Syntactic Grammaticality

Doesn’t depend on
• Having heard the sentence before
• The sentence being true
  – I saw a unicorn yesterday
• The sentence being meaningful
  – Colorless green ideas sleep furiously
  – *Furiously sleep ideas green colorless
  – I sperred a couple of gurry fipps.

Grammatically is a formal property that we can investigate and describe

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
    – Whole group appears in similar syntactic environment
    – E.g., before a verb
    – Preposed/postposed constructions
    – Note: notions of meaning play no role in syntax (sort-of)

What is Syntax?

• Study of structure of language
• Specifically, goal is to relate surface form (e.g., interface to phonological component) to semantics (e.g., interface to semantic component)
• Morphology, phonology, semantics farmed out (mainly), issue is word order and structure
• Representational device is tree structure

What About Chomsky?

• At birth of formal language theory (comp sci) and formal linguistics
• Major contribution: syntax is cognitive reality
• Humans able to learn languages quickly, but not all languages ⇒ universal grammar is biological
• Goal of syntactic study: find universal principles and language-specific parameters
• Specific Chomskyan theories change regularly
• These theories adopted by almost all contemporary syntactic theories (‘principles-and-parameters-type theories’)
Types of Linguistic Activity

- **Descriptive**: provide account of syntax of a language; often good enough for NLP engineering work
- **Explanatory**: provide principles-and-parameters style account of syntax of (preferably) several languages
- **Prescriptive**: “prescriptive linguistics” is an oxymoron

key ideas of syntax

- Constituency (we’ll spend most of our time on this)
- Subcategorization
- Grammatical relations
- Movement/long-distance dependency

Structure in Strings

- Some words: the a small nice big very boy girl sees likes
- Some good sentences:
  - (the) boy likes (a girl)
  - (the small) girl likes (the big) girl
  - (a very small nice) boy sees (a very nice) boy
- Some bad sentences:
  - *(the) boy (the girl)
  - *(small) boy (likes the nice girl)
- Can we find subsequences of words (constituents) which in some way behave alike?

Structure in Strings

Proposal 1

- Some words: the a small nice big very boy girl sees likes
- Some good sentences:
  - (the) boy (likes a girl)
  - (the small) girl (likes the big) girl
  - (a very small nice) boy (sees a very nice) boy
- Some bad sentences:
  - *(the) boy (the girl)
  - *(small) boy (likes the nice girl)

Structure in Strings

Proposal 2

- Some words: the a small nice big very boy girl sees likes
- Some good sentences:
  - (the) boy likes (a girl)
  - (the small) girl likes (the big) girl
  - (a very small nice) boy sees (a very nice) boy
- Some bad sentences:
  - *(the) boy (the girl)
  - *(small) boy (likes the nice girl)
- This is better proposal: fewer types of constituents (blue and red are of same type)

More Structure in Strings

Proposal 2 -- ctd

- Some words: the a small nice big very boy girl sees likes
- Some good sentences:
  - (the) boy (likes (a) girl)
  - (the) (small) girl (likes (the) (big) girl)
  - (a) ((very) small) (nice) boy sees ((a) ((very) nice) girl)
- Some bad sentences:
  - *(the) boy (the) girl
  - *(small) boy (likes (the) (nice) girl)
From Substrings to Trees

- ( ((the) boy) likes ( (a) girl ))

Node Labels?

- ( ((the) boy) likes ( (a) girl ) )
- Choose constituents so each one has one non-bracketed word: the head
- Group words by distribution of constituents they head (part-of-speech, POS).
  - Noun (N), verb (V), adjective (Adj), adverb (Adv), determiner (Det)
- Category of constituent: XP, where X is POS
  - NP, S, AdjP, AdvP, DetP

Node Labels

- ( ((the/DET boy/NP) likes/V NP ( (a/DET girl/NP) ))

Types of Nodes

- ( ((the/DET boy/NP) likes/V NP ( (a/DET girl/NP) ))

Determining Part-of-Speech

A blue seat/ a child seat: noun or adjective?

- Syntax:
  - a blue seat / a child seat
  - a very blue seat / a very child seat
  - this seat is blue / this seat is child
- Morphology:
  - bluer / childer
- blue and child are not the same POS
- blue is Adj, child is Noun

Determining Part-of-Speech (2)

- preposition or particle?
  - A he threw out the garbage
  - B he threw the garbage out the door
  - A he threw the garbage out
  - B *he threw the garbage the door out
  - The two out are not same POS; A is particle, B is Preposition
Word Classes (=POS)

- Heads of constituents fall into distributionally defined classes
- Additional support for class definition of word class comes from morphology

Constituency (Review)

- E.g., Noun phrases (NPs)
  - A red dog on a blue tree
  - A blue dog on a red tree
  - Some big dogs and some little dogs
  - A dog
    - *
    - Big dogs, little dogs, red dogs, blue dogs, yellow dogs, green dogs, black dogs, and white dogs
- How do we know these form a constituent?

