taggers are an adequate baseline, but….

- Like all rule-based methods, they carry important disadvantages:
  - Time-consuming to build
  - Difficult to update or generalize to new domains
  - Might miss important patterns latent in the specified text domain

Nice alternative to rule-based POS tagging?

- **Statistical POS Tagging:** A category of POS taggers that works by exploiting learned knowledge of POS tag distribution in a training corpus
  - *the* is usually tagged as DT
  - Words with uppercase letters are more likely to be tagged NNP or NNPS
  - Words starting with the prefix *un-* may be tagged JJ
  - Words ending with the suffix *–ly* may be tagged RB

Statistical POS Tagging

- Predicts POS tags based on the probabilities of those tags occurring
- Probabilities can be based on various sources of information

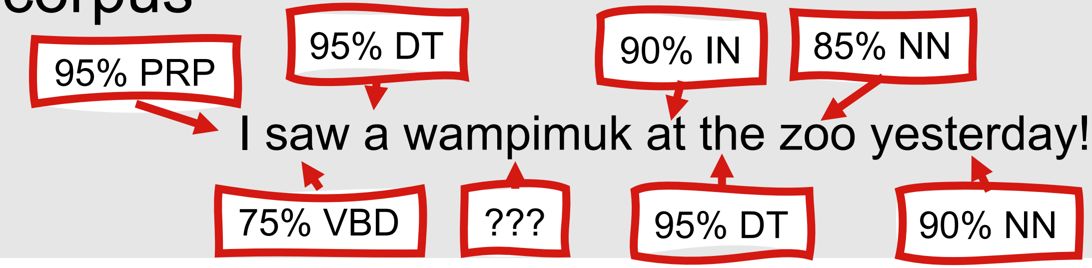
Simple Statistical POS Tagger

- Using a training corpus, determine the most frequent tag for each word
- Assign POS tags to new words based on those frequencies
- Assign NN to new words for which there is no information from the training corpus

I saw a wampimuk at the zoo yesterday!

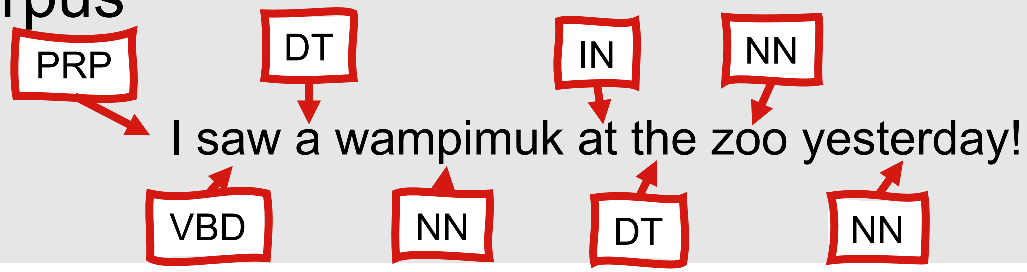
Simple Statistical POS Tagger

- Using a training corpus, determine the most frequent tag for each word
- Assign POS tags to new words based on those frequencies
- Assign NN to new words for which there is no information from the training corpus



Simple Statistical POS Tagger

- Using a training corpus, determine the most frequent tag for each word
- Assign POS tags to new words based on those frequencies
- Assign NN to new words for which there is no information from the training corpus



Simple Statistical POS Tagger

- This approach works reasonably well
  - Approximately 90% accuracy
- However, we can do much better!
- One way to improve upon our results is to use **HMMs**

HMM POS Tagger

- Selects the most likely tag sequence for a sequence of observed words, maximizing the following formula:
  - P(word | tag) * P(tag | previous *n* tags)
- More formally, letting $T = \{t_1, t_2, \ldots, t_n\}$ and $W = \{w_1, w_2, \ldots, w_n\}$, find the most probable sequence of tags *T* underlying the observed words *W*

What do we mean by “previous *n* tags”?

- For our example here, we’ll assume *n*=1 and create a bigram HMM tagger, meaning we’re only looking at a word/tag given the word/tag immediately preceding it

Bigram HMM Tagger

- To determine the tag t_i for a single word w_i:
  - $t_i = \underset{t_j \in \{t_0, t_1, \ldots, t_{t-1}\}}{\operatorname{argmax}} P(t_j | t_{i-1}) P(w_i | t_j)$
- This means we need to be able to compute two probabilities:
  - The probability that the tag is t_j given that the previous tag is t_{i-1}
    - $P(t_j | t_{i-1})$
  - The probability that the word is w_i given that the tag is t_j
    - $P(w_i | t_j)$
- We can compute both of these from corpora like the Penn Treebank or the Brown Corpus
- Then, we can find the most optimal sequence of tags using the Viterbi algorithm!

Example: Bigram HMM Tagger

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

- Given two possible sequences of tags for the following sentence, what is the best way to tag the word “race”?
- Brown Corpus tagset:
  - Contains a specific tag for the infinitive use of “to”
  - Labels “tomorrow” as NR (adverbial noun) rather than NN (singular common noun)

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

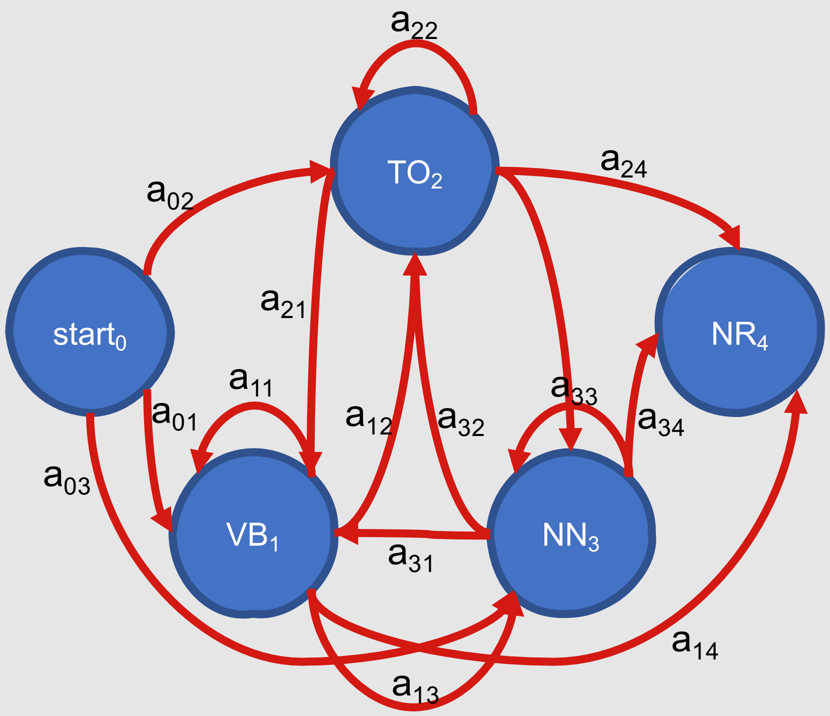
- Since we're creating a bigram HMM tagger and focusing on the word "race," we only need to be concerned with the subsequence "to race tomorrow"

Example: Bigram HMM Tagger

Example: Bigram HMM Tagger

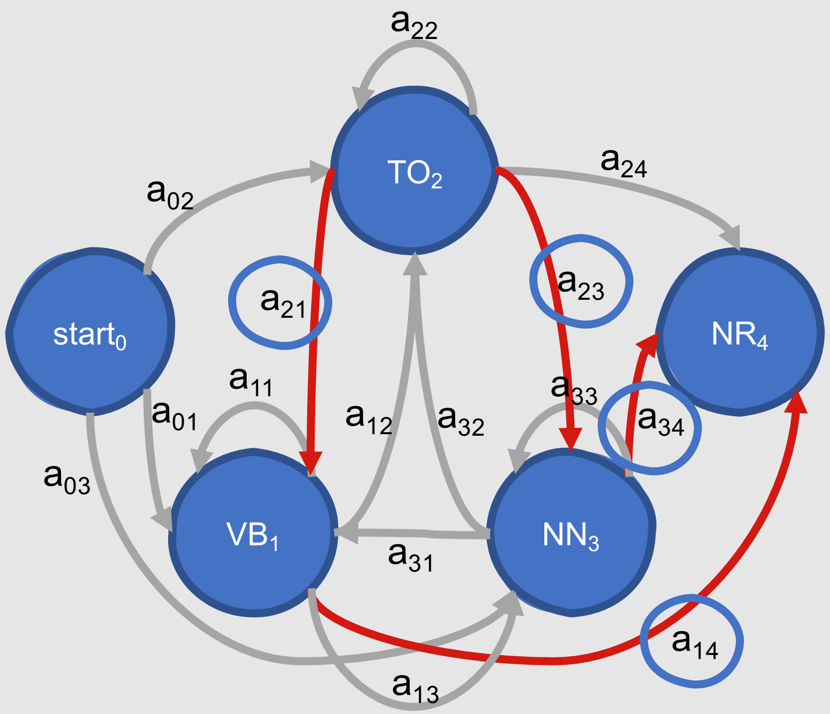
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

We can thus create the following Markov chain:



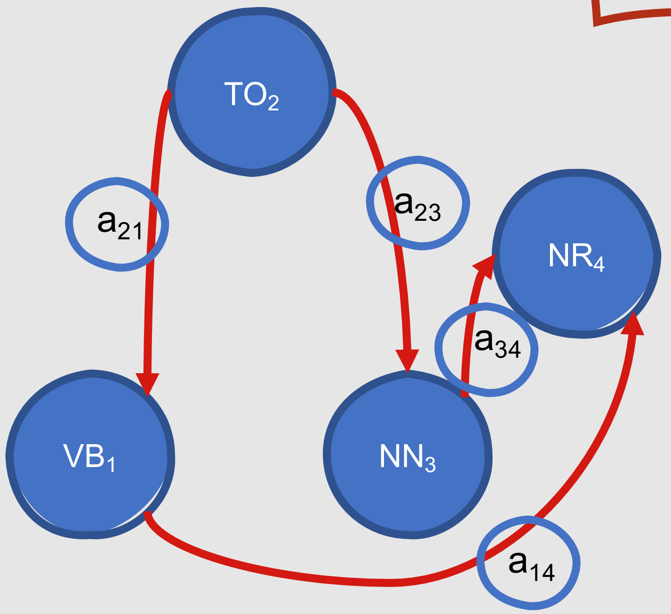
The specific transition probabilities we are interested in are:

