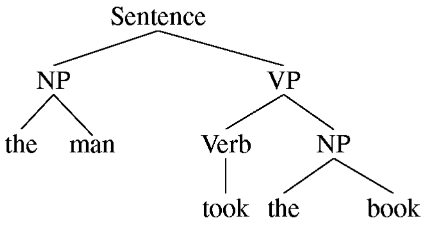
9 CONTEXT-FREE GRAMMARS FOR ENGLISH



The first context-free grammar parse tree (Chomsky, 1956)

If on a winter's night a traveler by Italo Calvino
Nuclear and Radiochemistry by Gerhart Friedlander et al.
The Fire Next Time by James Baldwin
A Tad Overweight, but Violet Eyes to Die For by G. B. Trudeau
Sometimes a Great Notion by Ken Kesey
Dancer from the Dance by Andrew Holleran

6 books in English whose titles are not constituents,
from Pullum (1991, p. 195)

In her essay *The Anatomy of a Recipe*, M. F. K. Fisher (1968) wryly comments that it is "modish" to refer to the *anatomy* of a thing or problem. The similar use of *grammar* to describe the structures of an area of knowledge had a vogue in the 19th century (e.g. Busby's (1818) *A Grammar of Music* and Field's (1888) *A Grammar of Colouring*). In recent years the word *grammar* has made a reappearance, although usually now it is *the* grammar rather than *a* grammar that is being described (e.g. *The Grammar of Graphics*, *The Grammar of Conducting*). Perhaps scholars are simply less modest than they used to be? Or perhaps the word *grammar* itself has changed a bit, from 'a listing of principles or structures', to 'those principles or structures as an field of inquiry'. Following this second reading, in this chapter we turn to what might be called *The Grammar of Grammar*, or perhaps *The Grammar of Syntax.*

SYNTAX

The word **syntax** comes from the Greek *sýntaxis*, meaning 'setting out together or arrangement', and refers to the way words are arranged together. We have seen various syntactic notions in previous chapters. Chapter 8 talked about part-of-speech categories as a kind of equivalence class for words. Chapter 6 talked about the importance of modeling word order. This chapter and the following ones introduce a number of more complex notions of syntax and grammar. There are three main new ideas: **constituency**, **grammatical relations**, and **subcategorization and dependencies**.

CON-STITUENT

The fundamental idea of constituency is that groups of words may behave as a single unit or phrase, called a **constituent**. For example we will see that a group of words called a **noun phrase** often acts as a unit; noun phrases include single words like *she* or *Michael* and phrases like *the house*, *Russian Hill*, and *a well-weathered three-story structure.* This chapter will introduce the use of **context-free grammars**, a formalism that will allow us to model these constituency facts.

Grammatical relations are a formalization of ideas from traditional grammar about SUBJECTS and OBJECTS. In the sentence:

(9.1) She ate a mammoth breakfast.

the noun phrase *She* is the SUBJECT and *a mammoth breakfast* is the OBJECT. Grammatical relations will be introduced in this chapter when we talk about syntactic **agreement**, and will be expanded upon in Chapter 11.

Subcategorization and **dependency relations** refer to certain kinds of relations between words and phrases. For example the verb *want* can be followed by an infinitive, as in *I want to fly to Detroit*, or a noun phrase, as in *I want a flight to Detroit.* But the verb *find* cannot be followed by an infinitive (**I found to fly to Dallas*). These are called facts about the *subcategory* of the verb, which will be discussed starting on page 337, and again in Chapter 11.

All of these kinds of syntactic knowledge can be modeled by various kinds of grammars that are based on context-free grammars. Context-free grammars are thus the backbone of many models of the syntax of natural language (and, for that matter, of computer languages). As such they are integral to most models of natural language understanding, of grammar checking, and more recently of speech understanding. They are powerful enough to express sophisticated relations among the words in a sentence, yet computationally tractable enough that efficient algorithms exist for parsing sentences with them (as we will see in Chapter 10). Later in Chapter 12 we will introduce probabilistic versions of context-free grammars, which model many aspects of human sentence processing and which provide sophisticated language models for speech recognition.

In addition to an introduction to the grammar formalism, this chapter also provides an overview of the grammar of English. We will be modeling example sentences from the Air Traffic Information System (ATIS) domain (Hemphill *et al.*, 1990). ATIS systems are spoken language systems that can help book airline reservations. Users try to book flights by conversing with the system, specifying constraints like *I'd like to fly from Atlanta to Denver.* The government funded a number of different research sites across the country to build ATIS systems in the early 90's, and so a lot of data was collected and a significant amount of research has been done on the resulting data. The sentences we will be modeling in this chapter are the user queries to the system.

9.1 CONSTITUENCY

How do words group together in English? How do we know they are really grouping together? Let's consider the standard grouping that is usually called the **noun phrase** or sometimes the **noun group**. This is a sequence of words surrounding at least one noun. Here are some examples of noun phrases (thanks to Damon Runyon):

NOUN PHRASE

NOUN GROUP

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

How do we know that these words group together (or 'form a constituent')? One piece of evidence is that they can all appear in similar syntactic environments, for example before a verb.

three parties from Brooklyn *arrive...*
a high-class spot such as Mindy's *attracts...*
the Broadway coppers *love...*
they *sit*

But while the whole noun phrase can occur before a verb, this is not true of each of the individual words that make up a noun phrase. The following are not grammatical sentences of English (recall that we use an asterisk (*) to mark fragments that are not grammatical English sentences):

*from *arrive...*
*as *attracts...*
*the *is...*
*spot *is...*

Thus in order to correctly describe facts about the ordering of these words in English, we must be able to say things like *"Noun Phrases can occur before verbs"*.