Constituency (II)

- They can all appear before a verb:
  - Some big dogs and some little dogs are going around in cars...
  - Big dogs, little dogs, red dogs, blue dogs, yellow dogs, green dogs, black dogs, and white dogs are all at a dog party!
  - I do not
- But individual words can’t always appear before verbs:
  - *little are going...
  - *blue are...
  - *and are
- Must be able to state generalizations like:
  - Noun phrases occur before verbs
Constituency (III)

- Preposing and postposing:
  - *Under a tree* is a yellow dog.
  - *A yellow dog is under a tree.*

- But not:
  - *Under, is a yellow dog a tree.
  - *Under a is a yellow dog tree.

- Prepositional phrases notable for ambiguity in attachment

Phrase Structure and Dependency Structure

Phrase Structure and Dependency Structure (ctd)

Types of Dependency

Grammatical Relations

- Types of relations between words
  - Arguments: subject, object, indirect object, prepositional object
  - Adjuncts: temporal, locative, causal, manner, ...
  - Function Words
Subcategorization

- List of arguments of a word (typically, a verb), with features about realization (POS, perhaps case, verb form etc)
- In canonical order Subject-Object-IndObj
- Example:
  - like: N-N, N-V(to-inf)
  - see: N, N-N, N-N-V(inf)
- Note: J&M talk about subcategorization only within VP

What About the VP?

- Existence of VP is a linguistic (i.e., empirical) claim, not a methodological claim
- Semantic evidence???
- Syntactic evidence
  - VP-fronting (and quickly clean the carpet he did!)
  - VP-ellipsis (He cleaned the carpets quickly, and so did she)
- Note: VP cannot be represented in a dependency representation

Context-Free Grammars

- Defined in formal language theory (comp sci)
- Terminals, nonterminals, start symbol, rules
- String-rewriting system
- Start with start symbol, rewrite using rules, done when only terminals left
- NOT A LINGUISTIC THEORY, just a formal device

CFG: Example

- Many possible CFGs for English, here is an example (fragment):
  - $S \rightarrow NP \ VP$
  - $VP \rightarrow V NP$
  - $NP \rightarrow DetP \ N \mid AdjP \ NP$
  - $AdjP \rightarrow Adj \mid Adv \ AdjP$
  - $N \rightarrow boy \mid girl$
  - $V \rightarrow sees \mid likes$
  - $Adj \rightarrow big \mid small$
  - $Adv \rightarrow very$
  - $DetP \rightarrow a \mid the$

the very small boy likes a girl

Derivations in a CFG

$S$

$S \rightarrow NP \ VP$
$VP \rightarrow V NP$
$NP \rightarrow DetP \ N \mid AdjP \ NP$
$AdjP \rightarrow Adj \mid Adv \ AdjP$
$N \rightarrow boy \mid girl$
$V \rightarrow sees \mid likes$
$Adj \rightarrow big \mid small$
$Adv \rightarrow very$
$DetP \rightarrow a \mid the$
Derivations in a CFG

NP VP

S → NP VP
VP → V NP
NP → DetP N | AdjP NP
AdjP → Adj | Adv AdjP
N → boy | girl
V → sees | likes
Adj → big | small
Adv → very
DetP → a | the

Derivations in a CFG

the boy VP

S → NP VP
VP → V NP
NP → DetP N | AdjP NP
AdjP → Adj | Adv AdjP
N → boy | girl
V → sees | likes
Adj → big | small
Adv → very
DetP → a | the

Derivations in a CFG

the boy likes NP

S → NP VP
VP → V NP
NP → DetP N | AdjP NP
AdjP → Adj | Adv AdjP
N → boy | girl
V → sees | likes
Adj → big | small
Adv → very
DetP → a | the

Derivations in a CFG

the boy likes a girl

S → NP VP
VP → V NP
NP → DetP N | AdjP NP
AdjP → Adj | Adv AdjP
N → boy | girl
V → sees | likes
Adj → big | small
Adv → very
DetP → a | the

Derivations in a CFG;
Order of Derivation Irrelevant

NP likes DetP girl
## Derivations of CFGs

- **String rewriting system:** we derive a string (=derived structure)
- **But derivation history represented by phrase-structure tree (=derivation structure):**

```
S
   NP        VP
  the boy  likes  a girl
```