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



Example: Bigram HMM Tagger

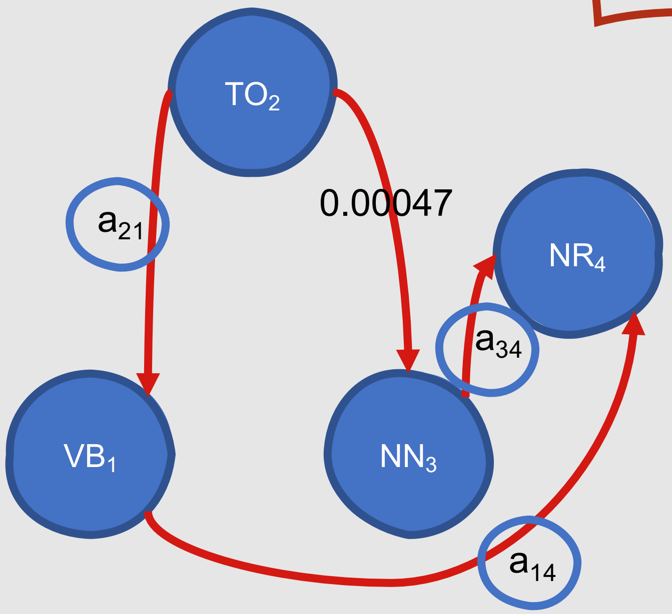
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can compute the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$

Example: Bigram HMM Tagger

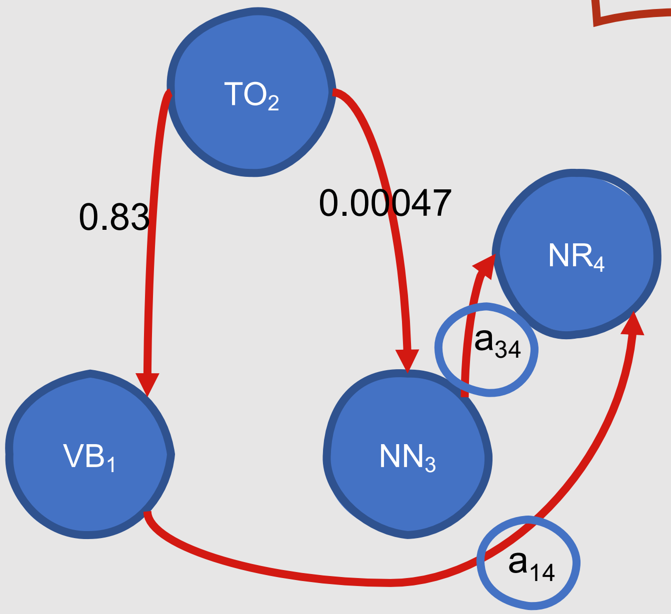
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can compute the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047

Example: Bigram HMM Tagger

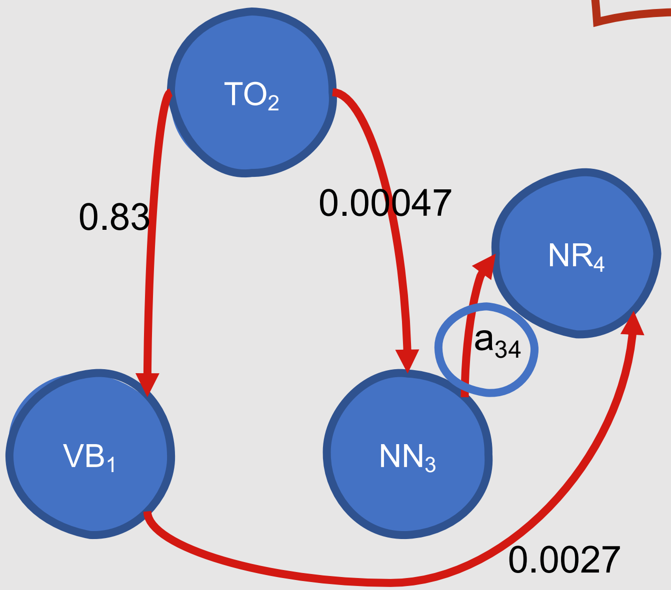
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can compute the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047
- Likewise, P(VB|TO) = C(TO VB) / C(TO) = 0.83

Example: Bigram HMM Tagger

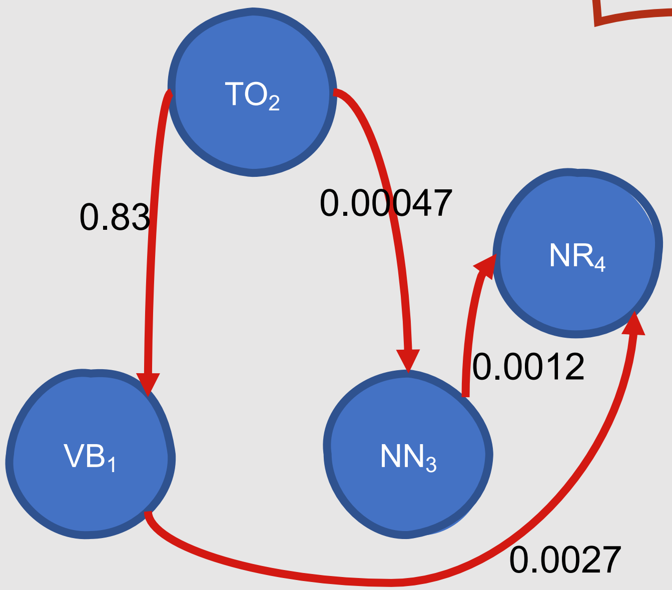
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can compute the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047
- Likewise, P(VB|TO) = C(TO VB) / C(TO) = 0.83
- P(NR|VB) = C(VB NR) / C(VB) = 0.0027

Example: Bigram HMM Tagger

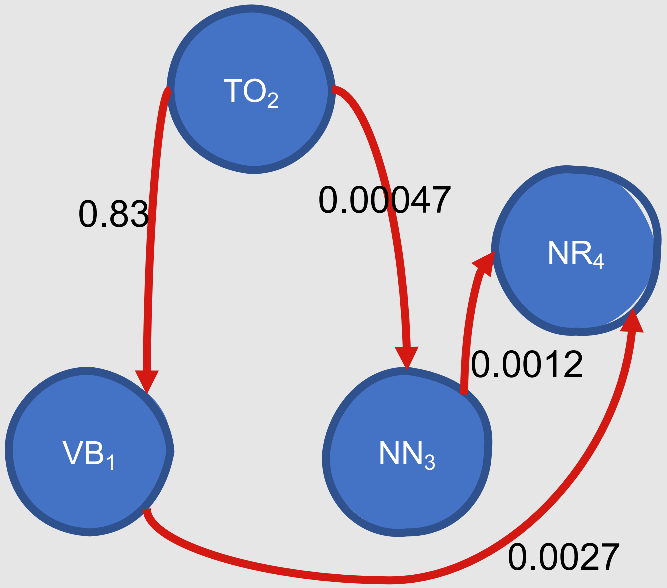
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can compute the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047
- Likewise, P(VB|TO) = C(TO VB) / C(TO) = 0.83
- P(NR|VB) = C(VB NR) / C(VB) = 0.0027
- Finally, P(NR|NN) = C(NN NR) / C(NN) = 0.0012

Example: Bigram HMM Tagger

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

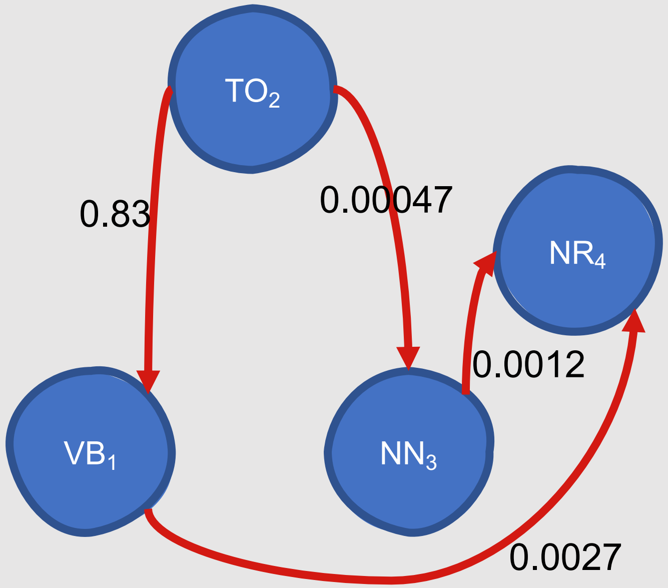


	race
VB	
NN	

- We have our transition probabilities …what now?
- Observation likelihoods!
- We can also compute these using frequency counts from the Brown Corpus
- $P(w_i|t_i) = \frac{c(w_i,t_i)}{c(t_i)}$
- Since we're trying to decide the best tag for "race," we need to compute both P(race|VB) and P(race|NN)

Example: Bigram HMM Tagger

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



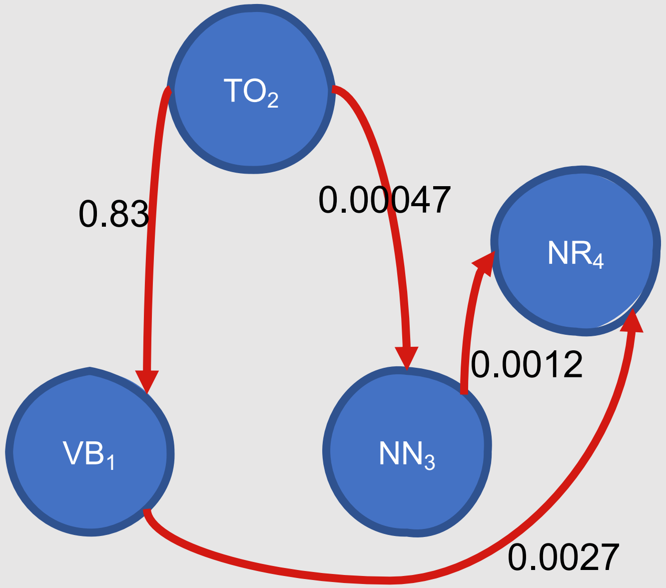
	race
VB	0.00012
NN	

- We have our transition probabilities …what now?
- Observation likelihoods!
- We can also compute these using frequency counts from the Brown Corpus
- $P(w_i|t_i) = \frac{c(w_i,t_i)}{c(t_i)}$
- Since we're trying to decide the best tag for "race," we need to compute both P(race|VB) and P(race|NN)
- P(race|VB) = C(race, VB) / C(VB) = 0.00012

