Natural Language Processing

Lecture 10

Today's Outline

Review

Syntax

Context Free Grammars

Derivations and Trees

Formal Language View

Grammar Development

Some Difficulties

Review

Parts of Speech and Tagging

- parts of speech represent classes of words based on morphological and distributional similarities
- accurate part of speech tagging is pretty much a solved problem (at least for English) by any one of a number of statistical or rule-based approaches
- for situations where there is *no* tagged training data one can fall back on unsupervised approaches

Syntax

By syntax, we mean various aspects of how words are strung together to form components of sentences and how those components are strung together to form sentences.

New concept: Constituency

- groups of words may behave as a single unit or constituent
- e.g., noun phrases
- evidence
  - the *whole* group appears in similar syntactic environments
  - e.g., before a verb
  - preposed/postposed constructions

Note also that notions of meaning play no role in what we're talking about (at least not yet).

Who Cares?

Grammar checkers

Information extraction

Generation

Context Free Grammars

Captures constituency and ordering.

We'll need something else for other aspects of syntax.

Modern linguistic theories of grammar are only vaguely based on context-free grammars.

Context Free Grammars (CFGs)

Consist of

- sets of terminals (either lexical items or parts of speech)
- sets of non-terminals (the constituents of the language)
- sets of *rules* (*productions*) of the form $A \rightarrow \alpha$
  - α is a string of one or more terminals and non-terminals
  - A is a non-terminal (POS for the lexicon entries)

They are exactly equivalent to Backus-Naur Form (BNF).

Also called Phrase-Structure Grammar.

Examples

S $\rightarrow$ NP VP

NP $\rightarrow$ Det Nominal

Nominal $\rightarrow$ Noun

VP $\rightarrow$ V

Det $\rightarrow$ the

Det $\rightarrow$ a

Noun $\rightarrow$ flight

V $\rightarrow$ left

Generativity

Just as we did with FSAs, you can view these rules as either structure imposing devices or generative devices.

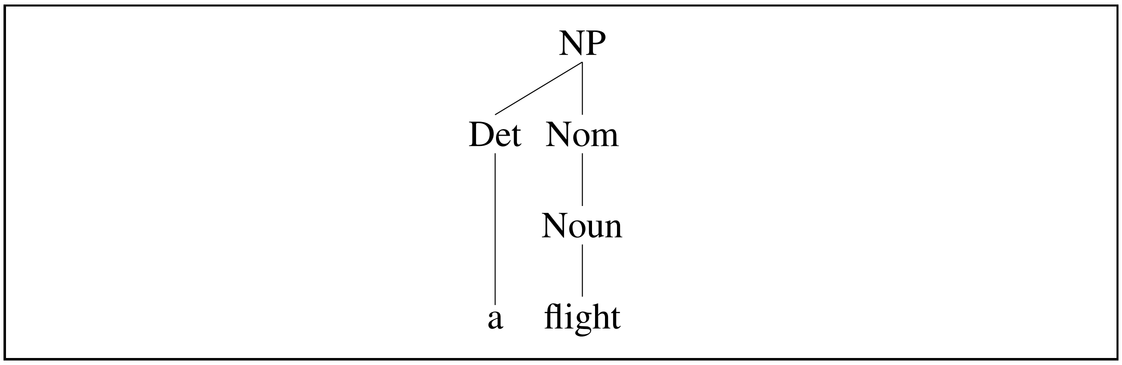
The grammar can be viewed as a formal device that specifies the strings in and not in a language.

Derivations and Trees

A <u>derivation</u> of a word string is just a sequence of rule applications.

- e.g., the string *a flight* can be derived from the non-terminal NP

Derivations can be visualized as <u>parse trees</u>.



There is also a bracketed notation.

The formal language defined by a CFG is the set of strings that are derivable from the *start symbol*, e.g., from S, the set of sentences in English.

As opposed to what?

Regular expressions (too weak).

