Natural Language Processing
CMSC 723 (spring, 2001)

March 28, 2001

- Syntax
- Context Free Grammars
- Chomsky Hierarchy
- Grammar Development
- Grammar Equivalance and Normal Form
- Intro to Parsing
- HOMEWORK #6:
  (1) Jurafksy and Martin, 9.6
  (2) Jurafksy and Martin, 9.7
  (3) Convert this grammar to CNF:
  $S \rightarrow aYX, \quad X \rightarrow aX \mid b, \quad Y \rightarrow Ya \mid b$

What is Syntax?

- How words and phrases are strung together
- Childrens' language
- Not meaning

Syntax

Who cares?

- Grammar checkers
- Question answering/database access
- Informatation extraction
- Generation
- Translation

Context Free Grammars

- Captures constituency and ordering
- Need something else!
- Used in modern linguistic theories of grammar?

Context Free Grammars

Consist of:

- Sets of terminals.
- Sets of non-terminals
- Sets of rules of the form $A \to \alpha$ where α is a string of zero or more terminals and non-terminals.
- Notation: $X \to aX \mid b$
- The use of "context" in CFG?
- Backus-Naur form grammars.

Sample Grammar

[Figure 9.3]

Derivations and Trees

[Figure 9.4]

As opposed to what?

- Regular expressions
- Context sensitive grammars
- Turing machines

Chomsky Hierarchy

[Figure 13.1]

Rule Skeletons for Chomsky Hierarchy

#	Common Name	Rule Skeleton	Linguistic Example
0	Turing Equivalent	$\alpha \rightarrow \beta$ s.t. $\alpha \neq \epsilon$	ATNs
1	Context Sensitive	$\alpha A \beta \rightarrow \alpha \gamma \beta$ s.t. $\gamma \neq \epsilon$	Tree-Adjoining Grammars
2	Context Free	$\alpha \rightarrow \gamma$	Phrase-Structure Grammars
3	Regular	$\alpha \rightarrow x\beta$ or $\alpha \rightarrow x$	Finite State Automata

Note!

The use of the term "context free" in the description of this formalism has nothing to do with the ordinary use of the word context.

Grammar Development

S → NP VP
NP → Det Nominal
NP → Pronoun
Nominal → Noun Nominal
Nominal → Noun
VP → V
VP → V NP
Pronoun → I
Det → a
Noun → morning
Noun → flight
V → want | fly

Key Constituents

- Sentences
- Noun phrases
- Verb phrases
- Prepositional phrases

Sentence Types

- Declaratives: John left.
- Imperatives: Leave!
- Yes-No Questions: Did John leave?
- WH Questions: When did John leave?

Recursive Structures

NP $\rightarrow$ NP PP The flight to Boston

VP $\rightarrow$ VP PP departed Miami at noon

Recursive Structures

Flights to Miami

Flights to Miami from Boston

Flights to Miami from Boston in April

Flights to Miami from Boston in April on Friday

Flights to Miami from Boston in April on Friday under $300.

Flights to Miami from Boston in April on Friday under $300 with lunch.

Conjunctions

S → S and S

NP → NP and NP

VP → VP and VP

Some Difficulties

- Agreement
- Subcategorization
- Movement

Agreement

This dog
Those dogs
∗Those dog
∗This dogs

Do [any flights] stop in Chicago?
Does [NP Delta] stop in Chicago?

How do we deal with agreement?

$\text{Sg}_{\text{Nominal}} \rightarrow \text{Sg}_{\text{Noun}} \mid \text{Sg}_{\text{Noun}}\ \text{Sg}_{\text{Noun}}$

$\text{Pl}_{\text{Nominal}} \rightarrow \text{Pl}_{\text{Noun}} \mid \text{Pl}_{\text{Noun}}\ \text{Pl}_{\text{Noun}}$

Problems?

Subcat

VP → Verb disappear
VP → Verb NP prefer a morning flight
VP → Verb NP PP leave Boston in the morning
VP → Verb PP leaving on Thursday

*I disappeared the cat.

Subcat

Frame	Verb	Example
$\emptyset$	eat, sleep	I want to eat
NP	prefer, find, leave,	Find [NP the flight from Pittsburgh to Boston]
NP NP	show, give	Show [NP me] [NP airlines with flights from Pittsburgh]
PP_{from} PP_{to}	fly, travel	I would like to fly [pp from Boston] [pp to Philadelphia]
NP PP_{with}	help, load	Can you help [NP me] [pp with a flight]
VP_{to}	prefer, want, need	I would prefer [VP to to go by United airlines]
VP_{brst}	can, would, might	I can [VP_{brst} go from Boston]
S	mean	Does this mean [S AA has a hub in Boston]?

Movement

- I looked up his grade.
  I looked his grade up.
- John put the book on the table.
  What did John put on the table?
- Long distance dependencies.

Grammar Equivalence and Normal Form

Weak equivalence:

Strong equivalence:

It is sometimes useful to have a normal form for grammars.

Chomsky Normal Form (CNF): A→BC or A→a

Any grammar can be converted into a weakly-equivalent CNF grammar.

Parsing

Parsing with a CFG is the task of assigning a correct tree (or derivation) to a string given some grammar.

Parsing as Search

As with finite-state recognition and transduction, parsing can be viewed as a search

A Note on the Input

Assumptions:

- The input is not tagged
- The input consists of unanalyzed word tokens
- All the words in the input are known
- All the words in the input are available simultaneuously

Example Context-Free Grammar and Example sentence

[Figure 10.2]

Example Context-Free Grammar and Example sentence

[Figure 10.1]