Exercise today!

Context-Free Grammars

CS 485, Fall 2023
Applications of Natural Language Processing
https://people.cs.umass.edu/~brenocon/cs485_f23/

Brendan O'Connor
College of Information and Computer Sciences
University of Massachusetts Amherst

- Syntax: how do words structurally combine to form sentences and meaning?

2 representations

directed edges
b/w words

- Constituents
  - [the big dogs] chase cats
  - [colorless green clouds] chase cats

NP

- Dependencies
  - The **dog** ← **chased** the cat.
  - My **dog**, who's getting old, **chased** the cat.

- Idea of a *grammar (G)*: global template for how sentences / utterances / phrases ***w*** are formed, via latent syntactic structure ***y***
  - Linguistics: what do G and P(w,y | G) look like?
  - Generation: score with, or sample from, P(w, y | G)
  - Parsing: predict P(y | w, G)

input

Syntax for NLP

- If we could predict syntactic structure from raw text (*parsing*), that could help with...
  - Language understanding: meaning formed from structure
  - Grammar checking
  - Preprocessing: Extract phrases and semantic relationships between words for features, viewing, etc.
- Provides a connection between the theory of *generative linguistics* and computational modeling of language
- Accurate full sentence parsing is challenging!

Is language context-free?

- Regular language: repetition of repeated structures
  - e.g. "base noun phrases": (Noun | Adj)* Noun
    - subset of the JK pattern
- Context-free: hierarchical recursion
- Center-embedding: classic theoretical argument for CFG vs. regular languages
  - (10.1) The cat is fat.
  - (10.2) The cat that the dog chased is fat.
  - (10.3) *The cat that the dog is fat.
  - (10.4) The cat that the dog that the monkey kissed chased is fat.
  - (10.5) *The cat that the dog that the monkey chased is fat.

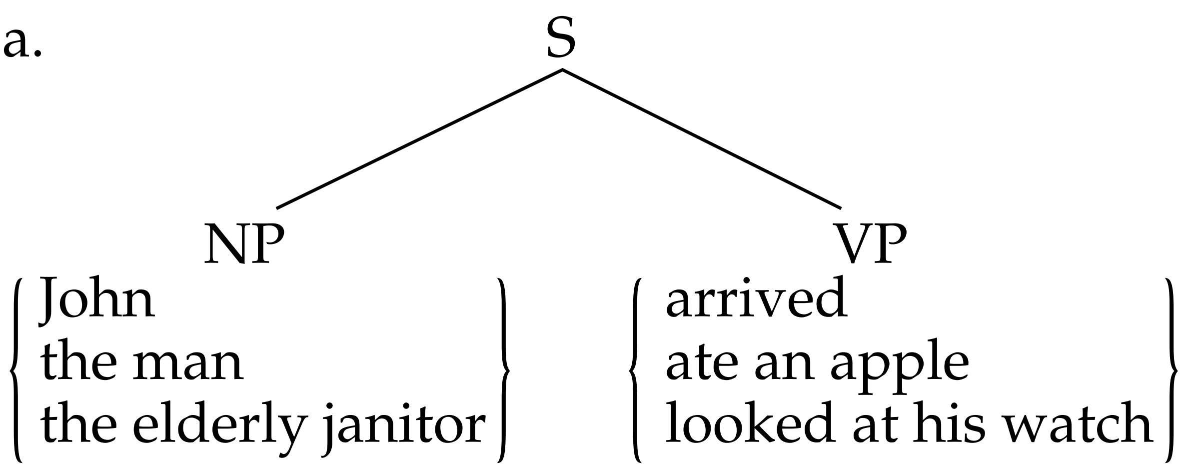
- Competence vs. Performance

[Examples from Eisenstein (2017)]

Hierarchical view of syntax

- "a Sentence made of Noun Phrase followed by a Verb Phrase"

a.



b. S $\rightarrow$ NP VP (1)

[From Phillips (2003)]

Context-free grammars (CFG)

- A CFG is a 4-tuple:

N	a set of non-terminals
Σ	a set of terminals (distinct from N)
R	a set of productions, each of the form $A \to \beta$, where $A \in N$ and $\beta \in (\Sigma \cup N)^*$
S	a designated start symbol