Context sensitive grammars (too strong).

Turing machines (way too strong).

Approaches that are too strong have the power to predict/describe/capture syntactic structures that don't exist in human languages.

And the computational processes associated with them are not as efficient as those associated with weaker methods.

A More Formal View

A CFG is a 4-tuple, and defines a formal language (a set of strings)

- a set of non-terminal symbols N
- a set of terminal symbols Σ
- a set of productions P
  - $A \rightarrow \alpha$
  - A is a non-terminal
  - α is a string of symbols from the infinite set of strings $(\Sigma \cup N)\star$
- a designated start symbol S

A More Formal View (cont.)

A language is defined via Derivation.

Sentences that can be derived by a grammar are in the formal language defined by the grammar, and are called Grammatical Sentences.

Those sentences that cannot be derived are Ungrammatical Sentences.

The language L_G generated by a grammar G is the set of strings composed of terminal symbols which can be derived from the start symbol.

- $L_G = W \mid w$ is in $\Sigma\star$ and S derives w

Parsing is mapping from a word string to its parse tree.

- see Chapter 10

Equivalence and Normal Form

Strong Equivalence

- two grammars generate the same set of strings and assign the same phrase structures

Weak Equivalence

- two grammars generate the same set of strings but do *not* assign the same phrase structures

Chomsky Normal Form (CNF)

- productions of the form $A \rightarrow BC$ or $A \rightarrow a$
- upper case indicates non-terminals, lowercase indicates terminals

Original

- A $\rightarrow$ B C D

CNF

- A $\rightarrow$ B X
- X $\rightarrow$ C D

What type of equivalence?

Note!

All the use of the term context-free really means is that the non-terminal on the left-hand side of the rule is sitting over there all by itself.

A $\rightarrow$ B C

In other words, I can rewrite A as BC, regardless of the context in which I find the A.

Developing Grammars

An awful lot goes into the development of grammars for human languages. We'll only scratch the surface.

The primary thing we'll consider is what constitutes a constituent. A consistent, recurring, syntactic substructure.