Example: Bigram HMM Tagger

Example: Bigram HMM Tagger

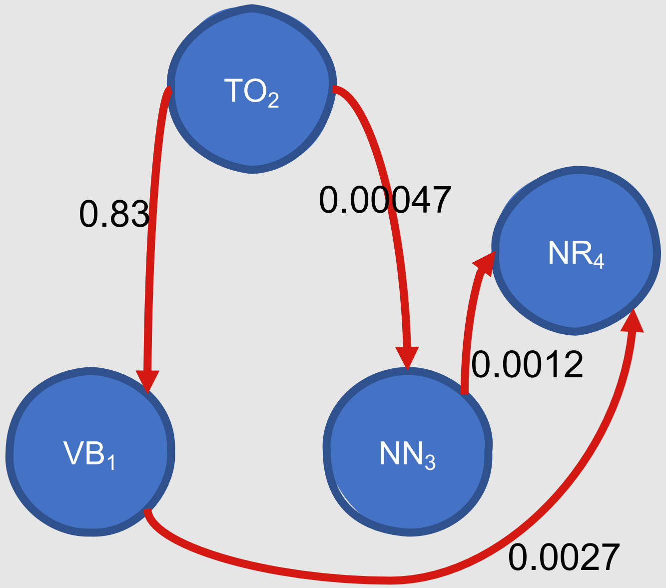
Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



	race
VB	0.00012
NN	0.00057

- We have our transition probabilities …what now?
- Observation likelihoods!
- We can also compute these using frequency counts from the Brown Corpus
- $P(w_i|t_i) = \frac{c(w_i,t_i)}{c(t_i)}$
- Since we're trying to decide the best tag for "race," we need to compute both P(race|VB) and P(race|NN)
- P(race|VB) = C(race, VB) / C(VB) = 0.00012
- P(race|NN) = C(race, NN) / C(NN) = 0.00057

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

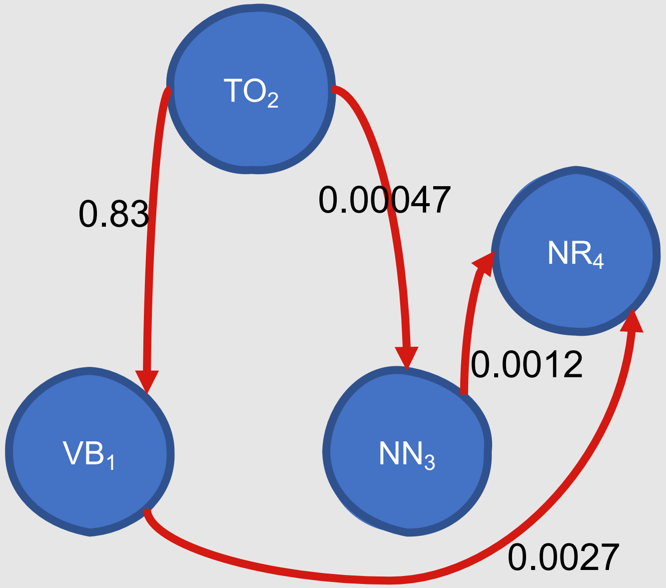


	race
VB	0.00012
NN	0.00057

- Now, to decide how to tag "race," we can consider our two possible sequences:
  - to (TO) race (VB) tomorrow (NR)
  - to (TO) race (NN) tomorrow (NR)
- We will select the tag that maximizes the probability:
  - $P(t_i|TO)P(NR|t_i)P(race|t_i)$
- We determine that:
  - P(VB|TO)P(NR|VB)P(race|VB) = 0.83 * 0.0027 * 0.00012 = 0.00000027
  - P(NN|TO)P(NR|NN)P(race|NN) = 0.00047 * 0.0012 * 0.00057 = 0.00000000032

Example: Bigram HMM Tagger

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	**TO**	**VB**	**NR**
NNP	VBZ	VBN	TO	NN	NR

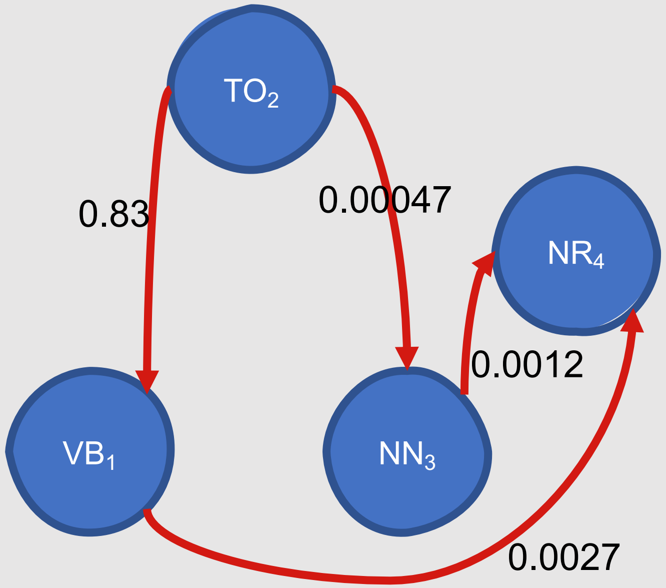


	race
VB	0.00012
NN	0.00057

- Now, to decide how to tag "race," we can consider our two possible sequences:
  - to (TO) race (VB) tomorrow (NR)
  - to (TO) race (NN) tomorrow (NR)
- We will select the tag that maximizes the probability:
  - $P(t_i|TO)P(NR|t_i)P(race|t_i)$
- We determine that:
  - **P(VB|TO)P(NR|VB)P(race|VB) = 0.83 * 0.0027 * 0.00012 = 0.00000027**
    - Optimal sequence!
  - P(NN|TO)P(NR|NN)P(race|NN) = 0.00047 * 0.0012 * 0.00057 = 0.00000000032

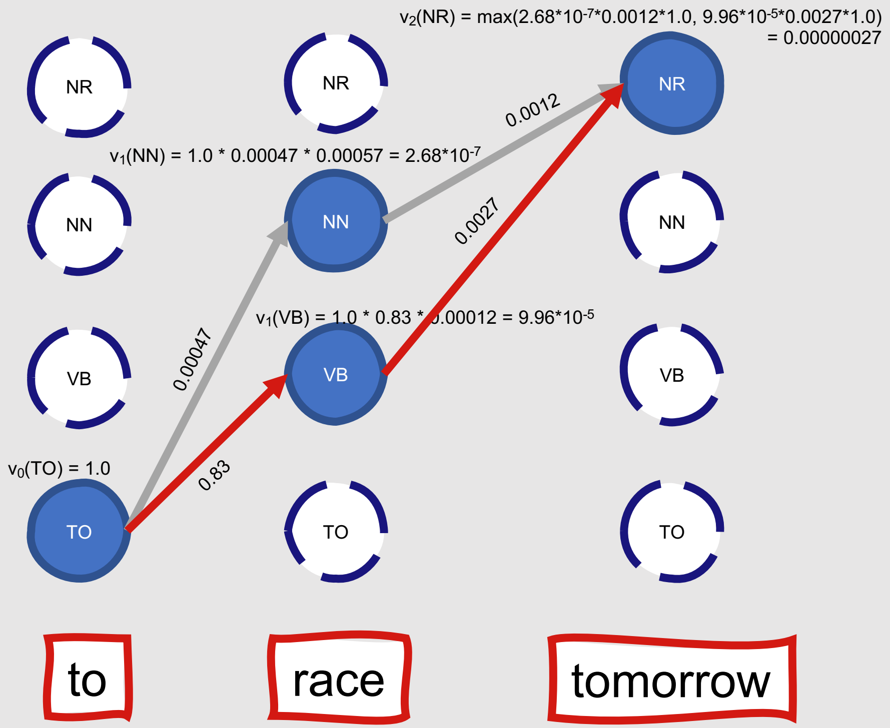
Example: Bigram HMM Tagger

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	**TO**	**VB**	**NR**
NNP	VBZ	VBN	TO	NN	NR



	race
VB	0.00012
NN	0.00057

- Visualized in a Viterbi trellis, this would look like:



Example: Bigram HMM Tagger

What if we used greater values of *n*?

- For example, a trigram HMM tagger instead of a bigram HMM tagger?
- Generally, more context → more accurate predictions
- However, greater values of *n* also require more computational work …you need to determine whether the trade-off is worth it

Transformation-Based POS Tagging

A popular method in the past that leverages a combination of rule-based and statistical methods

Automatically induces rules from a training corpus, and then applies them in a manner similar to that seen with rule-based models

Transformation-Based POS Tagging

- Basic Idea
  - **Set the most probable tag** for each word as a start value
  - **Change tags according to rules** in a specific order
    - For example, "if w_1 is a determiner and w_2 is a verb, than change the tag for w_2 to noun"
- **Learn these rules from a tagged corpus**
  - From start value, examine every possible transformation
  - Select the one that results in the most improved tagging (see example above)
  - Re-tag data according to this rule
  - Repeat previous two steps until stopping criterion is met
- Thus, **rules can make errors that are corrected by later rules**

Example Rule

- Start: Tagger labels every word with its most likely tag
  - P(NN|race) = 0.98
  - P(VB|race) = 0.02

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	NN	NR

- New rule learned: Change NN to VB when previous tag is TO
- Re-tag data according to this rule

Secretariat	is	expected	to	race	tomorrow
NNP	VBZ	VBN	TO	VB	NR

In theory, endless rules could be learned!

- In practice, this would be problematic:
  - Significant computational overhead
  - Prone to overfitting

Example Transformation-Based Tagger

- Brill tagger: https://dl.acm.org/doi/10.3115/974499.974526
- Addressed the problem of potentially unlimited rules by creating a small set of templates to which all rules had to adhere
  - Change tag a to tag b when the preceding (following) word is tagged z.
  - Change tag a to tag b when the word two before (after) is tagged z.
  - Change tag a to tag b when one of the two preceding (following) words is tagged z.
  - Change tag a to tag b when one of the three preceding (following) words is tagged z.
  - Change tag a to tag b when the preceding word is tagged z and the following word is tagged w.
  - Change tag a to tag b when the preceding (following) word is tagged z and the word two before (after) is tagged w.

Comparing POS Tagging Methods

- Generally, **rule-based approaches are faster and may work better for limited, well-defined domains**
- On the other hand, **statistical approaches are slower and may generalize better across broader domains**
  - HMM-based taggers can easily be trained on new languages, whereas rule-based taggers would have to be completely rewritten
- Statistical POS taggers are the most common in modern applications
  - State of the art statistical POS taggers use neural network architectures
  - Other strong models are HMM-based and CRF-based approaches

How can POS taggers handle unknown words?