PREPOSED
POSTPOSED

Other kinds of evidence for constituency come from what are called **preposed** or **postposed** constructions. For example, the prepositional phrase *on September seventeenth* can be placed in a number of different locations in the following examples, including preposed at the beginning, and postposed at the end:

On September seventeenth, I'd like to fly from Atlanta to Denver
I'd like to fly on September seventeenth from Atlanta to Denver
I'd like to fly from Atlanta to Denver on September seventeenth

But again, while the entire phrase can be placed differently, the individual words making up the phrase cannot be:

*On September, I'd like to fly seventeenth from Atlanta to Denver
*On I'd like to fly September seventeenth from Atlanta to Denver
*I'd like to fly on September from Atlanta to Denver seventeenth

Section 9.11 will give other motivations for context-free grammars based on their ability to model recursive structures.

There are many other kinds of evidence that groups of words often behave as a single constituent (see Radford (1988) for a good survey).

9.2 CONTEXT-FREE RULES AND TREES

CFG

The most commonly used mathematical system for modeling constituent structure in English and other natural languages is the **Context-Free Grammar**, or **CFG**. Context-free grammars are also called **Phrase-Structure**

Grammars, and the formalism is equivalent to what is also called **Backus-Naur Form** or **BNF**. The idea of basing a grammar on constituent structure dates back to the psychologist Wilhelm Wundt (1900), but was not formalized until Chomsky (1956), and, independently, Backus (1959).

RULES

LEXICON

NP

A context-free grammar consists of a set of **rules** or **productions**, each of which expresses the ways that symbols of the language can be grouped and ordered together, and a **lexicon** of words and symbols. For example, the following productions expresses that a **NP** (or **noun phrase**), can be composed of either a *ProperNoun* or of a determiner (*Det*) followed by a *Nominal*; a *Nominal* can be one or more *Noun*s.

$$NP \rightarrow Det\ Nominal \tag{9.2}$$

$$NP \rightarrow ProperNoun \tag{9.3}$$

$$Nominal \rightarrow Noun \mid Noun\ Nominal \tag{9.4}$$

Context free rules can be hierarchically embedded, so we could combine the previous rule with others like these which express facts about the lexicon:

$$Det \rightarrow a \tag{9.5}$$

$$Det \rightarrow the \tag{9.6}$$

$$Noun \rightarrow flight \tag{9.7}$$

TERMINAL

NONTERMINAL

The symbols that are used in a CFG are divided into two classes. The symbols that correspond to words in the language ('the', 'nightclub') are called **terminal** symbols; the lexicon is the set of rules that introduce these terminal symbols. The symbols that express clusters or generalizations of these are called **nonterminals**. In each context-free rule, the item to the right of the arrow (→) is an ordered list of one or more terminals and nonterminals, while to the left of the arrow is a single nonterminal symbol expressing some cluster or generalization. Notice that in the lexicon, the nonterminal associated with each word is its lexical category, or part-of-speech, which we defined in Chapter 8.

A CFG is usually thought of in two ways: as a device for generating sentences, or as a device for assigning a structure to a given sentence. As a generator, we could read the → arrow as 'rewrite the symbol on the left with the string of symbols on the right'. So starting from the symbol

NP,

we can use rule 9.2 to rewrite *NP* as

Det Nominal,

and then rule 9.4:

Det Noun,

and finally via rules 9.5 and 9.7 as

a flight,

DERIVED
DERIVATION
PARSE TREE

We say the string *a flight* can be **derived** from the nonterminal *NP*. Thus a CFG can be used to randomly generate a series of strings. This sequence of rule expansions is called a **derivation** of the string of words. It is common to represent a derivation by a **parse tree** (commonly shown inverted with the root at the top). Here is the tree representation of this derivation:

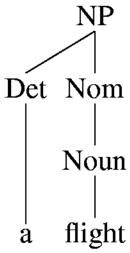


Figure 9.1 A parse tree for 'a flight'.

START SYMBOL

The formal language defined by a CFG is the set of strings that are derivable from the designated **start symbol**. Each grammar must have one designated start symbol, which is often called *S*. Since context-free grammars are often used to define sentences, *S* is usually interpreted as the 'sentence' node, and the set of strings that are derivable from *S* is the set of sentences in some simplified version of English.

VERB PHRASE

Let's add to our sample grammar a couple of higher-level rules that expand *S*, and a couple others. One will express the fact that a sentence can consist of a noun phrase and a **verb phrase**:

$S \rightarrow NP\ VP$ I prefer a morning flight

A verb phrase in English consists of a verb followed by assorted other things; for example, one kind of verb phrase consists of a verb followed by a noun phrase:

VP → *Verb NP* prefer a morning flight

Or the verb phrase may have a noun phrase and a prepositional phrase:

VP → *Verb NP PP* leave Boston in the morning

Or the verb may be followed just by a preposition-phrase:

VP → *Verb PP* leaving on Thursday

A prepositional phrase generally has a preposition followed by a noun phrase. For example, a very common type of prepositional phrase in the ATIS corpus is used to indicate location or direction:

PP → *Preposition NP* from Los Angeles

The NP inside a PP need not be a location; PPs are often used with times and dates, and with other nouns as well; they can be arbitrarily complex. Here are ten examples from the ATIS corpus:

to Seattle	on these flights
in Minneapolis	about the ground transportation in Chicago
on Wednesday	of the round trip flight on United Airlines
in the evening	of the AP fifty seven flight
on the ninth of July	with a stopover in Nashville

Figure 9.2 gives a sample lexicon and Figure 9.3 summarizes the grammar rules we've seen so far, which we'll call $\mathcal{L}_0$. Note that we can use the or-symbol | to indicate that a non-terminal has alternate possible expansions.

We can use this grammar to generate sentences of this 'ATIS-language'. We start with *S*, expand it to *NP VP*, then choose a random expansion of *NP* (let's say to *I*), and a random expansion of *VP* (let's say to *Verb NP*), and so on until we generate the string *I prefer a morning flight.* Figure 9.4 shows a parse tree that represents a complete derivation of *I prefer a morning flight.*

BRACKETED NOTATION

It is sometimes convenient to represent a parse tree in a more compact format called **bracketed notation**, essentially the same as LISP tree representation; here is the bracketed representation of the parse tree of Figure 9.4:

$$[_{S}\ [_{NP}\ [_{Pro}\ \text{I}]]\ [_{VP}\ [_{V}\ \text{prefer}]\ [_{NP}\ [_{Det}\ \text{a}]\ [_{Nom}\ [_{N}\ \text{morning}]\ [_{N}\ \text{flight}]]]]]$$

A CFG like that of $\mathcal{L}_0$ defines a formal language. We saw in Chapter 2 that a formal language is a set of strings. Sentences (strings of words) that can be derived by a grammar are in the formal language defined by that

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

Figure 9.2 The lexicon for $\mathcal{L}_0$.