A Tiny Lexicon

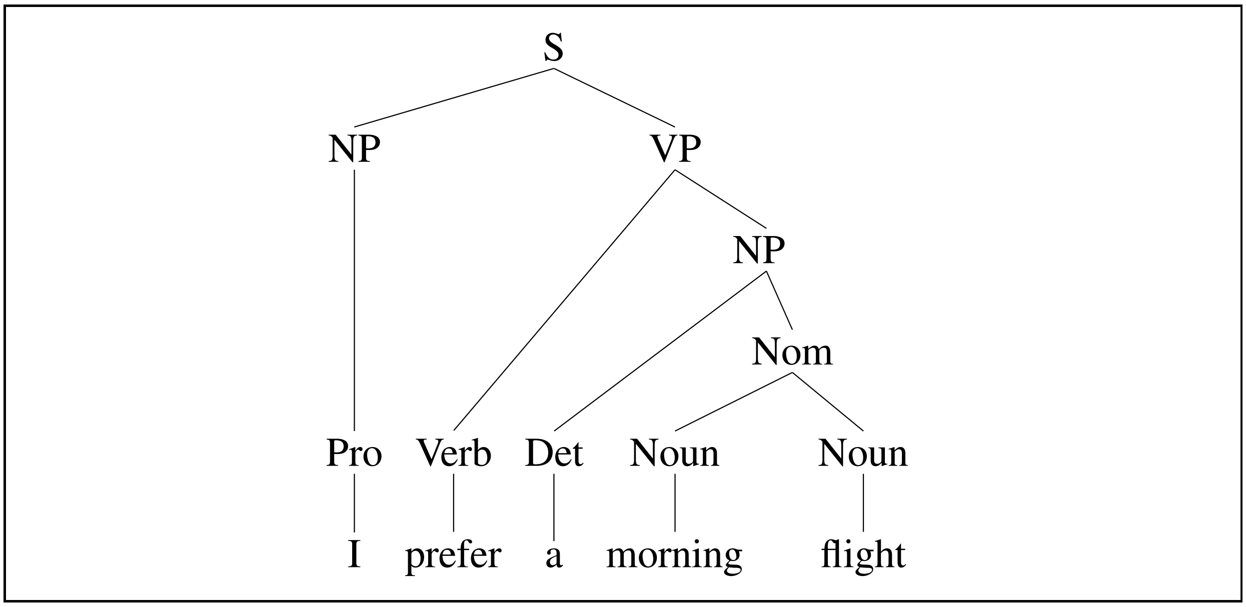
$$
\begin{aligned}
\textit{Noun} &\rightarrow flights \mid \textit{breeze} \mid trip \mid \textit{morning} \mid \ldots \\
\textit{Verb} &\rightarrow is \mid prefer \mid \textit{like} \mid \textit{need} \mid \textit{want} \mid fly \\
\textit{Adjective} &\rightarrow \textit{cheapest} \mid non-stop \mid first \mid \textit{latest} \\
&\quad \mid \textit{other} \mid \textit{direct} \mid \ldots \\
\textit{Pronoun} &\rightarrow \textit{me} \mid \textit{I} \mid \textit{you} \mid \textit{it} \mid \ldots \\
\textit{Proper-Noun} &\rightarrow \textit{Alaska} \mid \textit{Baltimore} \mid \textit{Los Angeles} \\
&\quad \mid \textit{Chicago} \mid \textit{United} \mid \textit{American} \mid \ldots \\
\textit{Determiner} &\rightarrow \textit{the} \mid \textit{a} \mid \textit{an} \mid \textit{this} \mid \textit{these} \mid \textit{that} \mid \ldots \\
\textit{Preposition} &\rightarrow from \mid to \mid \textit{on} \mid \textit{near} \mid \ldots \\
\textit{Conjunction} &\rightarrow \textit{and} \mid \textit{or} \mid \textit{but} \mid \ldots
\end{aligned}
$$

A Tiny Grammar (and examples)

S	→	*NP VP*	I + want a morning flight
NP	→	*Pronoun*	I
	|	*Proper-Noun*	Los Angeles
	|	*Det Nominal*	a + flight
Nominal	→	*Noun Nominal*	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

- "or" symbol indicates alternate expansions for a non-terminal

A Parse Tree



Key Constituents

Sentences

Noun phrases

Verb phrases

Prepositional phrases

Common Sentence Structures

Declaratives

- John left.
- S $\rightarrow$ NP VP

Imperatives

- Leave!
- S $\rightarrow$ VP

Yes-No Questions

- Did John leave?
- S $\rightarrow$ Aux NP VP

WH Questions (who, where, what, which, why, how)

- When did John leave?
- S $\rightarrow$ Wh-NP Aux NP VP
- S $\rightarrow$ Wh-NP VP

Noun Phrases (simplified)

Revolves around a Head (the central noun)

Prenominal modifiers come *before* the head noun

- NP $\rightarrow$ (Det) (Card) (Ord) (Quant) (AP) Nominal

Determiners (Det)

- *a* flight
- optional (plurals: *the* flights, $\star$*a* flights; mass nouns...)

Cardinal Numbers (Card)

- *one* stop

Ordinal Numbers (Ord)

- the *first* stop

Quantifiers (Quant)

- *many* fares

Adjective Phrases (AP)

- a *nonstop* flight

Noun Phrases (continued)

<u>Postnominal</u> modifiers come *after* the head noun

- Nominal → Nominal PP (PP) (PP)
- Nominal → Nominal GerundVP
- Nominal → Nominal RelClause

Prepositional Phrase (PP)

- the flights *from Pittsburgh* `to Denver`

Non-finite clauses

- Gerundive (GerundVP): flights *leaving on Thursday*
- infinitive: the last flight *to arrive*
- -ed form: the aircraft *used*

Relative Clauses (RelClause)

- RelClause → (RelPronoun) VP
- flights *that serve lunch*

Postnominal modifiers can be combined

- a flights *from Pittsburgh* **leaving Monday**

Recursive Structures

One of the more interesting patterns we'll have to deal with involves recursive rules - rules where the non-terminal on the left-hand side also appears on the right-hand side.