- New words are continually added to language, so it is likely that a POS tagger will encounter words not found in its training corpus
- Easy baseline approach: **Assume that unknown words are nouns**
- More sophisticated approach: **Assume that unknown words have a probability distribution similar to other words occurring only once in the training corpus**, and make an (informed) random choice
- Even more sophisticated approach: **Use morphological information** to choose the POS tag (for example, words ending with “ed” tend to be tagged VBN)

How are POS taggers evaluated?

- POS taggers are typically learned using (or rules are written based on) a training set, and then their performance is evaluated using a separate test set
- We can adapt the standard measures for text classification that we've already learned about to evaluate the predicted tags compared to the gold standard

Evaluation Metrics

- Common metrics for POS taggers are:
  - Accuracy
  - Precision (of the words predicted to be NN, how many were labeled as NN by humans?)
  - Recall (of the words labeled NN by humans, how many were predicted to be NN by the POS tagger?)
  - F-Measure (combination of precision and recall)

Comparison

- The scores computed for these metrics should be compared to alternative POS tagging methods, to place the values in context
  - Is this a good accuracy score, or just a so-so one?
- It's good to compare to both a lower-bound baseline and an upper-bound ceiling
  - Baseline: What should your POS tagger definitely perform better than?
    - Most Frequent Class
  - Ceiling: What is the highest possible value for this task?
    - Human Agreement

What factors can impact performance?

- Many factors can lead to your results being higher or lower than expected!
- Some common factors:
  - The size of the training dataset
  - The specific characteristics of your tag set
  - The difference between your training and test corpora
  - The number of unknown words in your test corpus

Summary: Part-of-Speech Tagging

POS tagging is the process of automatically assigning grammatical word classes (parts of speech) to individual tokens

The most common POS tagset is the **Penn Treebank** tagset

Ambiguity is common in natural language, and is a major issue that POS taggers must address

POS taggers can be rule-based, statistical, or transformation-based

Statistical POS taggers are most common and usually use neural approaches, HMMs, or CRFs

POS tags are one way to formalize language structure.

- Constituency grammars are another!
- Constituency grammars are:
  - A **set of rules** that describe how a language can be structured
  - A **lexicon** that defines the words and symbols that belong to the language

Constituency Grammars

- Function at the sentence level
  - Rather than at the word level like POS tagging
- Provide the necessary structure to answer important questions:
  - What are the **constituents** (groups of words that behave as a single unit or phrase) in this sentence?
  - What are the **grammatical relations** between these constituents?
  - Which words are **dependent** upon one another?

Although the models we've seen that focus on words model sentences as sequences, **formal grammars model sentences as recursive generating processes**.

How do they do this?

Usually, a tree structure

It’s all about finding the right balance!

- When constructing formal grammars, we want to strike a balance between:
  - **Capturing all of the sentence structures that are valid** for a given language
  - **Avoiding the sentence structures that are invalid**
- As usual, this is easier said than done!

English Grammar

Overgeneration:

Love NLP class my so much that don't care about being it after lunch right!

Did get the you email guy that that from class said he forward to you would?

Well, there just happened.

English:

I love my NLP class so much that I don't even care about it being right after lunch!

Did you get the email that that guy from class said he would forward to you?

Well, that just happened.

Undergeneration:

I love my class!

Did you get his email?

What happened?

Two terms to be aware of….

- **Grammar Formalisms:** A precise way to define and describe the structure of independent sentences.
  - There are many different grammar formalisms (you can learn much more about these in linguistics courses!)
- **Specific Grammars:** Implementations (according a specific formalism) for a particular language
  - English, Arabic, Mandarin, or Hindi
- Grammar Formalisms : Specific Grammars :: Programming Languages : Programs

Is it possible to define a grammar that generates all English sentences?

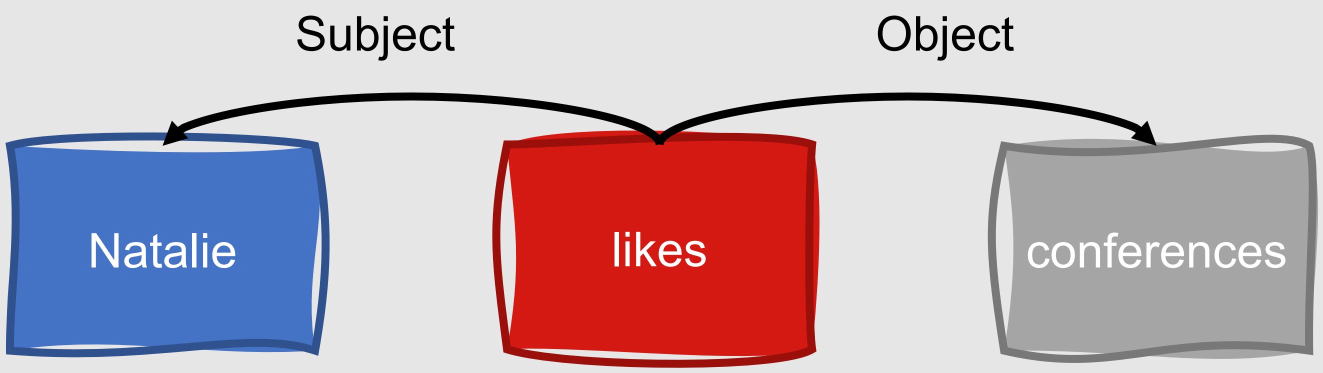
- Tricky question!
- The number of possible English sentences is infinite, but our grammar needs to be finite
- There *are* specific grammars that do a very good job at generating English sentences

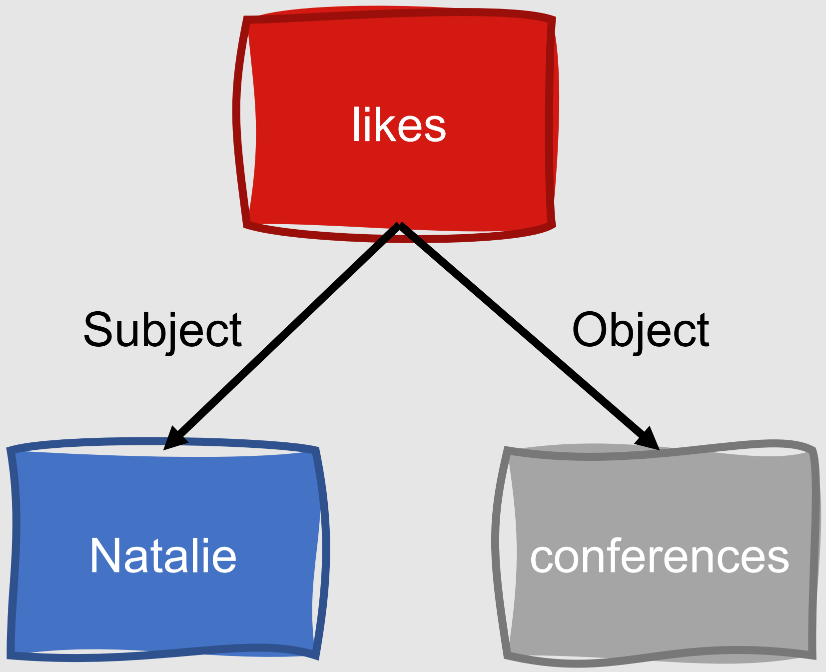
Basic English Sentence Structure

Natalie	likes	conferences
Noun (Subject)	Verb (Head)	Noun (Object)

There are many ways to represent a sentence!

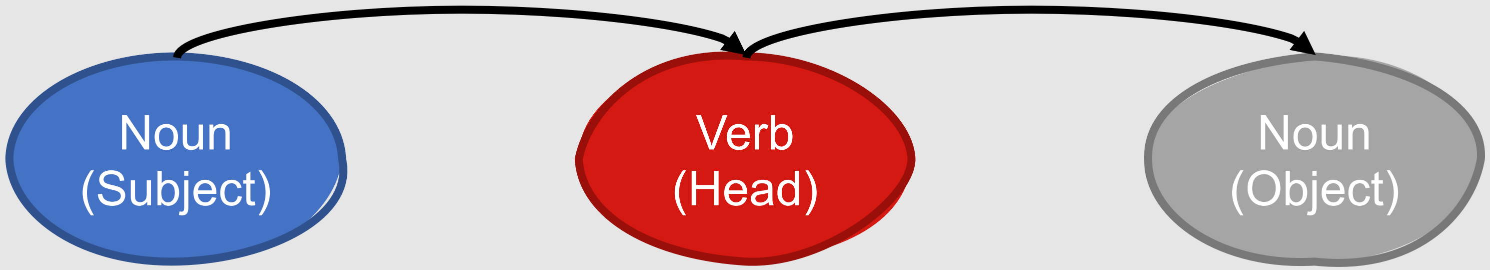
As a dependency graph:





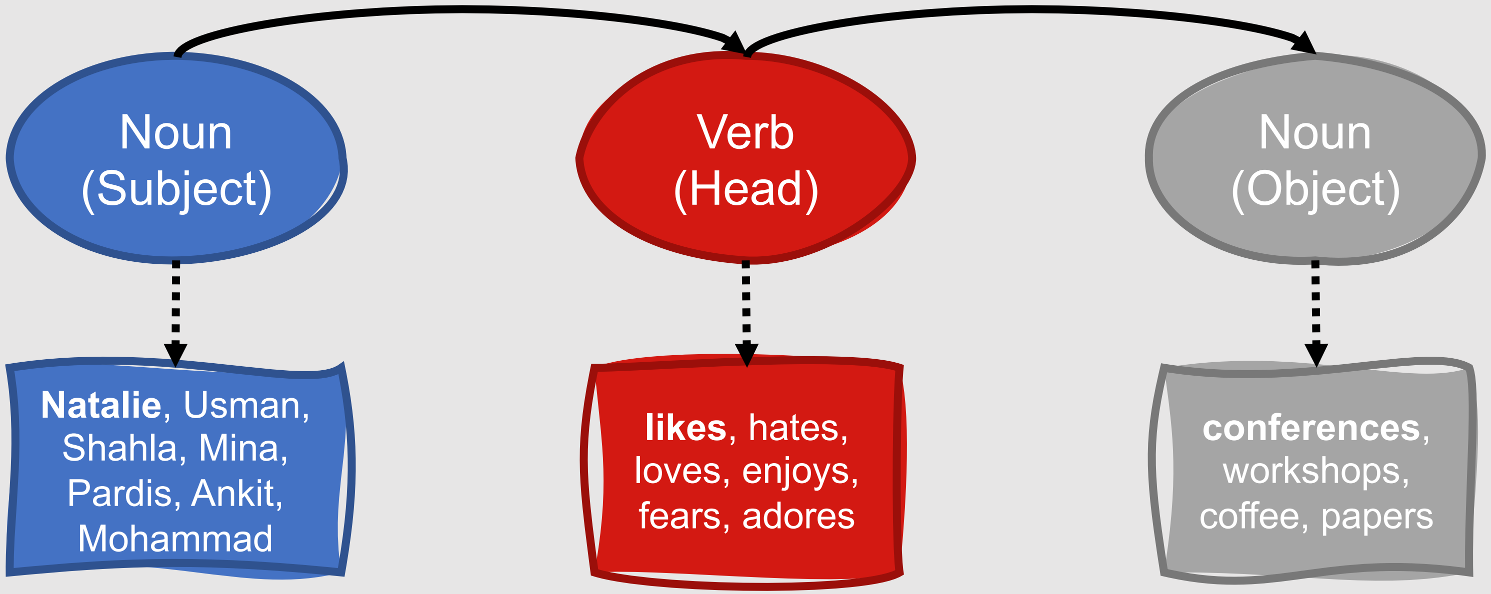
There are many ways to represent a sentence!