## Formal Definition of a CFG

\[ G = (V, T, P, S) \]

- \( V \): finite set of nonterminal symbols
- \( T \): finite set of terminal symbols, \( V \) and \( T \) are disjoint
- \( P \): finite set of productions of the form \( A \rightarrow \alpha, A \in V \) and \( \alpha \in (T \cup V)^* \)
- \( S \in V \): start symbol

## Context?

- The notion of context in CFGs has nothing to do with the ordinary meaning of the word context in language
- All it really means is that the non-terminal on the left-hand side of a rule is out there all by itself (free of context)

\[ A \rightarrow B C \]

Means that I can rewrite an \( A \) as a \( B \) followed by a \( C \) regardless of the context in which \( A \) is found

## Key Constituents (English)

- **Sentences**
- **Noun phrases**
- **Verb phrases**
- **Prepositional phrases**

## Sentence-Types

- **Declaratives:** I do not.
  \[ S \rightarrow NP \ VP \]
- **Imperatives:** Go around again!
  \[ S \rightarrow VP \]
- **Yes-No Questions:** Do you like my hat?
  \[ S \rightarrow Aux \ NP \ VP \]
- **WH Questions:** What are they going to do?
  \[ S \rightarrow WH \ Aux \ NP \ VP \]
NPs

- NP -> Pronoun
  - I came, you saw it, they conquered
- NP -> Proper-Noun
  - New Jersey is west of New York City
  - Lee Bollinger is the president of Columbia
- NP -> Det Noun
  - The president
- NP -> Nominal
- Nominal -> Noun Noun
  - A morning flight to Denver

PPs

- PP -> Preposition NP
  - Over the house
  - Under the house
  - To the tree
  - At play
  - At a party on a boat at night
Recursion

• We’ll have to deal with rules such as the following where the non-terminal on the left also appears somewhere on the right (directly)

  NP → NP PP  [[The flight] [to Boston]]
  VP → VP PP  [[departed Miami] [at noon]]

(indirectly)

  NP → NP Srel
  Srel → NP VP  [ [the dog] [[the cat] likes] ]

Implications of Recursion and Context-Freeness

• VP → V NP
• (I) hate 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

  This is why context-free grammars are appealing! If you have a rule like

  VP → V NP

  It only cares that the thing after the verb is an NP

  It doesn’t have to know about the internal affairs of that NP
Grammar Equivalence

- Can have different grammars that generate same set of strings (weak equivalence)
  - Grammar 1: NP → DetP N and DetP → a | the
  - Grammar 2: NP → a N | NP → the N
- Can have different grammars that have same set of derivation trees (strong equivalence)
  - With CFGs, possible only with useless rules
  - Grammar 2: NP → a N | NP → the N
- Strong equivalence implies weak equivalence

Chomsky Normal Form

A CFG is in Chomsky Normal Form (CNF) if all productions are of one of two forms:
- \( A \rightarrow BC \) with \( A, B, C \) nonterminals
- \( A \rightarrow \alpha \), with \( A \) a nonterminal and \( \alpha \) a terminal

Every CFG has a weakly equivalent CFG in CNF

Normal Forms &c

- There are weakly equivalent normal forms (Chomsky Normal Form, Greibach Normal Form)
- There are ways to eliminate useless productions and so on

“Generative Grammar”

- Formal languages: formal device to generate a set of strings (such as a CFG)
- Linguistics (Chomskyan linguistics in particular): approach in which a linguistic theory enumerates all possible strings/structures in a language (=competence)
- Chomskyan theories do not really use formal devices – they use CFG + informally defined transformations

Nobody Uses Simple CFGs (Except Intro NLP Courses)

- All major syntactic theories (Chomsky, LFG, HPSG, TAG-based theories) represent both phrase structure and dependency, in one way or another
- All successful parsers currently use statistics about phrase structure and about dependency
- Derive dependency through “head percolation”: for each rule, say which daughter is head

Massive Ambiguity of Syntax

- For a standard sentence, and a grammar with wide coverage, there are 1000s of derivations!
- Example:
  - The large portrait painter told the delegation that he sent money orders in a letter on Wednesday
Penn Treebank (PTB)

- Syntactically annotated corpus of newspaper texts (phrase structure)
- The newspaper texts are naturally occurring data, but the PTB is not!
- PTB annotation represents a particular linguistic theory (but a fairly “vanilla” one)
- Particularities
  - Very indirect representation of grammatical relations (need for head percolation tables)
  - Completely flat structure in NP (brown bag lunch, pink-and-yellow child seat)
  - Has flat Ss, flat VPs