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

Figure 9.3 The grammar for $\mathcal{L}_0$, with example phrases for each rule.

GRAMMATICAL

UNGRAMMATICAL

grammar, and are called **grammatical** sentences. Sentences that cannot be derived by a given formal grammar are not in the language defined by that grammar, and are referred to as **ungrammatical**. This hard line between ‘in’ and ‘out’ characterizes all formal languages but is only a very simplified model of how natural languages really work. This is because determining

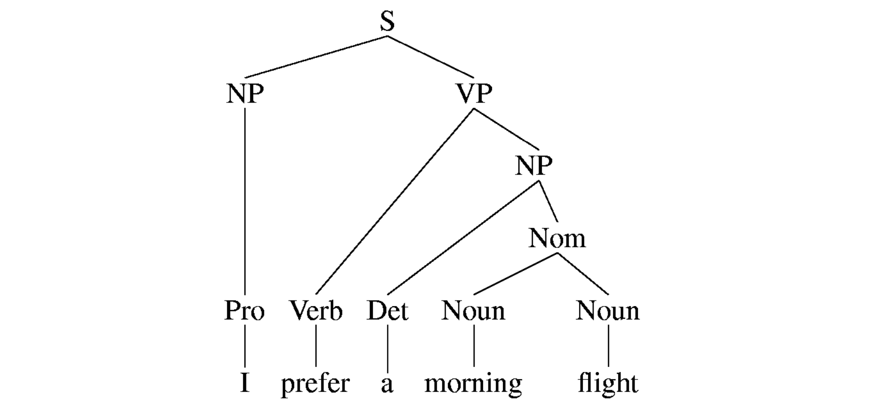


Figure 9.4 The parse tree for ‘I prefer a morning flight’ according to grammar $\mathcal{L}_0$

whether a given sentence is part of a given natural language (say English) often depends on the context. In linguistics, the use of formal languages to model natural languages is called **generative grammar**, since the language is defined by the set of possible sentences ‘generated’ by the grammar.

GENERATIVE GRAMMAR

We conclude this section by way of summary with a quick formal description of a context free grammar and the language it generates. A context-free grammar has four parameters (technically ‘is a 4-tuple’):

1. a set of non-terminal symbols (or ‘variables’) N
2. a set of terminal symbols Σ (disjoint from N)
3. a set of productions P, each of the form $A \rightarrow \alpha$, where A is a non-terminal and α is a string of symbols from the infinite set of strings $(\Sigma \cup N)*$.
4. a designated start symbol S

A language is defined via the concept of **derivation**. One string **derives** another one if it can be rewritten as the second one via some series of rule applications. More formally, following Hopcroft and Ullman (1979), if $A \rightarrow \beta$ is a production of P and α and γ are any strings in the set $(\Sigma \cup N)*$, then we say that $\alpha A \gamma$ **directly derives** $\alpha\beta\gamma$, or $\alpha A \gamma \Rightarrow \alpha\beta\gamma$. Derivation is then a generalization of direct derivation. Let $\alpha_1, \alpha_2, \ldots, \alpha_m$ be strings in $(\Sigma \cup N)*, m \geq 1$, such that

DIRECTLY DERIVES

$$\alpha_1 \Rightarrow \alpha_2, \alpha_2 \Rightarrow \alpha_3, \ldots, \alpha_{m-1} \Rightarrow \alpha_m \tag{9.8}$$

We say that α_1 **derives** α_m, or $\alpha_1 \overset{*}{\Rightarrow} \alpha_m$.

DERIVES

We can then formally define the language $\mathcal{L}_G$ generated by a grammar G as the set of strings composed of terminal symbols which can be derived from the designed start symbol S.

$$\mathcal{L}_G = W|w \text{ is in } \Sigma* \text{ and } S \overset{*}{\Rightarrow} w \tag{9.9}$$

PARSING

The problem of mapping from a string of words to its parse tree is called **parsing**; we will define algorithms for parsing in Chapter 10 and in Chapter 12.

9.3 Sentence-Level Constructions

The remainder of this chapter will introduce a few of the more complex aspects of the phrase structure of English; for consistency we will continue to focus on sentences from the ATIS domain. Because of space limitations, our discussion will necessarily be limited to highlights. Readers are strongly advised to consult Quirk *et al.* (1985a), which is by far the best current reference grammar of English.

In the small grammar $\mathcal{L}_0$, we only gave a single sentence-level construction for declarative sentences like *I prefer a morning flight*. There are a great number of possible overall sentence structures, but 4 are particularly common and important: declarative structure, imperative structure, yes-no-question structure, and wh-question structure,

DECLARATIVE

Sentences with **declarative** structure have a subject noun phrase followed by a verb phrase, like 'I prefer a morning flight'. Sentences with this structure have a great number of different uses that we will follow up on in Chapter 19. Here are a number of examples from the ATIS domain:

The flight should be eleven a.m tomorrow
I need a flight to Seattle leaving from Baltimore making a stop in Minneapolis
The return flight should leave at around seven p.m
I would like to find out the flight number for the United flight that arrives in San Jose around ten p.m
I'd like to fly the coach discount class
I want a flight from Ontario to Chicago
I plan to leave on July first around six thirty in the evening

IMPERATIVE

Sentences with **imperative** structure often begin with a verb phrase, and have no subject. They are called imperative because they are almost always used for commands and suggestions; in the ATIS domain they are commands to the system.