As a finite state automaton:



There are many ways to represent a sentence!

As a hidden Markov model:



Different types of words accept different types of **arguments**.

Natalie likes conferences. 🙂

Natalie drinks conferences. 🤨

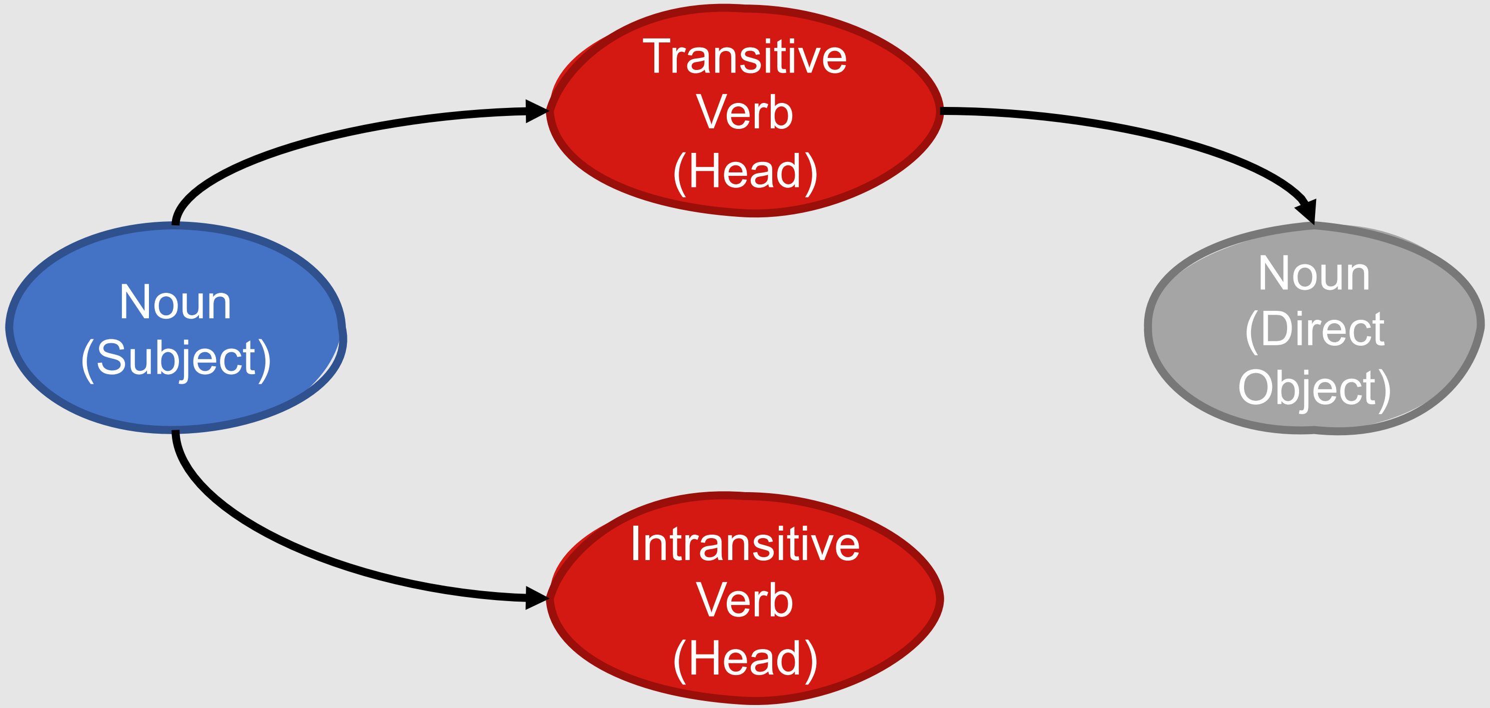
Some more terms to be aware of….

- **Subcategorization:** Syntactic constraints on the set of arguments that a group of words will accept.
  - **Intransitive verbs** accept only subjects
    - Sleep, arrive
  - **Transitive verbs** accept a subject and a direct object
    - Eat, drink
  - **Ditransitive verbs** accept a subject, a direct object, and an indirect object
    - Give, make

Some more terms to be aware of….

- **Selectional Preference:** Semantic constraints on the set of arguments that a group of words will accept.
  - The object of “drink” should be edible.
    - Natalie drinks conferences. 🤨
    - Natalie drinks tea. 🙂☕

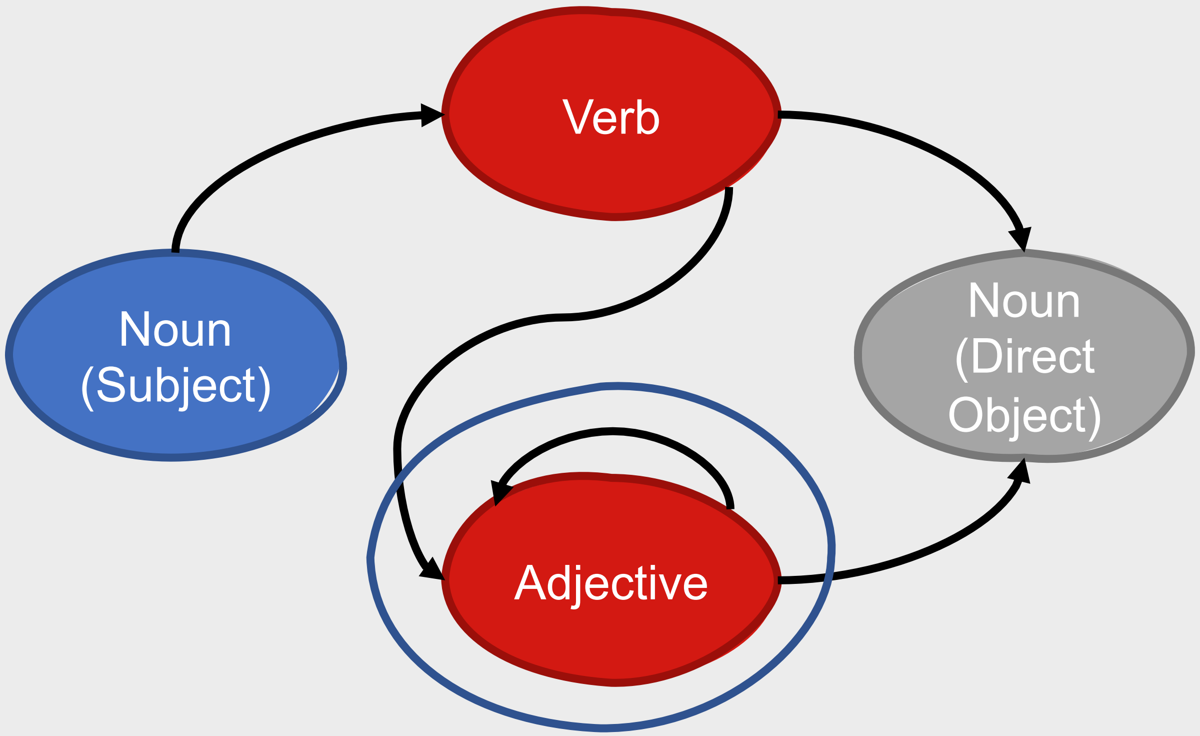
We might represent these as a finite state model like this:



One of the reasons why the number of possible English sentences is infinite?

- Language is recursive!
- In theory, we can have unlimited modifiers (adjectives and adverbs)
  - Natalie likes conferences.
  - Natalie likes academic conferences.
  - Natalie likes busy academic conferences.

We can easily model simple cases of recursion in a finite state model as well.



However, recursion in sentences can also be more complex.

Natalie likes conferences.

Natalie likes conferences **in Europe**.

Natalie likes conferences **in Europe in the summer**.

Still, can't we just make complex FSAs?

- FSAs can model recursion, but they can't model hierarchical structure
- In complex sentences, you must also handle **attachment ambiguity**

Natalie **likes conferences in** Europe **or** Asia.

Natalie likes conferences in either Europe or Asia.

Natalie **likes** conferences in Europe **or** Asia.

Natalie likes two things: Asia, or conferences in Europe.