Example from PTB

```plaintext
(S [NP SBJ It] [VP's [NP PRD [NP the latest investment craze] [VP sweeping [NP Wall Street]]] [PP of [NP [NP a rash] [NP [NP [NP [NP new closed-end country funds] [NP [NP those [ADJP publicly traded] portfolios] [PP of [NP [NP [NP [NP new single foreign country]))))))]))
```

Types of syntactic constructions

- Is this the same construction?
  - An elf **decided** to clean the kitchen
  - An elf **seemed** to clean the kitchen
  An elf cleaned the kitchen

- Is this the same construction?
  - An elf **decided** to be in the kitchen
  - An elf **seemed** to be in the kitchen
  An elf was in the kitchen

Types of syntactic constructions (ctd)

- Is this the same construction?
  - There is an elf in the kitchen
  - *There **decided** to be an elf in the kitchen
  - There **seemed** to be an elf in the kitchen

- Is this the same construction?
  - It is raining/it rains
  - ??It **decided** to rain/be raining
  - It **seemed** to rain/be raining

Types of syntactic constructions (ctd)

- Is this the same construction?
  - An elf **decided** that he would clean the kitchen
  - * An elf **seemed** that he would clean the kitchen
  An elf cleaned the kitchen

Conclusion:

- **to seem**: whatever is embedded surface subject can appear in upper clause
- **to decide**: only full nouns that are referential can appear in upper clause
- Two types of verbs
Types of syntactic constructions:

Analysis

<table>
<thead>
<tr>
<th>S</th>
<th>NP</th>
<th>VP</th>
</tr>
</thead>
<tbody>
<tr>
<td>an elf</td>
<td>V</td>
<td>S</td>
</tr>
<tr>
<td>to decide</td>
<td>NP</td>
<td>VP</td>
</tr>
<tr>
<td>an elf</td>
<td>V</td>
<td>PP</td>
</tr>
<tr>
<td>to be in the kitchen</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S</th>
<th>NP</th>
<th>VP</th>
</tr>
</thead>
<tbody>
<tr>
<td>an elf</td>
<td>V</td>
<td>S</td>
</tr>
<tr>
<td>to seem</td>
<td>NP</td>
<td>VP</td>
</tr>
<tr>
<td>an elf</td>
<td>V</td>
<td>PP</td>
</tr>
<tr>
<td>to be in the kitchen</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S</th>
<th>NP</th>
<th>VP</th>
</tr>
</thead>
<tbody>
<tr>
<td>an elf</td>
<td>V</td>
<td>S</td>
</tr>
<tr>
<td>decided</td>
<td>PRO</td>
<td>VP</td>
</tr>
<tr>
<td>an elf</td>
<td>V</td>
<td>PP</td>
</tr>
<tr>
<td>to be in the kitchen</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S</th>
<th>NP</th>
<th>VP</th>
</tr>
</thead>
<tbody>
<tr>
<td>an elf</td>
<td>V</td>
<td>S</td>
</tr>
<tr>
<td>seemed</td>
<td>NP</td>
<td>VP</td>
</tr>
<tr>
<td>an elf</td>
<td>V</td>
<td>PP</td>
</tr>
<tr>
<td>to be in the kitchen</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**to seem:** lower surface subject raises to upper clause; **raising verb**

seems (there to be an elf in the kitchen)
there seems (t to be an elf in the kitchen)
it seems (there is an elf in the kitchen)
Types of syntactic constructions: Analysis (ctd)

- *to decide*: subject is in upper clause and co-refers with an empty subject in lower clause; control verb

  an elf decided (an elf to clean the kitchen)
an elf decided (PRO to clean the kitchen)
an elf decided (he cleans/should clean the kitchen)
*it decided (an elf cleans/should clean the kitchen)

Lessons Learned from the Raising/Control Issue

- Use distribution of data to group phenomena into classes
- Use different underlying structure as basis for explanations
- Allow things to “move” around from underlying structure -> transformational grammar
- Check whether explanation you give makes predictions

Examples from PTB

(S NP-SBJ-1 The ropes)
  (VP seem
   (S (NP-SBJ *-1)
   (VP make
     (NP much sound))))

(S NP-SBJ-1 The ancient church vicar)
  (VP refuses
   (S (NP-SBJ *-1)
   (VP to
     (VP talk
       (PP-CLR about
         (NP it)))))

Developing Grammars

- We saw with the previous example a complex structure
- Let’s back off to simple English Structures and see how we would capture them with Context Free Grammars
- Developing a grammar of any size is difficult.

