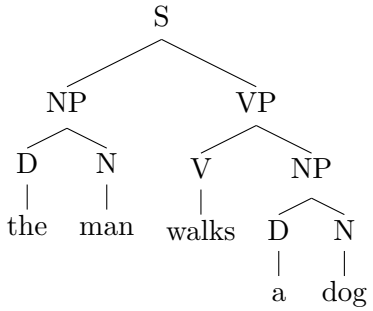
NLP: Parsing

Dan Garrette
dhg@cs.utexas.edu

November 2, 2013

1 Grammar



- "Consitiuency" parse
- S (sentence), NP (noun phrase), VP (verb phrase) are constituents
- Words combine to make phrases, and phrases combine to make larger phrases and sentences.

2 Context-Free Grammars

- Context-free grammars can be specified by a table of "productions"

S	→	NP VP		D	→	{*the, a*}
NP	→	D N	("a dog barks")	N	→	{*man, dog*}
NP	→	N	("dogs bark")	V	→	{*barks, walks, sees*}
VP	→	VP NP	(transitive verb)			
VP	→	VP	(intransitive verb)			

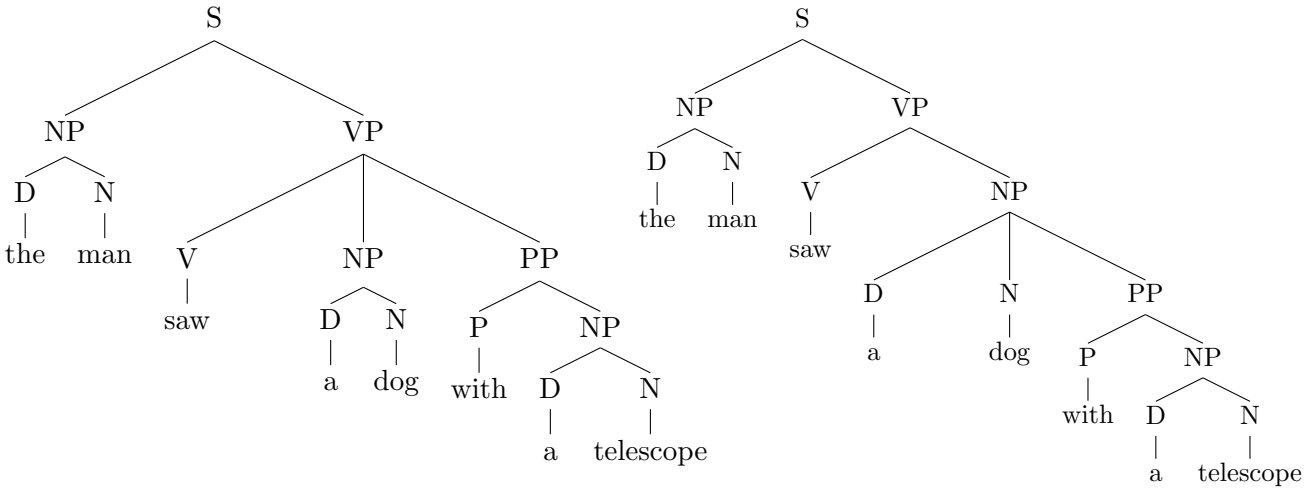
- Words are called "terminals" and other nodes are "non-terminals"

Syntactic Ambiguity

- For a given CFG, there can be multiple trees that describe the same sentence
- Add the following rules to the above:

NP	→	D N	("a dog barks")	N	→	{*telescope*}
NP	→	N	("dogs bark")			
VP	→	VP NP	(transitive verb)			
VP	→	VP	(intransitive verb)			

- "The man saw a dog with a telescope"



3 Generative Model

- Like naïve Bayes models, N-Gram models, and hidden Markov models
- Two probability distribution: $p(\beta \mid \alpha)$, for production rules $\alpha \rightarrow \beta$, and $p(\sigma)$, where σ is a possible "start" symbol
- Generative story:
  1. Choose a start symbol x from the distribution over start symbols $p(\sigma)$
  2. If x is a terminal, STOP
  3. Else, choose some β from $p(\beta \mid x)$
  4. For each symbol y in β, go to step 2
- For each node with symbol x, we choose a production rule of the form $x \rightarrow \beta$ according to their probabilities and then recursively choose rules for every node in β until we reach terminals for all branches.

4 Other Grammatical Formalisms

TAG: Tree-Adjoining Grammar: http://www.seas.upenn.edu/∼joshi/joshi-schabes-tag-97.pdf

CCG: Combinatory Categorial Grammar

Dependency Parsing