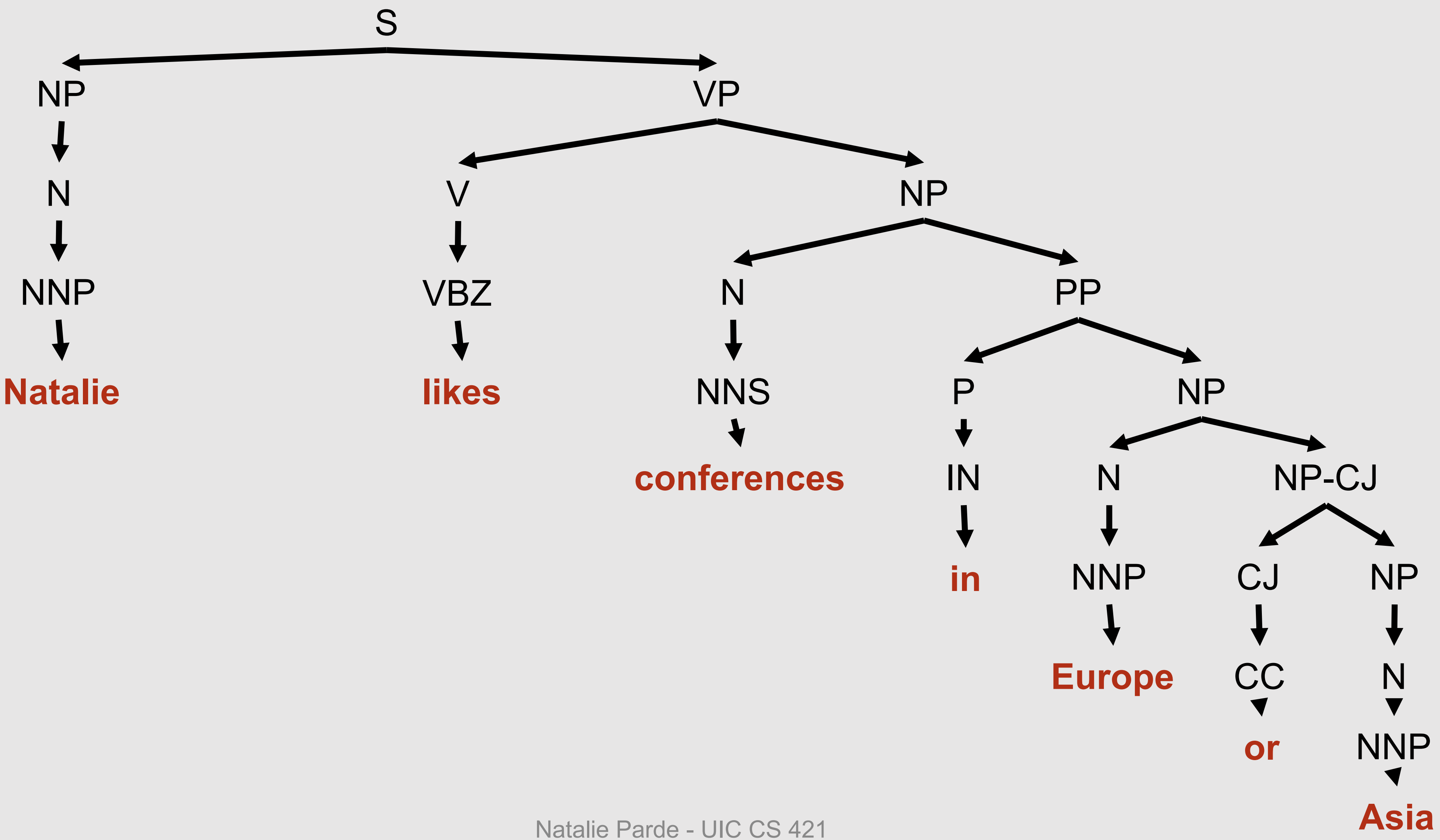
Sentences Form a Hierarchy

- A sentence consists of words that can be grouped into phrases (**constituents**)
- **Sentence structure** defines dependencies between these constituents

We can use trees to model this hierarchy.

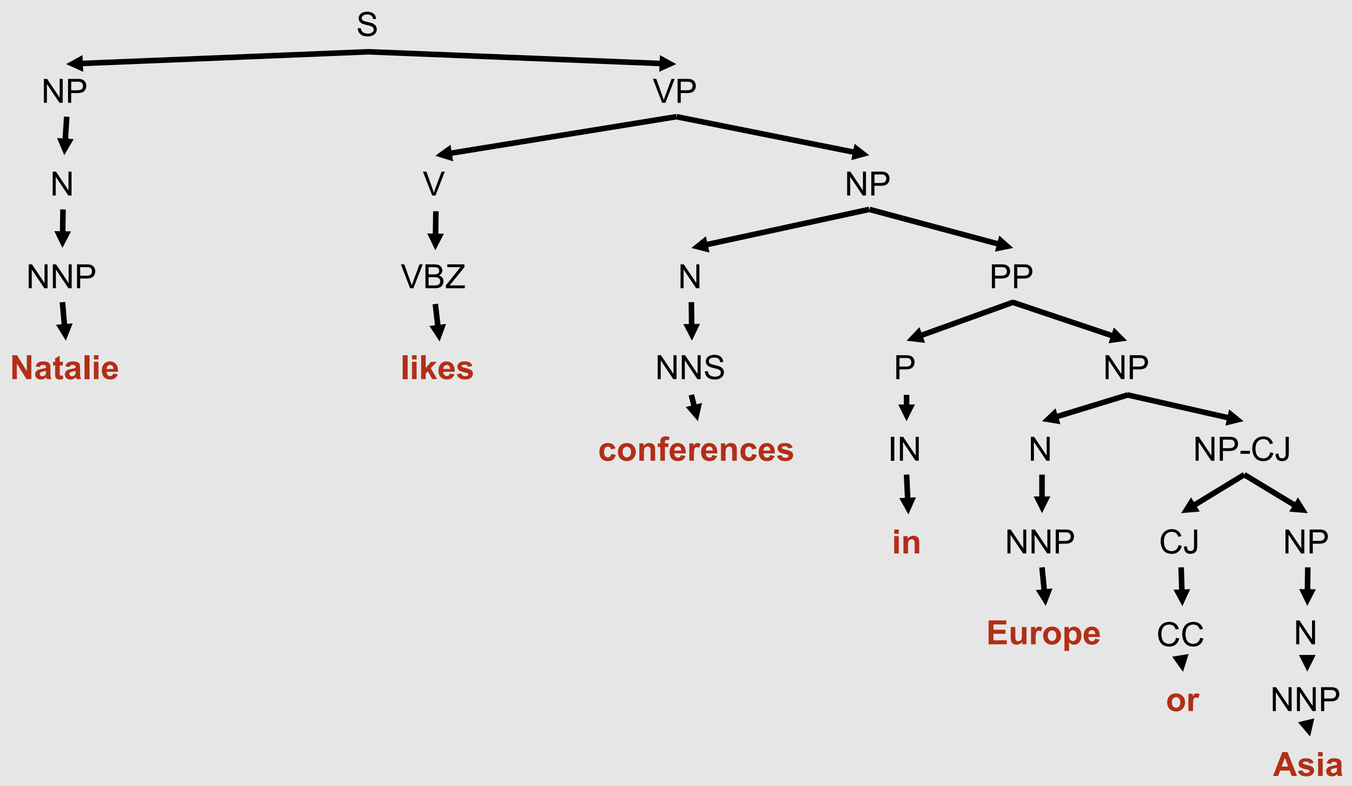
- Formal trees will usually have **internal (non-terminal) nodes** and **outer (terminal) leaves**
- **Nodes: Elements of sentence structure**
  - Constituent type
  - POS type
- **Leaves: Surface wordforms**
- The nodes and leaves are connected to one another by **branches**

What does this look like?



Trees can grow to be quite complex!

However, they can be reduced to simple subtrees defining underlying syntactic constituents



The grammars defining these hierarchical trees are context-free grammars.

- **Context-Free Grammar (CFG):** A mathematical system for modeling constituent structure in natural language.
- Also called **Phrase-Structure Grammars**
- CFGs can describe all regular languages
- Why is it called context-free?
  - A subtree can be replaced by a production rule independent of the greater context (other nodes in the hierarchy) in which it occurs.

CFGs are defined by productions that indicate which strings they can generate.

- **Production:** Rules expressing the allowable combinations of symbols (e.g., POS types) that can form a constituent
- Productions can be **hierarchically embedded**
  - Noun Phrase (NP) → Determiner Nominal
  - Nominal → Noun | Nominal Noun

Production rules determine how constituents can be combined.

- **Constituent:** A group of words that behaves as a single unit.
  - Noun Phrase: the woman, the woman with red hair, the last conference of the year
  - Prepositional Phrase: with red hair, of the year
  - Verb Phrase: drinks tea, likes going to conferences
- Phrases contain **heads** and **dependents**
  - **Heads:** the **woman** with red hair, the last **conference** of the year
  - **Dependents:** **the** woman **with red hair**, **the last** conference **of the year**

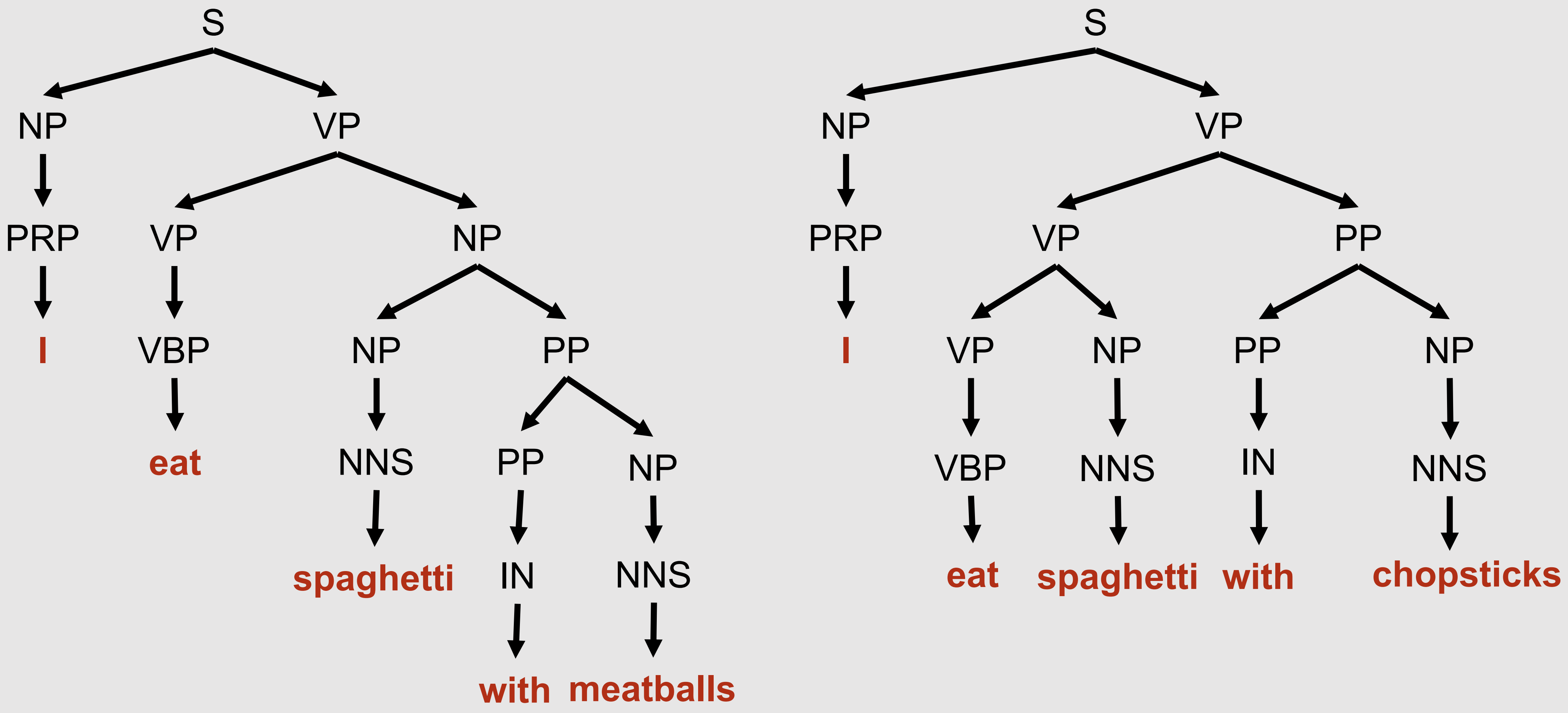
A Little More About Dependents

- Dependents can be arguments or adjuncts
- Arguments are **obligatory**
  - Natalie likes *conferences*. 🙂
  - Natalie likes. 🤨
- Adjuncts are **optional**
  - Natalie drinks *tea*. 🙂
  - Natalie drinks. 🙂

Properties of Constituents

- **Constituents can be substituted with one another** in the context of the greater sentence
  - **The woman with red hair** rolled her eyes as lightning immediately struck the man's house.
  - **The unicorn** rolled her eyes as lightning immediately struck the man's house.
- **A constituent can move around** within the context of the sentence
  - **The woman with red hair** rolled her eyes as lightning immediately struck the man's house.
  - Lightning immediately struck the man's house as **the woman with red hair** rolled her eyes.
- **A constituent can be used to answer a question** about the sentence
  - Who rolled her eyes? **The woman with red hair**.