Show the lowest fare
Show me the cheapest fare that has lunch
Give me Sunday's flights arriving in Las Vegas from Memphis and New York City
List all flights between five and seven p.m
List all flights from Burbank to Denver
Show me all flights that depart before ten a.m and have first class fares
Show me all the flights leaving Baltimore
Show me flights arriving within thirty minutes of each other
Please list the flights from Charlotte to Long Beach arriving after lunch time
Show me the last flight to leave

To model this kind of sentence structure, we can add another rule for the expansion of *S*:

$S \rightarrow VP$ Show the lowest fare

YES-NO-QUESTION

Sentences with **yes-no-question** structure are often (though not always) used to ask questions (hence the name), and begin with a auxiliary verb, followed by a subject *NP*, followed by a *VP*. Here are some examples (note that the third example is not really a question but a command or suggestion; Chapter 19 will discuss the **pragmatic** uses of these question forms):

Do any of these flights have stops?
Does American's flight eighteen twenty five serve dinner?
Can you give me the same information for United?

Here's the rule:

$S \rightarrow Aux\ NP\ VP$

WH- PHRASE

WH- WORD

The most complex of the sentence-level structures we will examine are the various **wh-** structures. These are so named because one of their constituents is a **wh- phrase**, i.e. one that includes a **wh- word** (who, where, what, which, how, why). These may be broadly grouped into two classes of sentence-level structures. The **wh-subject-question** structure is identical to the declarative structure, except that the first noun phrase contains some wh-word.

What airlines fly from Burbank to Denver?
Which flights depart Burbank after noon and arrive in Denver by six p.m?
Which flights serve breakfast?
Which of these flights have the longest layover in Nashville?

Here is a rule. Exercise 9.10 discusses rules for the constituents that make up the *Wh-NP*.

$S \rightarrow \textit{Wh-NP}\ VP$

WH-NON-SUBJECT-QUESTION

In the **wh-non-subject-question** structure, the wh-phrase is not the subject of the sentence, and so the sentence includes another subject. In these types of sentences the auxiliary appears before the subject *NP*, just as in the yes-no-question structures. Here is an example:

What flights do you have from Burbank to Tacoma Washington?

Here is a sample rule:

$S \rightarrow \textit{Wh-NP}\ Aux\ NP\ VP$

There are other sentence-level structures we won't try to model here, like **fronting**, in which a phrase is placed at the beginning of the sentence for various discourse purposes (for example often involving topicalization and focus):

On Tuesday, I'd like to fly from Detroit to Saint Petersburg

9.4 The Noun Phrase

HEAD

We can view the noun phrase as revolving around a **head**, the central noun in the noun phrase. The syntax of English allows for both prenominal (pre-head) modifiers and post-nominal (post-head) modifiers.

Before the Head Noun

We have already discussed some of the parts of the noun phrase; the determiner, and the use of the *Nominal* constituent for representing double noun phrases. We have seen that noun phrases can begin with a determiner, as follows:

a stop
the flights
that fare
this flight
those flights
any flights
some flights

There are certain circumstances under which determiners are optional in English. For example, determiners may be omitted if the noun they modify is plural:

Show me *flights* from San Francisco to Denver on weekdays

As we saw in Chapter 8, **mass nouns** don't require determination. Recall that mass nouns often (not always) involve something that is treated like a substance (including e.g. *water* and *snow*), don't take the indefinite article '*a*', and don't tend to pluralize. Many abstract nouns are mass nouns (*music*, *homework*). Mass nouns in the ATIS domain include *breakfast*, *lunch*, and *dinner*:

Does this flight serve dinner?

Exercise 9.4 asks the reader to represent this fact in the CFG formalism.

Word classes that appear in the NP before the determiner are called **predeterminers**. Many of these have to do with number or amount; a common predeterminer is *all*:

PREDETERMINERS

all the flights
all flights

A number of different kinds of word classes can appear in the NP between the determiner and the head noun (the 'post-determiners'). These include **cardinal numbers**, **ordinal numbers**, and **quantifiers**. Examples of cardinal numbers:

CARDINAL NUMBERS
ORDINAL NUMBERS
QUANTIFIERS

two friends
one stop

Ordinal numbers include *first*, *second*, *third*, etc, but also words like *next*, *last*, *past*, *other*, and *another*:

the first one
the next day
the second leg
the last flight
the other American flight
any other fares

Some quantifiers (*many*, *(a) few*, *several*) occur only with plural count nouns:

many fares

The quantifiers *much* and *a little* occur only with noncount nouns.
Adjectives occur after quantifiers but before nouns.

a *first-class* fare
a *nonstop* flight
the *longest* layover
the *earliest* lunch flight

ADJECTIVE PHRASE

AP

Adjectives can also be grouped into a phrase called an **adjective phrase** or **AP**. APs can have an adverb before the adjective (see Chapter 8 for definitions of adjectives and adverbs):

the *least expensive* fare

We can combine all the options for prenominal modifiers with one rule as follows:

$$NP \rightarrow (Det)\ (Card)\ (Ord)\ (Quant)\ (AP)\ Nominal \qquad (9.10)$$

This simplified noun phrase rule has a flatter structure and hence is simpler than most modern theories of grammar. We present this simplified rule because there is no universally agreed-upon internal constituency for the noun phrase.

Note the use of parentheses () to mark **optional constituents**. A rule with one set of parentheses is really a shorthand for two rules, one with the parentheses, one without.

After the Noun

A head noun can be followed by **postmodifiers**. Three kinds of nominal postmodifiers are very common in English:

prepositional phrases	all flights *from Cleveland*
non-finite clauses	any flights *arriving after eleven a.m.*
relative clauses	a flight *that serves breakfast*

Prepositional phrase postmodifiers are particularly common in the ATIS corpus, since they are used to mark the origin and destination of flights. Here are some examples, with brackets inserted to show the boundaries of each PP; note that more than one PP can be strung together:

any stopovers *[for Delta seven fifty one]*
all flights *[from Cleveland] [to Newark]*
arrival *[in San Jose] [before seven p.m]*
a reservation *[on flight six oh six] [from Tampa] [to Montreal]*

Here's a new *NP* rule to account for one to three *PP* postmodifiers:

$$Nominal \rightarrow Nominal\ PP\ (PP)\ (PP)$$

The three most common kinds of **non-finite** postmodifiers are the gerundive (*-ing*), *-ed*, and infinitive forms.

