Context Free Grammars:
Phrases for English
Problems with CFGs

Key Constituents for English

- English has headed phrase structure
  - X-bar theory: in natural languages, phrases are headed by particular kinds of words with modifiers and qualifiers around them (specifiers, adjuncts, and complements)
- Verb Phrases VP → … VB* …
- Noun Phrases NP → … NN* …
- Adjective Phrases ADJP → … JJ* …
- Adverb Phrases ADVP → … RB* …
- Sentences (and clauses): SBAR → S | SINV | SQ …
  - Sentences, inverted sentences, direct questions, … can also appear in larger clause structure SBAR where sentence is preceded by *that*
- Plus minor phrase types:
  - QP (quantifier phrase in NP, PP (prepositional phrase), CONJP (multi word constructions: *as well as*), INTJ (interjections), etc.

e.g. Penn Treebank Constituent Tags:
http://www.surdeanu.info/mihai/teaching/ista555-fall13/readings/PennTreebankConstituents.html

Sentences

- Sentences
  - Declaratives: A plane left

    S -> NP VP
  - Imperatives: Leave!

    S -> VP
  - Yes-No Questions: Did the plane leave?

    S -> Aux NP VP
  - WH Questions: When did the plane leave?

    S -> WH Aux NP VP

Noun Phrases

- Noun phrases have a head noun with pre and post-modifiers
  - Determiners, Cardinals, Ordinals, Quantifiers and Adjective Phrases are all optional, indicated here with parentheses

    NP -> (DT) (Card) (Ord) (Quan) (AP) Noun
    Noun -> NN | NP | NPS | NNS   *(the four noun POS tags)*
  - Post-modifiers include prepositional phrases, gerundive phrases, and relative clauses

    the man [from Moscow]
    any flights [arriving after 11pm] (gerundive)
    the spy[who came in from the cold] (relative clause)

Some examples on these slides are from the Jurafsky and Martin text and from Jim Martin's online course materials.

Recursive Rules

- One type of Noun phrase is a Noun Phrase followed by a Prepositional phrase

  * NP -> NP PP

  PP -> Prep NP

- Of course, this is what makes syntax interesting

  flights from Denver

  flights from Denver to Miami

  flights from Denver to Miami in February

  flights from Denver to Miami in February on a Friday

  flights from Denver to Miami in February on a Friday under $300

  flights from Denver to Miami in February on a Friday under $300 with lunch

  – Syntax trees for these examples also need rules for NP -> Noun, etc.

* This grammar illustrates the recursion, but may not give the best derivation for these phrases!

Verb Phrases

- Simple Verb phrases

VP -> Verb	*leave*
| Verb NP	*leave Boston*
| Verb NP PP	*leave Boston in the morning*
| Verb PP	*leave in the morning*

- Verbs may also be followed by a clause

  VP -> Verb S

  I think I would like to take a 9:30 flight

- The phrase or clause following a verb is sometimes called the complementizer

Conjunctive Constructions

- S -> S and S
  - John went to NY and Mary followed him
- NP -> NP and NP
- VP -> VP and VP
- …
- In fact the right rule for English is

  X -> X and X

Problems

- Context-Free Grammars can represent many parts of natural language adequately
- Here are some of the problems that are difficult to represent in a CFG:
  - Agreement
  - Subcategorization
  - Movement (for want of a better term)

Agreement

• This dog	• *This dogs
• Those dogs	• *Those dog
• This dog eats	• *This dog eat
• Those dogs eat	• *Those dogs eats

- In English,
  - subjects and verbs have to agree in person and number
  - Determiners and nouns have to agree in number
- Many languages have agreement systems that are far more complex than this.
- Solution can be either to add rules with agreement or to have a layer on the grammar called the features

Subcategorization

- Subcategorization expresses the constraints that a particular verb (sometimes called the predicate) places on the number and syntactic types of arguments it wants to take (occur with).
  - Sneeze: John sneezed
  - Find: Please find [a flight to NY]$_{NP}$
  - Give: Give [me]$_{NP}$[a cheaper fare]$_{NP}$
  - Help: Can you help [me]$_{NP}$[with a flight]$_{PP}$
  - Prefer: I prefer [to leave earlier]$_{TO\text{-}VP}$
  - Told: I was told [United has a flight]$_{S}$
  - …

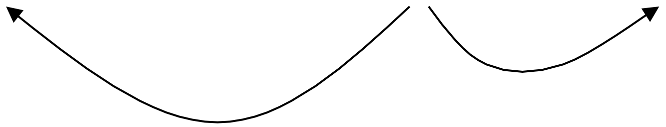
Subcategorization

- Should these be correct?
  - John sneezed the book
  - I prefer United has a flight
  - Give with a flight
- The various rules for VPs *overgenerate*.
  - They permit the presence of strings containing verbs and arguments that don't go together
  - For example VP -> V NP therefore

    Sneezed the book is a VP since "sneeze" is a verb and "the book" is a valid NP
- Now *overgeneration* is a problem for a generative approach.
  - The grammar should represent all and only the strings in a language
- From a practical point of view... Not so clear that there's a problem

Movement

- Consider the verb “booked” in the following example:
  - [[My travel agent]$_{NP}$ [booked [the flight]$_{NP}$]$_{VP}$]$_{S}$



- i.e. “book” is a straightforward transitive verb. It expects a single NP arg within the VP as an argument, and a single NP arg as the subject.

Movement

- But what about?
  - Which flight do you want me to have the travel agent book?
- The direct object argument to "book" isn't appearing in the right place. It is in fact a long way from where it's supposed to appear.
- And note that it's separated from its verb by 2 other verbs.
- In Penn Treebank, these types of movement are annotated by have an empty Trace constituent appear in the right place.

The Point about CFGs

- CFGs appear to be just about what we need to account for a lot of basic syntactic structure in English.
- But there are problems
  - that can be dealt with adequately, although not elegantly, by staying within the CFG framework.
- There are simpler, more elegant, solutions that take us out of the CFG framework (beyond its formal power)
  - For example, Feature Structures for CFGs place constraints on how the rules can be applied