Parsing

(These slides are modified from Dan Jurafsky's slides.)

Today

- Formal Grammars
  - Context-free grammar
  - Grammars for English
  - Treebanks
  - Dependency grammars

2

Syntax

- By grammar, or syntax, we have in mind the kind of implicit knowledge of your native language that you had mastered by the time you were 3 years old without explicit instruction
- Not the kind of stuff you were later taught in "grammar" school

3

Syntax

- Why should you care?
- Grammars (and parsing) are key components in many applications
  - Grammar checkers
  - Dialogue management
  - Question answering
  - Information extraction
  - Machine translation

4

Syntax

- Key notions that we'll cover
  - Constituency
  - Grammatical relations and Dependency
    - Heads
- Key formalism
  - Context-free grammars
- Resources
  - Treebanks

5

Constituency

- A sequence of words that acts as a single unit
  - Noun phrases
  - Verb phrases
- These units form coherent classes that behave in similar ways
  - For example, we can say that noun phrases can come before verbs

6

Constituency

- For example, following are all ***noun phrases*** in English...

Harry the Horse	a high-class spot such as Mindy's
the Broadway coppers	the reason he comes into the Hot Box
they	three parties from Brooklyn

7

Context-Free Grammars

- Context-free grammars (CFGs)
  - Also known as
    - Phrase structure grammars
    - Backus-Naur form
- Consist of
  - Rules
  - Terminals
  - Non-terminals

8

Context-Free Grammars

- Terminals
  - words
- Non-Terminals
  - The constituents in a language
    - Such as noun phrases, verb phrases and sentences
- Rules
  - Rules are equations that consist of a single non-terminal on the left and any number of terminals and non-terminals on the right.

9

Some NP Rules

- Here are some rules for our noun phrases

$$NP \rightarrow Det\ Nominal$$
$$NP \rightarrow ProperNoun$$
$$Nominal \rightarrow Noun \mid Nominal\ Noun$$

- Together, these describe two kinds of NPs.
  - One that consists of a determiner followed by a nominal
  - And another that says that proper names are NPs.
  - The third rule illustrates two things:
    - An explicit disjunction
    - A recursive definition

10

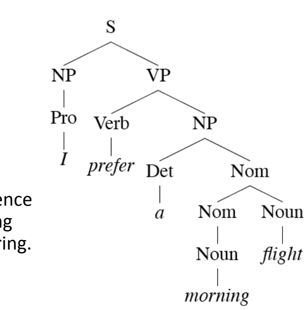
L0 Grammar

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

11

Derivations

A "***derivation***" is a sequence of rules applied to a string that *accounts* for that string.



12

Definition

- More formally, a CFG consists of

N a set of **non-terminal symbols** (or **variables**)
Σ a set of **terminal symbols** (disjoint from N)
R a set of **rules** or productions, each of the form $A \rightarrow \beta$,
where A is a non-terminal,
β is a string of symbols from the infinite set of strings $(\Sigma \cup N)*$
S a designated **start symbol**

13

Parsing

- Parsing is the process of taking a string and a grammar and returning a (or multiple) parse tree(s) for that string
- It is completely analogous to running a finite-state transducer with a tape
  - It's just more powerful ➔ there are languages we can capture with CFGs that we can't capture with finite-state machines.

14

An English Grammar Fragment

- Sentences
- Noun phrases
  - Agreement
- Verb phrases
  - Subcategorization

15

Sentence Types

- Declaratives: *A plane left.*

  S → NP VP
- Imperatives: *Leave!*

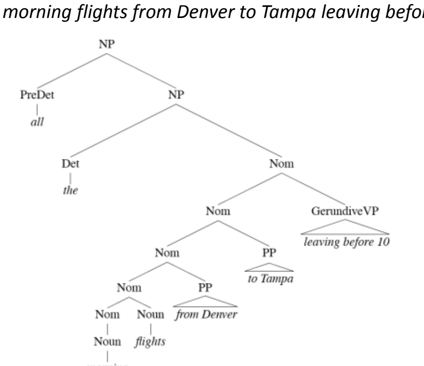
  S → VP
- Yes-No Questions: *Did the plane leave?*

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

  S → WH-NP Aux NP VP

16

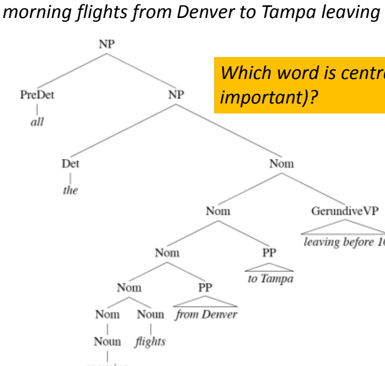
All the morning flights from Denver to Tampa leaving before 10



17

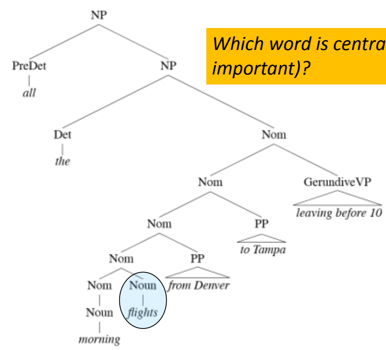
All the morning flights from Denver to Tampa leaving before 10

Which word is central (most important)?