Direct recursion:

- NP $\rightarrow$ NP PP (the flight to Boston)
- VP $\rightarrow$ VP PP (departed Miami at noon)

In general, recursion cannot be handled using FSAs, which is why we need the more powerful CFG representation.

However, there are FSA *approximations* of English syntax.

Recursive Structures (continued)

Note this is what allows us to do the following:

- 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 with lunch.

Conjunctions

S $\rightarrow$ S and S

NP $\rightarrow$ NP and NP

VP $\rightarrow$ VP and VP

In fact, any phrasal constituent can be *conjoined* with a constituent of the same type to form a new constituent of that type. In other words, English really has the following rule:

X $\rightarrow$ X and X

Some Difficulties

Agreement

Subcategorization

Agreement

Examples

- This dog
- Those dogs
- *Those dog
- *This dogs
- What flight leaves in the morning
- *What flight leave in the morning

Could expand our grammar with multiple sets of rules, but doubles the size.

Chapter 11 will present a better approach.

Subcategorization

Verbs have preferences for the kinds of constituents they co-occur with.

Examples:

- I found the cat
- *I disappeared the cat

A traditional subcategorization of verbs:

- transitive (takes a direct object NP)
- intransitive

Today, there are often a hundred subcategories for the *complements* of a verb.

- e.g., *find* subcategorizes for an NP (can take an NP complement)

The possible sets of complements are called the *Subcategorization Frame* for the verb.

Like with agreement, the obvious CFG solution yields rule explosion. Again, Chapter 11 will introduce a better approach.

Subcategorization (continued)

See Table on Page 343 in Jurafsky and Martin.

Grammars for Spoken Language

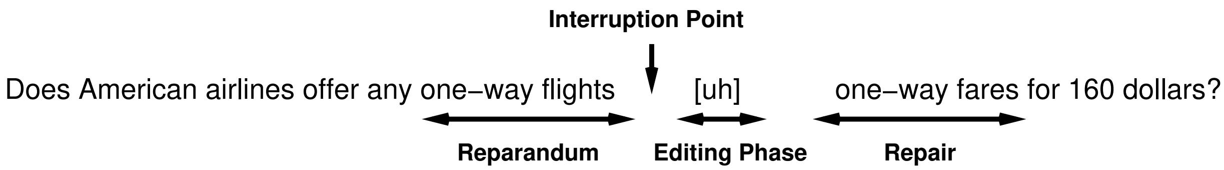
Spoken language is both similar, yet different, from written language.

the . [exhale] . . . [inhale] . . [uh] does American airlines . offer any . one way flights . [uh] one way fares, for one hundred and sixty one dollars
[mm] i'd like to leave i guess between [um] . [smack] . five o'clock no, five o'clock and [uh], seven o'clock . P M
around, four, P M
all right, [throat_clear] . . i'd like to know the . give me the flight . times . in the morning . for September twentieth . nineteen ninety one
[uh] one way
[uh] seven fifteen, please
on United airlines . . give me, the . . time . . from New York . [smack] . to Boise-, to . I'm sorry . on United airlines . [uh] give me the flight, numbers, the flight times from . [uh] Boston . to Dallas

Differences include...

- lexical statistics (e.g., spoken language has more pronouns)
- disfluencies (e.g., *uh*, *um*, word repetitions, false starts)
- fragments

Disfluencies in Spoken Language



- false start
- *uh*

Homework

See Homework 2.

For Next Time

Read Chapter 10

Acknowledgements

Some of this material was taken from the course web pages of James Martin.