CMPT-825
Natural Language Processing

Anoop Sarkar
http://www.cs.sfu.ca/~anoop

October 26, 2010

Parts of Speech

- We have seen that individual words can be classified into groups or classes that we call **parts of speech**
  - Determiners: *a, the*
  - Verbs: *arrive, attracts, love, sit*
  - Prepositions: *of, by, in, outside, on*
  - Nouns: *he, she, it, San, Diego*
- But these individual words can group together to form larger groups which possess meaning when put together, e.g. *San Diego*, *the man outside the building*

Constituents

- Let's consider the grouping of words into **noun phrases**
  - *three parties from Brooklyn*
  - *a high class spot such as Mindy's*
  - *they*
  - *Harry the Horse*
  - *the fact that he came into the Hot Box*
  - *swimming on a hot day*

Constituents

- These *noun phrases* are selected by verbs as a whole unit:
  - three parties from Brooklyn *arrived* ...
  - $*$ three from *arrived* ...
  - a high class spot such as Mindy's *attracts* ...
  - they *sit* ...
  - they *like* swimming on a hot day

Testing for constituents

- Things that can be moved around together: *preposed* or *postposed* elements in a sentence.
  - *On Sept 17th, I'd like to fly to Toronto*
  - *I'd like to fly, On Sept 17th, to Toronto*
  - *I'd like to fly to Toronto On Sept 17th*
  - $*$ *On I'd like to fly Sept to Toronto 17th*

Testing for constituents

- Things that can be questioned:
  - Who came to the negotiating table?
    three parties from Brooklyn
  - Where would a high roller like Deckard go?
    a high class spot such as Mindy's
  - What is it that Mary would like to do when she visits?
    swimming on a hot day

Testing for constituents

- Things that can be referred to with a pronoun:
  - three parties from Brooklyn arrived
    they were late
  - a high class spot such as Mindy's is where Deckard would go
    But it is closed today
  - swimming on a hot day is what Mary would like to do
    Even though it is bad for health

Testing for constituents

- Things that can be coordinated:
  - *John and Mary*
  - *the barrier islands and frogs that provide hallucinations when you lick them*
  - *swimming on a hot day and taking a long skiing lesson*

Testing for constituents

- Movement is stricter than coordination:
  - John bought the large cup and small picture
  - the large cup, John bought
  - $*$ large cup, John bought the

Things that are not constituents

- Who does John think stole the cookies?
  Ans: $*$ John thinks Mary
- *But*: John thinks Mary and Bill thinks Frida stole the cookies
- John bought the photo of a clown.
  Q: What was done to the photo of a clown?
  A: $*$ John bought
- But: John bought and Bill installed the photo of a clown.
- $*$ What did John buy and Peter bought chocolates.
- John thinks Mary and John bought the tickets.
- John thinks Mary and John bought the tickets.

Chunking Noun Phrases: Not as easy as it seems

- Finding noun phrases can be treated as finding a sequence of words that is a noun phrase (the **chunking** approach). Finding chunks is not trivial:
  - *(NNP San) (NNP Diego)*
  - *(NNPS Wednesdays)*
  - *(DT the) (NN company) (POS 's) (VBN refocused) (NN direction)*
  - *(DT the) (NN government) (VBZ 's) (VBG dawdling)*
  - $*$ *(DT The) (NNP Dow) (NNP Jones) (VBZ is) (VBG swimming) (IN in) (NN tech) (NNS stocks)*

Recursion in Regular Languages

- Consider a regular expression for arithmetic expressions:
  $2 + 3 * 4$
  $8 * 10 + 24$
  $2 + 3 * 2 + 8 + 10$

  `digit (+|* digit)*`
- *Can we compute the* meaning *of these expressions?*

Recursion in Regular Languages

- Construct the finite state automata and associate the meaning with the state sequence
- However, this solution is missing something crucial about arithmetic expressions – *what is it?*

Recursion in Regular Languages

- Going back to noun phrases (NP, for short): let's attempt to provide a regular expression grammar for a subset of all the possible noun phrases
- Consider the noun phrases: *the man in the park*, *the person with the big head in the park*, *the unicorn in the garden inside the dream with a strange mark on the head*, ...
- These are simple noun phrases that have prepositional phrases (PP, for short) modifying nouns. PPs are another example of a constituent, but now we need to combine them with NPs

Recursion in Regular Languages

- Consider the noun phrases: *the man in the park*, *the person with the big head in the park*, *the unicorn in the garden inside the dream with a strange mark on the head*, ...
- (NP) $(PP)^*$ $\rightarrow$ (Det N) $(PP)*$ $\rightarrow$ (Det N) $(P\ NP)^*$
- (Det N) (P (Det N)) PP^* $\rightarrow$ (Det N) (P (Det N))*
- So, it's possible, but it gets ugly fast, let's widen our view of what can occur inside NPs.