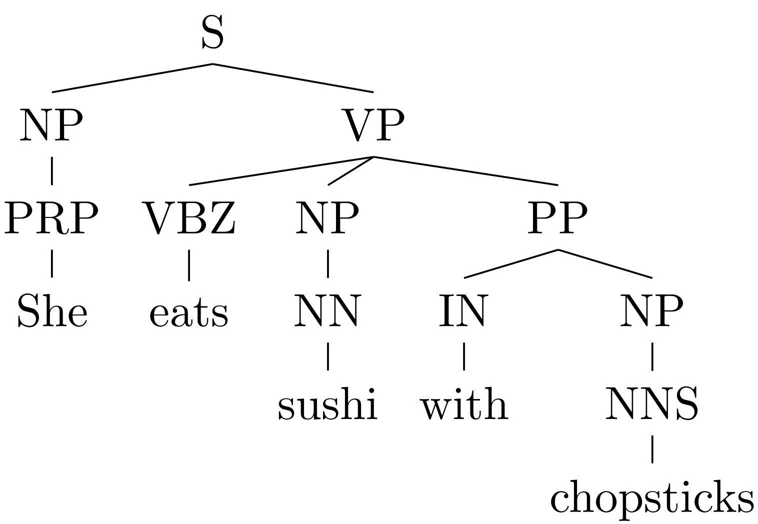
Example: see handout!

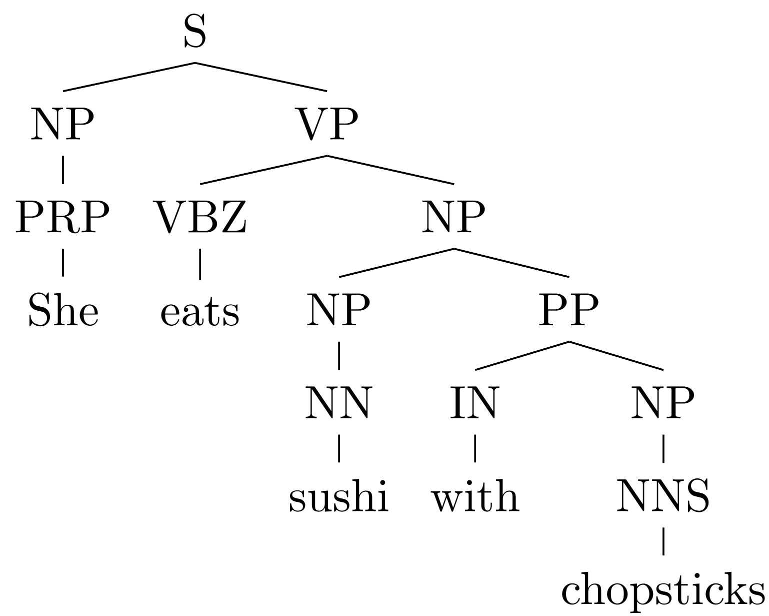
- Derivation: a sequence of rewrite steps from S to a string (sequence of terminals, i.e. words)
- Yield: the final string (sentence)
- The parse tree or constituency tree corresponds to the rewrite steps that were used to derive the string

- A CFG is a "boolean language model"
  - A grammar (4-tuple) defines to a set of strings it could generate

- Example: derivation from worksheet's grammar

Example





(S(NP(PRP *She*)(VP(VBZ *eats*)
(NP(NN *sushi*))
(PP(IN*with*)(NP(NNS *chopsticks*))))))

(S(NP(PRP *She*)(VP(VBZ *eats*)
(NP(NP(NN *sushi*))(PP (IN*with*)(NP(NNS *chopsticks*)))))))

- All useful grammars are *ambiguous*: multiple derivations with same yield
- [Parse tree representations: Nested parens *or* non-terminal spans]

[Examples from Eisenstein (2017)]

Constituents

- Constituent tree/parse is one representation of sentence's syntax. What should be considered a constituent, or constituents of the same category?
    - Movement tests
    - Substitution tests
    - Coordination tests

- Simple grammar of English
    - Must balance *overgeneration* versus *undergeneration*
    - Noun phrases
    - NP modification: adjectives, PPs
    - Verb phrases
    - Coordination
    - etc...
- Better coverage: machine-learned grammars, if you have a treebank (labeled dataset)

Is language context-free?

- CFGs nicely explain nesting and agreement (if you stuff grammatical features into the non-terminals)
    - *The **processor** __**has**__ 10 million times fewer transistors on it than todays typical micro-processors, __**runs**__ much more slowly, and __**operates**__ at five times the voltage...*
    - $$\begin{aligned} S &\rightarrow NN\ \ VP \\ VP &\rightarrow VP3S \mid VPN3S \mid \ldots \\ VP3S &\rightarrow VP3S, VP3S, \text{ and } VP3S \mid VBZ \mid VBZ\ NP \mid \ldots \end{aligned}$$

[Examples from Eisenstein (2017)]

[Examples from Eisenstein (2017)]

- Real sentences have massively ambiguous syntax!