NON-FINITE

Gerundive postmodifiers are so-called because they consist of a verb phrase that begins with the gerundive (*-ing*) form of the verb. In the following examples, the verb phrases happen to all have only prepositional phrases after the verb, but in general this verb phrase can have anything in it (anything, that is, which is semantically and syntactically compatible with the gerund verb).

GERUNDIVE

any of those *(leaving on Thursday)*
any flights *(arriving after eleven a.m)*
flights *(arriving within thirty minutes of each other)*

We can define the NP as follows, making use of a new nonterminal *GerundVP*:

$$Nominal \rightarrow Nominal\ GerundVP$$

We can make rules for *GerundVP* constituents by duplicating all of our VP productions, substituting *GerundV* for *V*.

$$\begin{array}{lcl} \textit{GerundVP} & \rightarrow & \textit{GerundV NP} \\ & | & \textit{GerundV PP} \\ & | & \textit{GerundV} \\ & | & \textit{GerundV NP PP} \end{array}$$

GerundV can then be defined as:

$$\textit{GerundV} \rightarrow \textit{being} \mid \text{prefering} \mid \textit{arriving} \mid \textit{leaving} \mid \ldots$$

The phrases in italics below are examples of the two other common kinds of non-finite clauses, infinitives and *-ed* forms:

the last flight *to arrive in Boston*
I need to have dinner *served*
Which is the aircraft *used by this flight*?

RELATIVE PRONOUN

A postnominal relative clause (more correctly a **restrictive relative clause**), is a clause that often begins with a **relative pronoun** (*that* and *who* are the most common). The relative pronoun functions as the subject of the embedded verb in the following examples:

a flight *that serves breakfast*
flights *that leave in the morning*
the United flight *that arrives in San Jose around ten p.m.*
the one *that leaves at ten thirty five*

We might add rules like the following to deal with these:

$$\textit{Nominal} \rightarrow \textit{Nominal RelClause} \tag{9.11}$$

$$\textit{RelClause} \rightarrow (\textit{who} \mid \textit{that})\, \textit{VP} \tag{9.12}$$

$$\tag{9.13}$$

The relative pronoun may also function as the object of the embedded verb, as in the following example; we leave as an exercise for the reader writing grammar rules for more complex relative clauses of this kind.

the earliest American Airlines flight that I can get

Various postnominal modifiers can be combined, as the following examples show:

a flight *(from Phoenix to Detroit) (leaving Monday evening)*
I need a flight *(to Seattle) (leaving from Baltimore) (making a stop in Minneapolis)*
evening flights *(from Nashville to Houston) (that serve dinner)*
a friend *(living in Denver) (that would like to visit me here in Washington DC)*

9.5 COORDINATION

Noun phrases and other units can be **conjoined** with **conjunctions** like *and*, *or*, and *but*. For example a **coordinate** noun phrase can consist of two other noun phrases separated by a conjunction (we used brackets to mark the constituents):

CONJUNCTIONS
COORDINATE

Please repeat [$_{NP}$ [$_{NP}$ the flights] *and* [$_{NP}$ the costs]]
I need to know [$_{NP}$ [$_{NP}$ the aircraft] *and* [$_{NP}$ flight number]]
I would like to fly from Denver stopping in [$_{NP}$ [$_{NP}$ Pittsburgh] *and* [$_{NP}$ Atlanta]]

Here's a new rule for this:

$$NP \rightarrow NP\ and\ NP \tag{9.14}$$

In addition to NPs, most other kinds of phrases can be conjoined (for example including sentences, VPs, and PPs):

What flights do you have [$_{VP}$ [$_{VP}$ leaving Denver] *and* [$_{VP}$ arriving in San Francisco]]
[$_{S}$ [$_{S}$ I'm interested in a flight from Dallas to Washington] *and* [$_{S}$ I'm also interested in going to Baltimore]]

Similar conjunction rules can be built for *VP* and *S* conjunction:

$$VP \rightarrow VP\ and\ VP \tag{9.15}$$

$$S \rightarrow S\ and\ S \tag{9.16}$$

9.6 AGREEMENT

In Chapter 3 we discussed English inflectional morphology. Recall that most verbs in English can appear in two forms in the present tense: the form used for third-person, singular subjects (*the flight* does), and the form used for all other kinds of subjects (*all the flights* do, *I* do). The third-person-singular (*3sg* form usually has a final *-s* where the non-3sg form does not. Here are some examples, again using the verb *do*, with various subjects:

You [$_{VP}$ [$_V$ said [$_S$ there were two flights that were the cheapest]]]
Do [$_{NP}$ any flights] stop in Chicago?
Do [$_{NP}$ all of these flights] offer first class service?
Do [$_{NP}$ I] get dinner on this flight?
Do [$_{NP}$ you] have a flight from Boston to Forth Worth?
Does [$_{NP}$ this flight] stop in Dallas?
Does [$_{NP}$ that flight] serve dinner?
Does [$_{NP}$ Delta] fly from Atlanta to San Francisco?

Here are more examples with the verb *leave*:

What flights *leave* in the morning?
What flight *leaves* from Pittsburgh?

This agreement phenomenon occurs whenever there is a verb that has some noun acting as its subject. Note that sentences in which the subject does not agree with the verb are ungrammatical:

*[What flight] *leave* in the morning?
*Does [$_{NP}$ you] have a flight from Boston to Forth Worth?
*Do [$_{NP}$ this flight] stop in Dallas?

How can we modify our grammar to handle these agreement phenomena? One way is to expand our grammar with multiple sets of rules, one rule set for *3sg* subjects, and one for non-*3sg* subjects. For example, the rule that handled these yes-no-questions used to look like this:

$$S \rightarrow Aux\ NP\ VP$$

We could replace this with two rules of the following form:

$$S \rightarrow 3sgAux\ 3sgNP\ VP$$
$$S \rightarrow Non3sgAux\ Non3sgNP\ VP$$

We could then add rules for the lexicon like these:

3sgAux → *does* | *has* | *can* | ...
Non3sgAux → *do* | *have* | *can* | ...