18

All the morning flights from Denver to Tampa leaving before 10



Which word is central (most important)?

19

NP Structure

➔ *All the morning flights from Denver to Tampa leaving before 10*

- Clearly this NP is really about *flights*. That's the central critical noun in this NP. Such word is called as the ***head***.
- We can dissect this kind of NP into the stuff that can come before the head, and the stuff that can come after it.

20

Determiners

- Noun phrases can start with determiners...
- Determiners can be
  - Simple lexical items: *the, this, a, an*, etc.
    - A car
  - Or simple possessives
    - John's car
  - Or complex recursive versions of that
    - John's sister's husband's son's car

21

Nominals

- Contains the head and any pre- and post- modifiers of the head.
  - Pre-
    - Quantifiers, cardinals, ordinals...
      - Three cars
    - Adjectives and Aps
      - large cars
    - Ordering constraints
      - Three large cars
      - ?large three cars

22

Postmodifiers

- Three kinds
  - **Prepositional phrases**
    - Flights ***from Seattle***
  - **Non-finite clauses**
    - Flights ***arriving before noon***
  - **Relative clauses**
    - Flights ***that serve breakfast***
- Same general (recursive) rule to handle these
  - *Nominal → Nominal PP*
  - *Nominal → Nominal GerundVP*
  - *Nominal → Nominal RelClause*

23

Agreement

- Constraints that hold among various constituents.
- For example, in English, determiners and the head nouns in NPs have to agree in their number.
- Which of the following cannot be parsed by the rule ***NP → Det Nominal*** ?

(O) This flight	(X) This flights
(O) Those flights	(X) Those flight

24

Agreement

- Constraints that hold among various constituents.
- For example, in English, determiners and the head nouns in NPs have to agree in their number.

- Which of the following cannot be parsed by the rule

NP → Det Nominal **?**

➔ This rule does not handle agreement! (The rule does not detect whether the agreement is correct or not.)

(O) This flight	(X) This flights
(O) Those flights	(X) Those flight

25

Problem

- Our earlier NP rules are clearly deficient since they don't capture the agreement constraint
  - *NP → Det Nominal*
    - Accepts, and assigns correct structures, to grammatical examples (*this flight*)
    - But its also happy with incorrect examples (*these flight)
  - Such a rule is said to ***overgenerate***.
  - We'll come back to this in a bit

26

Verb Phrases

- English *VP*s consist of a head verb along with 0 or more following constituents which we'll call ***arguments***.

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

27

Subcategorization

- But, even though there are many valid VP rules in English, not all verbs are allowed to participate in all those VP rules.
- We can subcategorize the verbs in a language according to the sets of VP rules that they participate in.
- This is a modern take on the traditional notion of transitive/intransitive.
- Modern grammars may have 100s or such classes.

28

Subcategorization

- 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]$_{S}$
- …

29

Subcategorization

- ***John sneezed the book**
- ***I prefer United has a flight**
- ***Give with a flight**

- As with agreement phenomena, we need a way to formally express the constraints!

30

Why?

- Right now, 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

31

Possible CFG Solution

- Possible solution for agreement.
- Can use the same trick for all the verb/VP classes.

- SgS -> SgNP SgVP
- PlS -> PlNp PlVP
- SgNP -> SgDet SgNom
- PlNP -> PlDet PlNom
- PlVP -> PlV NP
- SgVP ->SgV Np
- ...

32

CFG Solution for Agreement

- It works and stays within the power of CFGs
- But its ugly
- And it doesn't scale all that well because of the interaction among the various constraints explodes the number of rules in our grammar.

33

To conclude

- CFGs are simple and capture a lot of basic syntactic structure in English.
- But there are problems
  - Don't handle "agreement" and "subcategorization"
  - Overgenerate!
- Advanced grammars
  - LFG
  - HPSG
  - Construction grammar
  - XTAG

34

Treebanks

- Treebanks are corpora in which each sentence has been paired with a parse tree (presumably the right one).

35

Penn Treebank

- Penn TreeBank is a widely used treebank.