Attachment ambiguity *we eat sushi with chopsticks, I shot an elephant in my pajamas.*

Modifier scope *southern food store*

Particle versus preposition *The puppy tore up the staircase.*

Complement structure *The tourists objected to the guide that they couldn't hear.*

Coordination scope *"I see," said the blind man, as he picked up the hammer and saw.*

Multiple gap constructions *The chicken is ready to eat*

Penn Treebank

```
( (S
    (NP-SBJ (NNP General) (NNP Electric) (NNP Co.) )
    (VP (VBD said)
      (SBAR (-NONE- 0)
        (S
          (NP-SBJ (PRP it) )
          (VP (VBD signed)
            (NP
              (NP (DT a) (NN contract) )
              (PP (-NONE- *ICH*-3) ))
            (PP (IN with)
              (NP
                (NP (DT the) (NNS developers) )
                (PP (IN of)
                  (NP (DT the) (NNP Ocean) (NNP State) (NNP Power) (NN project) ))))
            (PP-3 (IN for)
              (NP
                (NP (DT the) (JJ second) (NN phase) )
                (PP (IN of)
                  (NP
                    (NP (DT an) (JJ independent)
                      (ADJP
                        (QP ($ $) (CD 400) (CD million) )
                        (-NONE- *U*) )
                      (NN power) (NN plant) )
                    (, ,)
                    (SBAR
                      (WHNP-2 (WDT which) )
                      (S
                        (NP-SBJ-1 (-NONE- *T*-2) )
                        (VP (VBZ is)
                          (VP (VBG being)
                            (VP (VBN built)
                              (NP (-NONE- *-1) )
                              (PP-LOC (IN in)
                                (NP
                                  (NP (NNP Burrillville) )
                                  (, ,)
                                  (NP (NNP R.I) ))))))))))))))))
```

Context-Free Grammar

- CFG describes a generative process for an (infinite) set of strings
  - 1. Nonterminal symbols
    - "S": START symbol / "Sentence" symbol
  - 2. Terminal symbols: word vocabulary
  - 3. Rules (a.k.a. Productions). Practically, two types:

<u>"Grammar": *one* NT expands to >=1 NT</u>
always one NT on left side of rule

S	→	*NP VP*	I + want a morning flight
NP	→	*Pronoun*	I
	|	*Proper-Noun*	Los Angeles
	|	*Det Nominal*	a + flight
Nominal	→	*Nominal Noun*	morning + flight
	|	*Noun*	flights
VP	→	*Verb*	do
	|	*Verb NP*	want + a flight
	|	*Verb NP PP*	leave + Boston + in the morning
	|	*Verb PP*	leaving + on Thursday
PP	→	*Preposition NP*	from + Los Angeles

<u>Lexicon: NT expands to a terminal</u>

Noun	→	*flights* | *breeze* | *trip* | *morning* | ...
Verb	→	*is* | *prefer* | *like* | *need* | *want* | *fly*
Adjective	→	*cheapest* | *non* – *stop* | *first* | *latest* | *other* | *direct* | ...
Pronoun	→	*me* | *I* | *you* | *it* | ...
Proper-Noun	→	*Alaska* | *Baltimore* | *Los Angeles* | *Chicago* | *United* | *American* | ...
Determiner	→	*the* | *a* | *an* | *this* | *these* | *that* | ...
Preposition	→	*from* | *to* | *on* | *near* | ...
Conjunction	→	*and* | *or* | *but* | ...

Probabilistic CFGs

S → NP VP	[.80]	Det → that [.10] $\mid$ a [.30] $\mid$ the [.60]
S → Aux NP VP	[.15]	Noun → book [.10] $\mid$ flight [.30]
S → VP	[.05]	$\mid$ meal [.15] $\mid$ money [.05]
NP → Pronoun	[.35]	$\mid$ flights [.40] $\mid$ dinner [.10]
NP → Proper-Noun	[.30]	Verb → book [.30] $\mid$ include [.30]
NP → Det Nominal	[.20]	$\mid$ prefer; [.40]
NP → Nominal	[.15]	Pronoun → I [.40] $\mid$ she [.05]
Nominal → Noun	[.75]	$\mid$ me [.15] $\mid$ you [.40]
Nominal → Nominal Noun	[.20]	Proper-Noun → Houston [.60]
Nominal → Nominal PP	[.05]	$\mid$ TWA [.40]
VP → Verb	[.35]	Aux → does [.60] $\mid$ can [40]
VP → Verb NP	[.20]	Preposition → from [.30] $\mid$ to [.30]
VP → Verb NP PP	[.10]	$\mid$ on [.20] $\mid$ near [.15]
VP → Verb PP	[.15]	$\mid$ through [.05]
VP → Verb NP NP	[.05]	
VP → VP PP	[.15]	
PP → Preposition NP	[1.0]	

- Defines a probabilistic generative process for words in a sentence
  - Can parse with a modified form of CKY
- How to learn? Fully supervised with a treebank... unsupervised learning possible too, but doesn't give great results...

[J&M textbook]

PCFG as LM

Is a PCFG a *good* LM? Yes...

Is a PCFG a *good* LM? No...