But we would also need to add rules for *3sgNP* and *Non3sgNP*, again by making two copies of each rule for *NP*. While pronouns can be first, second, or third person, full lexical noun phrases can only be third person, so for them we just need to distinguish between singular and plural:

3SgNP → *(Det) (Card) (Ord) (Quant) (AP) SgNominal*
Non3SgNP → *(Det) (Card) (Ord) (Quant) (AP) PlNominal*
SgNominal → *SgNoun* | *SgNoun SgNoun*
PlNominal → *PlNoun* | *SgNoun PlNoun*
SgNoun → *flight* | *fare* | *dollar* | *reservation* | ...
PlNoun → *flights* | *fares* | *dollars* | *reservations* | ...

Dealing with the first and second person pronouns is left as an exercise for the reader.

A problem with this method of dealing with number agreement is that it doubles the size of the grammar. Every rule that refers to a noun or a verb needs to have a 'singular' version and a 'plural' version. This rule proliferation will also have to happen for the noun's **case**; for example English pronouns have **nominative** (*I, she, he, they*) and **accusative** (*me, her, him, them*) versions. We will need new versions of every *NP* and *N* rule for each of these.

CASE
NOMINATIVE
ACCUSATIVE

A more significant problem occurs in languages like German or French, which not only have noun-verb agreement like English, but also have **gender agreement**; the gender of a noun must agree with the gender of its modifying adjective and determiner. This adds another multiplier to the rule sets of the language.

GENDER AGREEMENT

Chapter 11 will introduce a way to deal with these agreement problems without exploding the size of the grammar, by effectively **parameterizing** each nonterminal of the grammar with **feature structures**.

9.7 THE VERB PHRASE AND SUBCATEGORIZATION

The verb phrase consists of the verb and a number of other constituents. In the simple rules we have built so far, these other constituents include *NP*'s and *PP*'s and combinations of the two:

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

SENTENTIAL COMPLEMENTS

Verb phrases can be significantly more complicated than this. Many other kinds of constituents can follow the verb, such as an entire embedded sentence. These are called **sentential complements**:

You [$_{VP}$ [$_{V}$ said [$_{S}$ there were two flights that were the cheapest]]]
You [$_{VP}$ [$_{V}$ said [$_{S}$ you had a two hundred sixty six dollar fare]]
[$_{VP}$ [$_{V}$ Tell] [$_{NP}$ me] [$_{S}$ how to get from the airport in Philadelphia to downtown]]
I [$_{VP}$ [$_{V}$ think [$_{S}$ I would like to take the nine thirty flight]]

Here's a rule for these:

VP → *Verb S*

Another potential constituent of the VP is another VP. This is often the case for verbs like *want*, *would like*, *try*, *intend*, *need*:

I want [$_{VP}$ to fly from Milwaukee to Orlando]
Hi, I want [$_{VP}$ to arrange three flights]
Hello, I'm trying [$_{VP}$ to find a flight that goes from Pittsburgh to Denver after two PM

Recall from Chapter 8 that verbs can also be followed by *particles*, words that resemble a preposition but that combine with the verb to form a *phrasal verb* like *take off*). These particles are generally considered to be an integral part of the verb in a way that other post-verbal elements are not; phrasal verbs are treated as individual verbs composed of two words.

While a verb phrase can have many possible kinds of constituents, not every verb is compatible with every verb phrase. For example, the verb *want* can either be used with an NP complement (*I want a flight...*), or with an infinitive VP complement (*I want to fly to...*). By contrast, a verb like *find* cannot take this sort of VP complement. (* *I found to fly to Dallas*).

This idea that verbs are compatible with different kinds of complements is a very old one; traditional grammar distinguishes between **transitive** verbs like *find*, which take a direct object NP (*I found a flight*), and **intransitive** verbs like *disappear*, which do not (**I disappeared a flight*).

TRANSITIVE

INTRANSITIVE

Where traditional grammars **subcategorize** verbs into these two categories (transitive and intransitive), modern grammars distinguish as many as 100 subcategories. (In fact tagsets for many such **subcategorization frames** exists; see (Macleod *et al.*, 1998) for the COMLEX tagset, Sanfilippo (1993) for the ACQUILEX tagset, and further discussion in Chapter 11). We say that a verb like *find* **subcategorizes for** an *NP*, while a verb like *want* subcategorizes for either an *NP* or an infinite *VP*. We also call these constituents the **complements** of the verb (hence our use of the term **sentential complement** above). So we say that *want* can take a *VP* complement. These possible sets of complements are called the **subcategorization frame** for the verb. Another way of talking about the relation between the verb and these other constituents is to think of the verb as a predicate and the constituents as arguments of the predicate. So we can think of such predicate-argument relations as FIND (I, A FLIGHT), or WANT (I, TO FLY). We will talk more about this view of verbs and arguments in Chapter 14 when we talk about predicate calculus representations of verb semantics.

SUBCATEGORIZE

SUBCATEGORIZATION FRAMES

SUBCATEGORIZES FOR

COMPLEMENTS

SUBCATEGORIZATION FRAME

Here are some subcategorization frames and example verbs:

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

Note that a verb can subcategorize for a particular type of verb phrase, such as a verb phrase whose verb is an infinitive (*VPto*), or a verb phrase whose verb is a bare stem (uninflected: *VPbrst*). Note also that a single verb can take different subcategorization frames. The verb *find*, for example, can take an *NP NP* frame (*find me a flight*) as well as an *NP* frame.