▪Most well known is the Wall Street Journal section of the Penn TreeBank.

  ▪1 M words from the 1987-1989 Wall Street Journal.

```
( (S ('' '')
    (S-TPC-2
      (NP-SBJ-1 (PRP We) )
      (VP (MD would)
        (VP (VB have)
          (S
            (NP-SBJ (-NONE- *-1) )
            (VP (TO to)
              (VP (VB wait)
                (SBAR-TMP (IN until)
                  (S
                    (NP-SBJ (PRP we) )
                    (VP (VBP have)
                      (VP (VBN collected)
                        (PP-CLR (IN on)
                          (NP (DT those)(NNS assets)))))))))))))
    (, ,) ('' '')
    (NP-SBJ (PRP he) )
    (VP (VBD said)
      (S (-NONE- *T*-2) ))
    (. .) ))
```

36

Treebank Grammars

- Such grammars tend to be very flat due to the fact that they tend to avoid recursion.
  - To ease the annotators burden
- For example, the Penn Treebank has 4500 different rules for VPs. Among them...

$$VP \rightarrow VBD\ PP$$
$$VP \rightarrow VBD\ PP\ PP$$
$$VP \rightarrow VBD\ PP\ PP\ PP$$
$$VP \rightarrow VBD\ PP\ PP\ PP\ PP$$

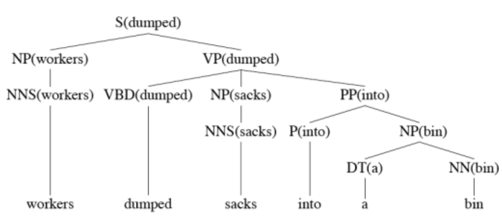
37

Heads in Trees

- Finding heads in treebank trees is a task that arises frequently in many applications.
  - Particularly important in statistical parsing
- We can visualize this task by annotating the nodes of a parse tree with the heads of each corresponding node.

38

Lexically Decorated Tree



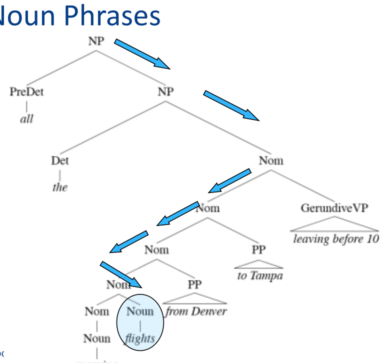
39

Head Finding

- The standard way to do head finding is to use a simple set of tree traversal rules specific to each non-terminal in the grammar.

40

Noun Phrases



41 Proc 10/24/2010

Treebank Uses

- Treebanks (and headfinding) are particularly critical to the development of statistical parsers
  - Chapter 14
- Also valuable to *Corpus Linguistics*
  - Investigating the empirical details of various constructions in a given language

42

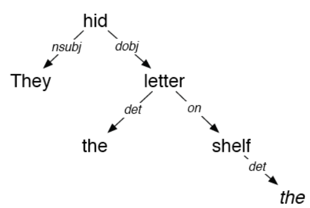
Dependency Grammars

- In CFG-style phrase-structure grammars the main focus is on ***constituents***.
- But it turns out you can get a lot done with just binary relations among the words in an utterance.
- In a **dependency grammar** framework, a parse is a tree where
  - the nodes stand for the words in an utterance
  - The links between the words represent dependency relations between pairs of words.
    - Relations may be typed (labeled), or not.

Dependency Relations

Argument Dependencies	Description
nsubj	nominal subject
csubj	clausal subject
dobj	direct object
iobj	indirect object
pobj	object of preposition
Modifier Dependencies	**Description**
tmod	temporal modifier
appos	appositional modifier
det	determiner
prep	prepositional modifier

Dependency Parse



They hid the letter on the shelf

Dependency Parsing

- The dependency approach has a number of advantages over full phrase-structure parsing.
  - Deals well with free word order languages where the constituent structure is quite fluid
  - Parsing is much faster than CFG-bases parsers
  - Dependency structure often captures the syntactic relations needed by later applications
    - CFG-based approaches often extract this same information from trees anyway.

Dependency Parsing

- There are two modern approaches to dependency parsing
  - Optimization-based approaches that search a space of trees for the tree that *best* matches some criteria
  - Shift-reduce approaches that greedily take actions based on the current word and state.

Summary

- Context-free grammars can be used to model various facts about the syntax of a language.
- When paired with parsers, such grammars constitute a critical component in many applications.
- Constituency is a key phenomena easily captured with CFG rules.
  - But agreement and subcategorization do pose significant problems
- Treebanks pair sentences in corpus with their corresponding trees.