The structure of constituents in a tree corresponds to their meaning.



Case Example

- Draw a constituent tree for the sentence:
  - **Time flies like an arrow.**

Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	

Case Example

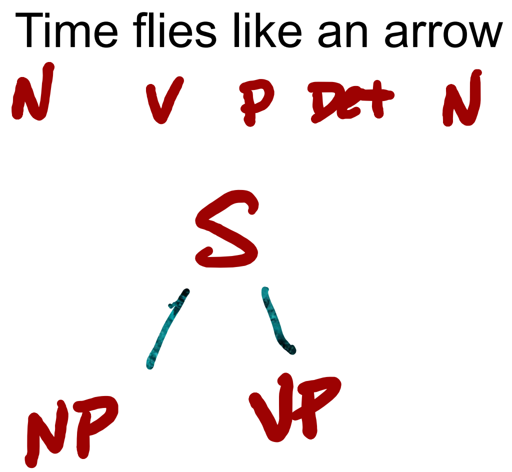
Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	

Time flies like an arrow

N V P Det N

Case Example

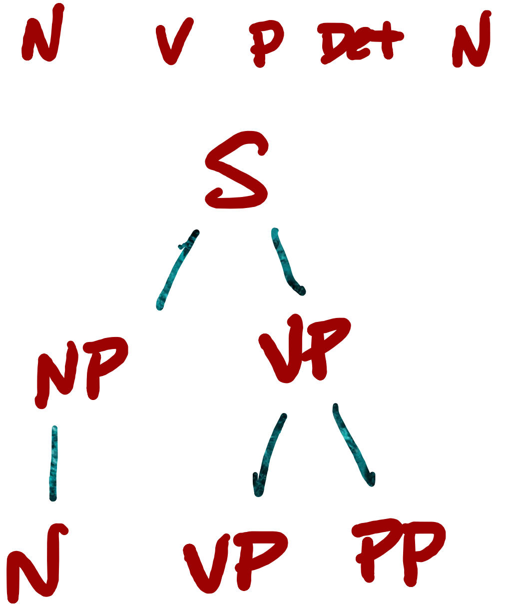
Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	



Case Example

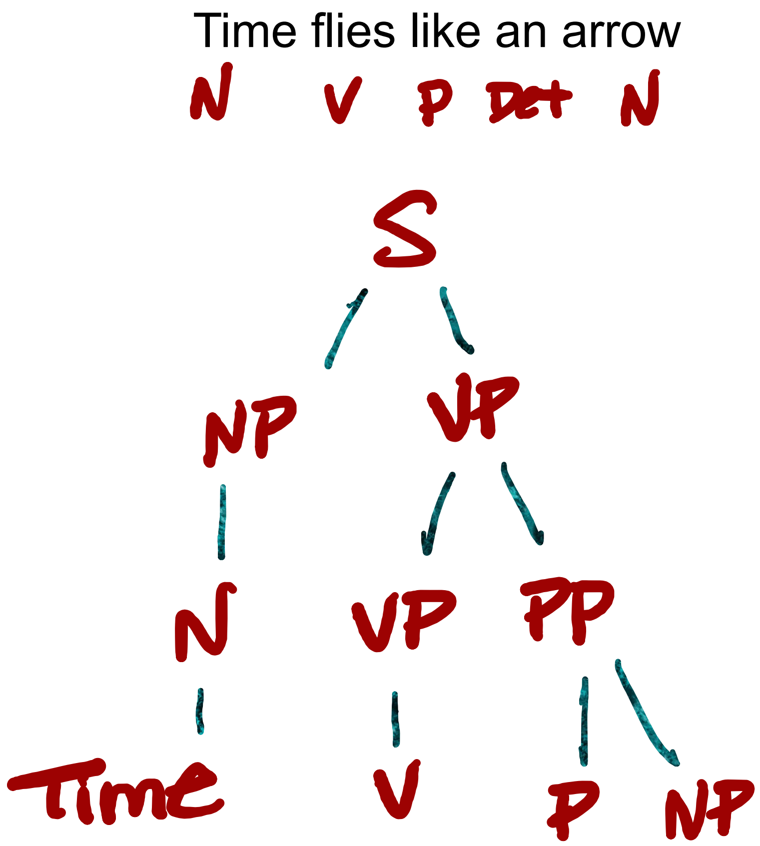
Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	