How can we represent the relation between verbs and their complements in a context-free grammar? One thing we could do is to do what we did with agreement features: make separate subtypes of the class Verb (*Verb-with-NP-complement Verb-with-Inf-VP-complement Verb-with-S-complement*

Verb-with-NP-plus-PP-complement, and so on):

$$
\begin{aligned}
\textit{Verb-with-NP-complement} &\rightarrow \textit{find} \mid \textit{leave} \mid \textit{repeat} \mid \ldots \\
\textit{Verb-with-S-complement} &\rightarrow \textit{think} \mid \textit{believe} \mid \textit{say} \mid \ldots \\
\textit{Verb-with-Inf-VP-complement} &\rightarrow \textit{want} \mid \textit{try} \mid \textit{need} \mid \ldots
\end{aligned}
$$

Then each of our *VP* rules could be modified to require the appropriate verb subtype:

VP → *Verb-with-no-complement* disappear
VP → *Verb-with-NP-comp NP* prefer a morning flight
VP → *Verb-with-S-comp S* said there were two flights

The problem with this approach, as with the same solution to the agreement feature problem, is a vast explosion in the number of rules. The standard solution to both of these problems is the **feature structure**, which will be introduced in Chapter 11. Chapter 11 will also discuss the fact that nouns, adjectives, and prepositions can subcategorize for complements just as verbs can.

9.8 Auxiliaries

AUXILIARIES MODAL PERFECT PROGRESSIVE PASSIVE

The subclass of verbs called **auxiliaries** or **helping verbs** have particular syntactic constraints which can be viewed as a kind of subcategorization. Auxiliaries include the **modal** verbs *can, could, may, might, must, will, would, shall*, and *should*, the **perfect** auxiliary *have*, the **progressive** auxiliary *be*, and the **passive** auxiliary *be*. Each of these verbs places a constraint on the form of the following verb, and each of these must also combine in a particular order.

Modal verbs subcategorize for a VP whose head verb is a bare stem, e.g. *can go in the morning*, *will try to find a flight*. The perfect verb *have* subcategorizes for a VP whose head verb is the past participle form: *have booked 3 flights*. The progressive verb *be* subcategorizes for a VP whose head verb is the gerundive participle: *am going from Atlanta*. The passive verb *be* subcategorizes for a VP whose head verb is the past participle: *was delayed by inclement weather*.

A sentence can have multiple auxiliary verbs, but they must occur in a particular order:

modal < perfect < progressive < passive

Here are some examples of multiple auxiliaries:

modal perfect	could have been a contender
modal passive	will be married
perfect progressive	have been feasting
modal perfect passive	might have been prevented

Auxiliaries are often treated just like verbs such as *want*, *seem*, or *intend*, which subcategorize for particular kinds of VP complements. Thus *can* would be listed in the lexicon as a *verb-with-bare-stem-VP-complement*. One way of capturing the ordering constraints among auxiliaries, commonly used in the **systemic grammar** of Halliday (1985a), is to introduce a special constituent called the **verb group**, whose subconstituents include all the auxiliaries as well as the main verb. Some of the ordering constraints can also be captured in a different way. Since modals, for example, do not having a progressive or participle form, they simply will never be allow to follow progressive or passive *be* or perfect *have*. Exercise 9.8 asks the reader to write grammar rules for auxiliaries.

SYSTEMIC GRAMMAR

VERB GROUP

The passive construction has a number of properties that make it different than other auxiliaries. One important difference is a semantic one; while the subject of non-passive (**active**) sentence is often the semantic agent of the event described by the verb (*I prevented a catastrophe*) the subject of the passive is often the undergoer or patient of the event (*a catastrophe was prevented*). This will be discussed further in Chapter 15.

ACTIVE

9.9 Spoken Language Syntax

The grammar of written English and the grammar of conversational spoken English share many features, but also differ in a number of respects. This section gives a quick sketch of a number of the characteristics of the syntax of spoken English.

We usually use the term **utterance** rather than **sentence** for the units of spoken language. Figure 9.5 shows some sample spoken ATIS utterances that exhibit many aspects of spoken language grammar.

UTTERANCE

This is a standard style of transcription used in transcribing speech corpora for speech recognition. The comma ‘,’ marks a short pause, each

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

Figure 9.5 Some sample spoken utterances from users interacting with the ATIS system.

period '.' marks a long pause, and the square brackets '[uh]' mark non-verbal events (breaths, lipsmacks, uhs and ums).

There are a number of ways these utterances differ from written English sentences. One is in the lexical statistics; for example spoken English is much higher in pronouns than written English; the subject of a spoken sentence is almost invariably a pronoun. Another is in the presence of various kinds of disfluencies (hesitations, repairs, restarts, etc) to be discussed below. Spoken sentences often consist of short fragments or phrases (*one way* or *around four PM*, which are less common in written English.

PROSODY

PITCH CONTOUR

STRESS PATTERN

Finally, these sentences were spoken with a particular **prosody**. The prosody of an utterance includes its particular **pitch contour** (the rise and fall of the fundamental frequency of the soundwave), its **stress pattern** or rhythm (the series of stressed and unstressed syllables that make up a sentence) and other similar factors like the rate (speed) of speech.

Disfluencies

DISFLUENCIES

Perhaps the most salient syntactic feature that distinguishes spoken and written language is the class of phenomena known as **disfluencies**. Disfluencies include the use of *uh* and *um*, word repetitions, and false starts. The ATIS sentence in Figure 9.6 shows examples of a false start and the use of *uh*. The false start here occurs when the speaker starts by asking for *one-way flights*.

and then stops and corrects herself, beginning again and asking about *one-way fares*.

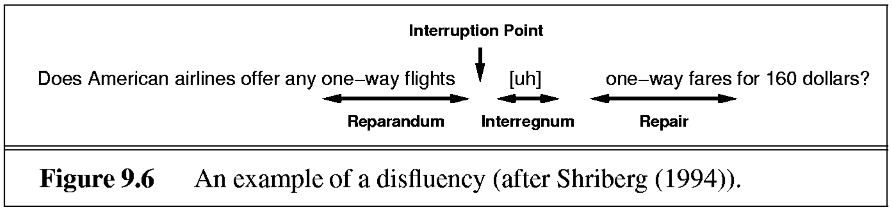

Interruption Point
Does American airlines offer any one–way flights
[uh]
one–way fares for 160 dollars?
Reparandum
Interregnum
Repair

Figure 9.6 An example of a disfluency (after Shriberg (1994)).