Recursion in Regular Languages

- Let's call (Det N) a **basal** NP and now consider that (Det N) is not the only base NP that is possible: (N) or (A N) or (A^+ N) or even:
  (D A^* N POS N) *the short man 's dream* ...
- So this means that we can now have (P (N)) or (P (A N)) or (P (A^+ N)) or ...
- Each former type of NP can be modified by each latter type of PP
- What is the only way to rescue the regular expression approach?
  combinatorial explosion of combinations

Context-Free Grammars

- A CFG is a 4-tuple: (N, T, P, S), where
  - N is a set of non-terminal symbols,
  - T is a set of terminal symbols which can include the empty string ϵ. T is analogous to Σ the alphabet in FSAs.
  - P is a set of rules of the form $A \rightarrow \alpha$, where $A \in N$ and $\alpha \in \{N \cup T\}^*$
  - S is a set of start symbols, $S \in N$

Context-Free Grammars

- Here's an example of a CFG, let's call this one G:
  1. $S \rightarrow a\ S\ b$
  2. $S \rightarrow \epsilon$
- What is the language of this grammar, which we will call $L(G)$, the set of strings *generated* by this grammar How? Notice that there cannot be any FSA that corresponds exactly to this set of strings $L(G)$ Why?
- What is the *tree set* or derivations produced by this grammar?

Context-Free Grammars

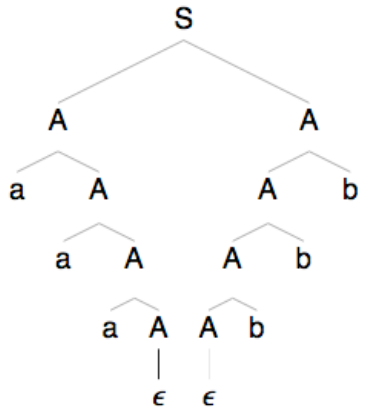
- This notion of generating both the strings and the trees is an important one for Computational Linguistics
- Consider the trees for the grammar G':

$$\begin{aligned} P = \{ \quad & S \rightarrow A\ A \\ & A \rightarrow aA \\ & A \rightarrow A\ b \\ & A \rightarrow \epsilon \\ \} & \end{aligned}$$

  $\Sigma = \{a, b\}, N = \{S, A\}, T = \{a, b, \epsilon\}, S = \{S\}$
- Why is it called *context-free* grammar?

Context-Free Grammars

- Can the grammar G' produce only trees of the kind shown below?



Context-Free Grammars

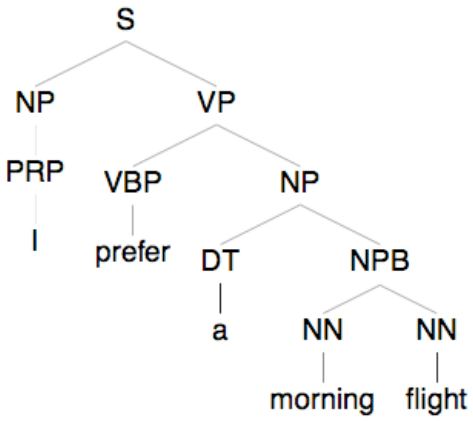
- We will come back to this issue when we try to figure out whether human languages are more powerful than CFLs.
- The distinction between strings and the trees (or any kind of structural description) is called *weak* vs. *strong* generative capacity.

Parse Trees

Consider the grammar with rules:

$$
\begin{aligned}
S &\rightarrow NP\ VP \\
NP &\rightarrow PRP \\
NP &\rightarrow DT\ NPB \\
VP &\rightarrow VBP\ NP \\
NPB &\rightarrow NN\ NN \\
PRP &\rightarrow I \\
VBP &\rightarrow prefer \\
DT &\rightarrow a \\
NN &\rightarrow morning \\
NN &\rightarrow flight
\end{aligned}
$$

Parse Trees



Parse Trees: Equivalent Representations

- (S (NP (PRP I)) (VP (VBP prefer) (NP (DT a) (NPB (NN morning) (NN flight)))))
- [$_S$ [$_{NP}$ [$_{PRP}$ I]] [$_{VP}$ [$_{VBP}$ prefer] [$_{NP}$ [$_{DT}$ a] [$_{NPB}$ [$_{NN}$ morning] [$_{NN}$ flight]]]]]