Key Constituents (English)

- Sentences
- Noun phrases
- Verb phrases
- Prepositional phrases

See text for examples of these!
Some NP Rules

- Here are some rules for our noun phrases

\[
NP \rightarrow \text{Det} \ \text{Nominal} \\
NP \rightarrow \text{ProperNoun} \\
\text{Nominal} \rightarrow \text{Nom} \mid \text{Nominal Noun}
\]

- 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
  - Two kinds of nominals
  - A recursive definition
  - Same non-terminal on the right and left side of the rule

An English Grammar Fragment

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

L0 Grammar

<table>
<thead>
<tr>
<th>Grammar Rules</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S \rightarrow NP \ VP )</td>
<td>I want a morning flight</td>
</tr>
<tr>
<td>( NP \rightarrow \text{ProperNoun} )</td>
<td>Los Angeles</td>
</tr>
<tr>
<td>( NP \rightarrow \text{DetNoun} )</td>
<td>a flight</td>
</tr>
<tr>
<td>( \text{Nominal} \rightarrow \text{Nominal Noun} )</td>
<td>morning a flight</td>
</tr>
<tr>
<td>( \text{Nominal} \rightarrow \text{Nom} )</td>
<td>flights</td>
</tr>
<tr>
<td>( \text{VP} \rightarrow \text{Verb} )</td>
<td>do</td>
</tr>
<tr>
<td>( \text{Verb} \rightarrow \text{VerbNoun} )</td>
<td>want a flight</td>
</tr>
<tr>
<td>( \text{Verb} \rightarrow \text{VerbNounPP} )</td>
<td>leave + Boston + in the morning</td>
</tr>
<tr>
<td>( \text{Verb} \rightarrow \text{VerbPP} )</td>
<td>leaving + on Thursday</td>
</tr>
<tr>
<td>( \text{PP} \rightarrow \text{PrepositionNoun} )</td>
<td>from + Los Angeles</td>
</tr>
</tbody>
</table>

Common Sentence Types

- Declaratives: John left
  \( S \rightarrow NP \ VP \)
- Imperatives: Leave!
  \( S \rightarrow NP \ VP \)
- Yes-No Questions: Did John leave?
  \( S \rightarrow \text{Aux} \ NP \ VP \)
- WH Questions (who, what, where, when, which, why, how): When did John leave?
  \( S \rightarrow WH \ Aux \ NP \ VP \)

Noun Phrases

- Let's consider the following rule in more detail...
  \( NP \rightarrow \text{Det} \ \text{Nominal} \)
- Most of the complexity of English noun phrases is hidden in this rule.
- Consider the derivation for the following example
  - All the morning flights from Denver to Tampa leaving before 10
NP Structure

- Clearly this NP is really about *flights*. That's the central critical noun in this NP. Let's call that 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.

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

Nominals

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

Postmodifiers

- Three kinds
  - Prepositional phrases
    - From Seattle
  - Non-finite clauses
    - Arriving before noon
  - Relative clauses
    - That serve breakfast
- Same general (recursive) rule to handle these
  - Nominal → Nominal PP
  - Nominal → Nominal GerundVP
  - Nominal → Nominal RelClause

Agreement

- By *agreement*, we have in mind constraints that hold among various constituents that take part in a rule or set of rules
- For example, in English, determiners and the head nouns in NPs have to agree in their number.

  This flight
  Those flights

  *This flights
  * Those flight

Problem

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

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

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.

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-VP
- Told: I was told [United has a flight]S
- ...

Possible CFG Solution

- VP -> V
- VP -> V NP
- VP -> V NP PP
- ...
- VP -> IntransV
- VP -> TransV NP
- VP -> TransPP NP PP
- ...

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
    Sneeze the book is a VP since "sneeze" is a verb and "the book" is a valid NP
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

• Agreement
• Subcategorization
• Movement (for want of a better term)

Agreement

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

Handing Number Agreement in CFGs

• To handle, would need to expand the grammar with multiple sets of rules – but it gets rather messy quickly.
  • NP_sg → Det_sg N_sg
  • NP_pl → Det_pl N_pl
  • ...
  • VP_sg → V_sg NP_sg
  • VP_sg → V_sg NP_pl
  • VP_pl → V_pl NP_sg
  • VP_pl → V_pl NP_pl

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.

Movement

• Core example
  → My travel agent booked the flight
Movement

• Core example
  – [[My travel agent]\(_\text{agt}\) [booked [the flight]\(_\text{obj}\)]]

• 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.

The Point

• 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)
    • LFG, HPSG, Construction grammar, XTAG, etc.
    • Chapter 15 explores the unification approach in more detail