Time flies like an arrow



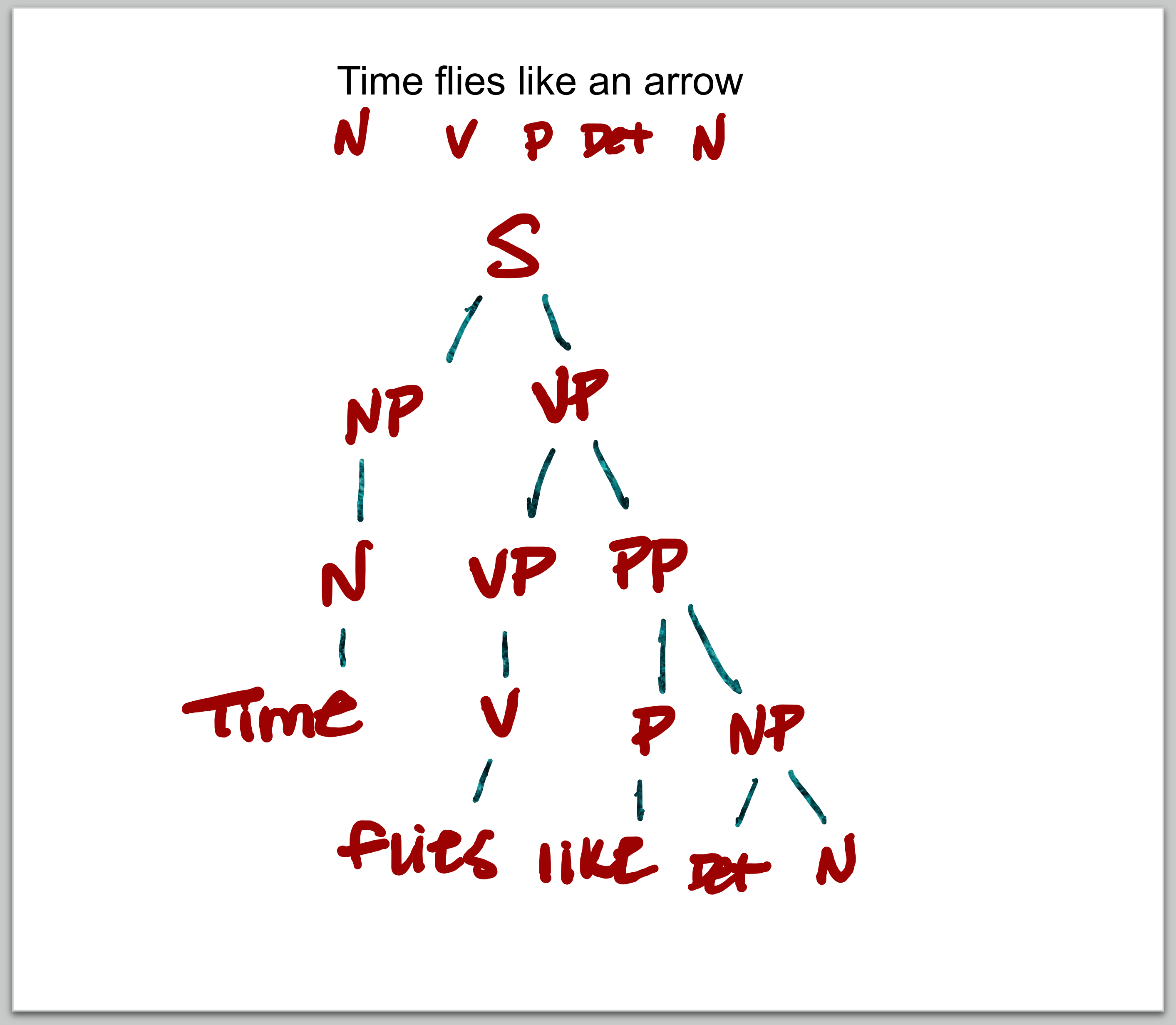
Case Example

Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	



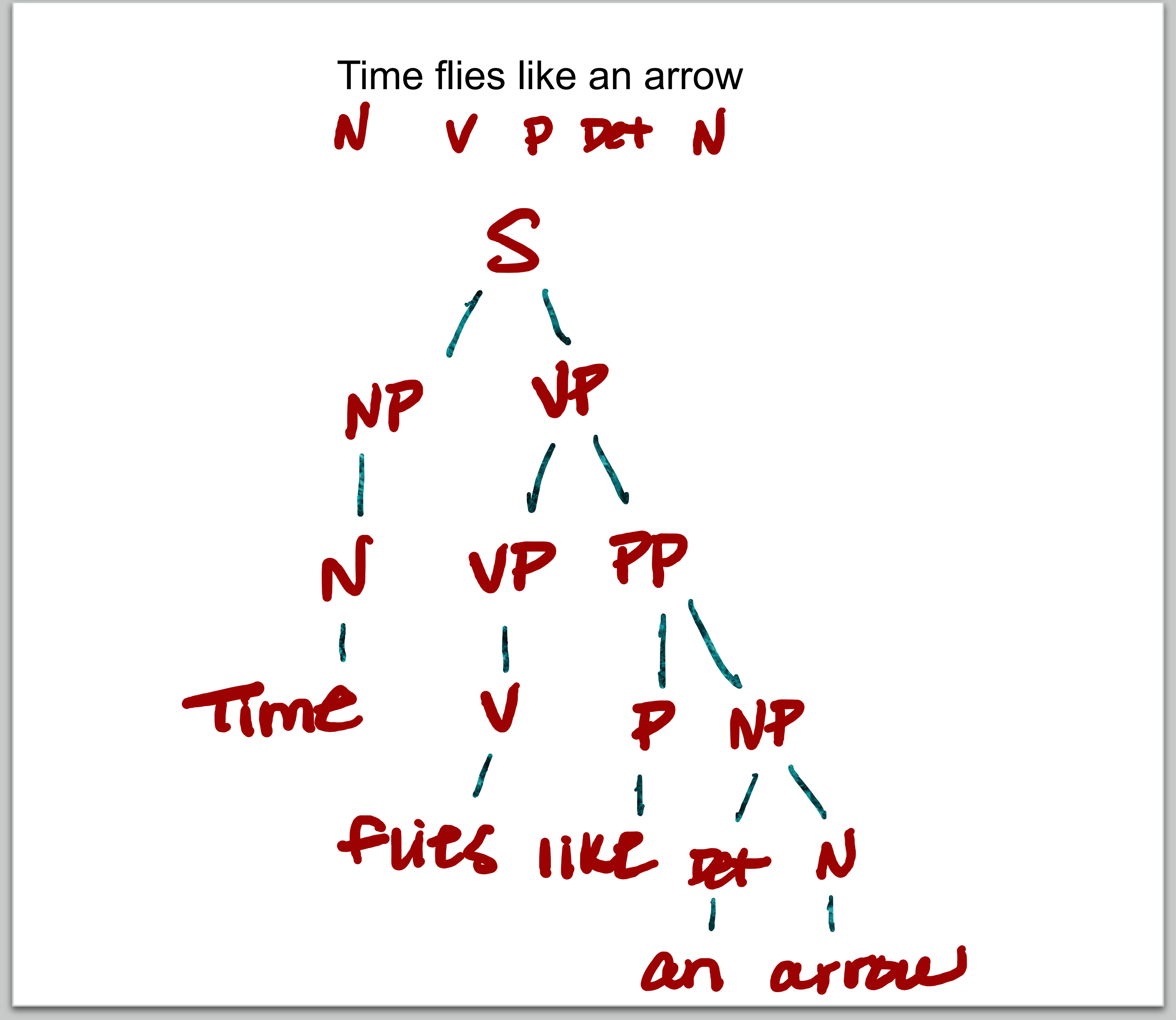
Case Example

Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	



Case Example

Production Rules	
S ! NP VP	PP ! P NP
NP ! DET N	PP ! P
NP ! N	P ! like
NP ! N N	V ! flies | like
VP ! VP PP	DET ! a | an
VP ! V NP	N ! time | fruit | flies | arrow | banana
VP ! V	



Formal Definition

- A CFG is a 4-tuple $\langle \boldsymbol{N}, \boldsymbol{\Sigma}, \boldsymbol{R}, \boldsymbol{S} \rangle$ consisting of:
  - A set of non-terminal nodes ***N***
    - ***N*** = {S, NP, VP, PP, N, V, …}
  - A set of terminal nodes (leaves) $\boldsymbol{\Sigma}$
    - $\boldsymbol{\Sigma}$ = {time, flies, like, an, arrow, …}
  - A set of rules ***R***
  - A start symbol ***S*** ∈ ***N***

Which sentences are grammatically correct?

- Any sentences for which the CFG can construct a tree (all words in the sentence must be reachable as leaf nodes) are accepted by the CFG.

What about really complex sentences?

Natalie knew a lot. 🙂

The zebra **that Natalie knew** knew a lot. 😕

The unicorn **that the zebra that Natalie knew knew** knew a lot. 🤯

CFGs and Center Embedding

- Formally, these sentences are all grammatical, because they can be generated by the CFG that is required for the first sentence:
  - S → NP VP
  - NP → NP RelClause
  - RelClause → that NP ate
- However, very few humans would consider the last sentence to be grammatically correct!

CFGs and Center Embedding

- **CFGs are unable to capture bounded recursion** (e.g., embedding only one relative clause)
- So, linguists acknowledge that formal grammaticality is not perfectly equivalent to human perception of grammaticality
  - They additionally consider human grammatical knowledge, as well as processing and memory limitations
- In the context of this class, we'll just assume that if something is accepted by a CFG, it is grammatically correct

Refresher: Typical CFG Constituents (English)

- Noun phrases (NPs)
  - Simple:
    - **She** talks. (**pronoun**)
    - **Natalie** talks. (**proper noun**)
    - **A person** talks. (**determiner** + **common noun**)
  - Complex:
    - **A professorial person** talks. (**determiner** + **adjective** + **common noun**)
    - **The person at the lectern** talks. (**noun phrase (determiner + common noun)** + **prepositional phrase**)
    - **The person who teaches NLP** talks. (**noun phrase (determiner + common noun)** + **relative clause**)

Refresher: Typical CFG Constituents (English)

- Visualized as production rules:
  - NP → Pronoun
  - NP → Proper Noun
  - NP → Determiner Common Noun
  - NP → Determiner Adjective Common Noun
  - NP → NP PP
  - NP → NP RelClause

  - Pronoun → {she}
  - Determiner → {a}
  - Proper Noun → {Natalie}
  - Common Noun → {person}
  - Adjective → {professorial}

Refresher: Typical CFG Constituents (English)

- Adjective Phrases (AdjP) and Prepositional Phrases (PP)
  - AdjP → Adjective
  - AdjP → Adverb AdjP
  - Adj → {professorial}
  - Adv → {very}
    - A very professorial person talks.

  - PP → Preposition NP
  - Preposition → {at}

Refresher: Typical CFG Constituents (English)

- Verb Phrases (VPs)
  - She **drinks**. (**verb**)
  - She **drinks** **tea**. (**verb** + **noun phrase**)
  - She **drinks tea** **from a mug**. (**verb phrase** + **prepositional phrase**)
- Visualized as production rules:
  - VP → V
  - VP → V NP
  - VP → V NP PP
  - VP → VP PP
  - V → {drinks}

Refresher: Typical CFG Constituents (English)

- We can also capture subcategorization this way!
  - She **drinks**. (**verb**)
  - She **drinks** **tea**. (**verb** + **noun phrase**)
  - She **gives** **him** **tea**. (**verb phrase** + **noun phrase** + **noun phrase**)
- Visualized as production rules:
  - $VP \rightarrow V_{intransitive}$
  - $VP \rightarrow V_{transitive}\ NP$
  - $VP \rightarrow V_{ditransitive}\ NP\ NP$
  - $V_{intransitive} \rightarrow \{drinks, talks\}$
  - $V_{transitive} \rightarrow \{drinks\}$
  - $V_{ditransitive} \rightarrow \{gives\}$

Refresher: Typical CFG Constituents (English)

- Production rules can also recursively include sentences
  - She drinks tea. (noun phrase + verb phrase)
  - Sometimes, she drinks tea. (adverbial phrase + sentence)
  - In England, she drinks tea. (prepositional phrase + sentence)
- Visualized as production rules:
  - $S \rightarrow NP\ VP$
  - $S \rightarrow AdvP\ S$
  - $S \rightarrow PP\ S$

To comprehensively cover English grammar, more complex production rules are necessary.

- She drinks tea.
- I drinks tea.
- They drinks tea.

- To avoid situations like the above, the simpler $S \rightarrow NP\ VP$ could be expanded to:
  - $S \rightarrow NP_{3sg}\ VP_{3sg}$
  - $S \rightarrow NP_{1sg}\ VP_{1sg}$
  - $S \rightarrow NP_{3pl}\ VP_{3pl}$

CFG Covering English Verb Tenses

- Present Tense: She drinks tea.
- Simple Past Tense: She drank tea.
- Past Perfect Tense: She has drunk tea.
- Future Perfect Tense: She will have drunk tea.
- Passive: The tea was drunk by her.
- Progressive: She will be drinking tea.

- $VP \rightarrow V_{have}\ VP_{pastPart}$
- $VP \rightarrow V_{be}\ VP_{pass}$
- $VP_{pastPart} \rightarrow V_{pastPart}\ NP$
- $VP_{pass} \rightarrow V_{pastPart}\ PP$
- $V_{have} \rightarrow \{has\}$
- $V_{pastPart} \rightarrow \{drunk\}$
- etc.…

Multiple sentences or clauses can be coordinated with one another via conjunction.

- **She drinks tea** and **he drinks coffee**.
- **Natalie** and **her mom** drink tea.
- She **drinks tea** and **eats cake**.

- S → S conj S
- NP → NP conj NP
- VP → VP conj VP

Relative Clauses

- **Relative clauses modify a noun phrase** by adding extra information
  - She had **a poodle** **that drank my tea**.
- Importantly, relative clauses do not have their own noun phrase!
  - Instead, it is understood that the NP is filled by the NP that the relative clause is modifying
    - She had a poodle **that** drank my tea. → that = a poodle
- There are two types of relative clauses
  - Subject: She had a poodle **that drank my tea**.
    - We cannot drop the relative pronoun
  - Object: I’d really been enjoying the tea **that her poodle drank**.
    - We can drop the relative pronoun and the sentence still works

The only things remaining are questions!

Yes/No Questions

- Auxiliary + Subject + Verb Phrase
  - Does she drink tea?
- YesNoQ → Aux NP VP

Wh-Questions

- Subject wh-questions contain a wh-word, an auxiliary, and a verb phrase
  - Who has drunk the tea?
- Object wh-questions contain a wh-word, an auxiliary, a noun phrase and a verb phrase
  - What does Natalie drink?

CFGs and dependency grammars for regular languages can be highly complex!

However, they facilitate automated syntactic and semantic parsing …two essential tools for NLP systems

Summary: Constituency Grammars

Constituency grammars describe a language's syntactic structure

Constituents, a core component of constituency grammars, are groups of words that function as a single unit

There are many ways to represent constituency grammars, but the most common way is by using **trees**

Constituency grammars can generate any sentences belonging to their language using (potentially recursive) combinations of **production rules**