REPARANDUM

REPAIR

INTERRUPTION POINT

The segment *one-way flights* is referred to as the **reparandum**, and the replacing sequence *one-way fares* is referred to as the **repair** (these terms are from Levelt (1983)). The **interruption point**, where the speaker breaks off the original word sequence, here occurs right after the word *'flights'*.

FILLED PAUSES

The words *uh* and *um* (sometimes called **filled pauses**) can be treated in the lexicon like regular words, and indeed this is often how they are modeled in speech recognition. The HMM pronunciation lexicons in speech recognizers often include pronunciation models of these words, and the N-gram grammar used by recognizers include the probabilities of these occurring with other words.

For speech understanding, where our goal is to build a meaning for the input sentence, it may be useful to detect these restarts in order to edit out what the speaker probably considered the 'corrected' words. For example in the sentence above, if we could detect that there was a restart, we could just delete the reparandum, and parse the remaining parts of the sentence:

> Does American airlines offer any one-way flights uh one-way fares for 160 dollars?

How do disfluencies interact with the constituent structure of the sentence? Hindle (1983) showed that the repair often has the same structure as the constituent just before the interruption point. Thus in the example above, the repair is a PP, as is the reparandum. This means that if it is possible to automatically find the interruption point, it is also often possible to automatically detect the boundaries of the reparandum.

9.10 Grammar Equivalence & Normal Form

A formal language is defined as a (possibly infinite) set of strings of words. This suggests that we could ask if two grammars are equivalent by asking if they generate the same set of strings. In fact it is possible to have two distinct context-free grammars generate the same language.

We usually distinguish two kinds of grammar equivalence: **weak equivalence** and **strong equivalence**. Two grammars are strongly equivalent if they generate the same set of string *and* if they assign the same phrase structure to each sentence (allowing merely for renaming of the non-terminal symbols). Two grammars are weakly equivalent if they generate the same set of strings but do not assign the same phrase structure to each sentence.

NORMAL FORM

CHOMSKY NORMAL FORM

It is sometimes useful to have a **normal form** for grammars, in which each of the productions takes a particular form. For example a context-free grammar is in **Chomsky normal form** (CNF) (Chomsky, 1963) if it is ε-free and if in addition each production is either of the form $A \rightarrow B\ C$ or $A \rightarrow a$. That is, the righthand side of each rule either has two non-terminal symbols or one terminal symbol. Chomsky normal form grammars have binary trees (down to the prelexical nodes), which can be useful for certain algorithms.

Any grammar can be converted into a weakly-equivalent Chomsky normal form grammar. For example a rule of the form

$$A \rightarrow B\ C\ D$$

can be converted into the following two CNF rules:

$$A \rightarrow B\ X$$

$$X \rightarrow C\ D$$

Exercise 9.11 asks the reader to formulate the complete algorithm.

9.11 Finite State & Context-Free Grammars

RECURSION

We argued in Section 9.1 that a complex model of grammar would have to represent constituency. This is one reason that finite-state models of grammar are often inadequate. Now that we have explored some of the details of the syntax of noun phrases, we are prepared to discuss another problem with finite-state grammars. This problem is **recursion**. Recursion in a grammar occurs when an expansion of a non-terminal includes the non-terminal itself, as we saw in rules like *Nominal* → *Nominal PP* in the previous section.

In order to see why this is a problem for finite-state grammars, let's first attempt to build a finite-state model for some of the grammar rules we have seen so far. For example, we could model the noun phrase up to the head with a regular expression (= FSA) as follows:

(Det) (Card) (Ord) (Quant) (AP) Nominal

What about the postmodifiers? Let's just try adding the *PP*. We could then augment the regular expression as follows:

*(Det) (Card) (Ord) (Quant) (AP) Nominal (PP)**

So to complete this regular expression we just need to expand inline the definition of *PP*, as follows:

*(Det) (Card) (Ord) (Quant) (AP) Nominal (P NP)**

But wait; our definition of *NP* now presupposes an *NP*! We would need to expand the rule as follows:

*(Det) (Card) (Ord) (Quant) (AP) Nominal (P (Det) (Card) (Ord) (Quant) (AP) Nominal (P NP))**

But of course the *NP* is back again! The problem is that NP is a **recursive rule**. There is actually a sneaky way to 'unwind' this particular **right-recursive** rule in a finite-state automaton. In general, however, recursion cannot be handled in finite automata, and recursion is quite common in a complete model of the *NP* (for example for *RelClause* and *GerundVP*, which also have *NP* in their expansion):

RECURSIVE RULE

*(Det) (Card) (Ord) (Quant) (AP) Nominal (RelClause|GerundVP|PP)**

In particular, Chomsky (1959a) proved that a context-free language L can be generated by a finite automaton if and only if there is a context-free grammar that generates L that does not have any **center-embedded** recursions (recursions of the form $A \rightarrow \alpha A \beta$).

While it thus seems at least likely that we can't model all of English syntax with a finite state grammar, it is possible to build an FSA that approximates English (for example by expanding only a certain number of *NP*s). In fact there are algorithms for automatically generating finite-state grammars that approximate context-free grammars (Pereira and Wright, 1997).

Chapter 10 will discuss an augmented version of the finite-state automata called the **recursive transition network** or **RTN** that adds the complete power of recursion to the FSA. The resulting machine is exactly isomorphic to the context-free grammar, and can be a useful metaphor for studying CFGs in certain circumstances.

9.12 GRAMMARS & HUMAN PROCESSING

Do people use context-free grammars in their mental processing of language? It has proved very difficult to find clear-cut evidence that they do. For example, some early experiments asked subjects to judge which words in a sentence were more closely connected (Levelt, 1970), finding that their intuitive group corresponded to syntactic constituents. Other experimenters examined the role of constituents in auditory comprehension by having subjects listen to sentences while also listening to short "clicks" at different times. Fodor and Bever (1965) found that subjects often mis-heard the clicks as if they occurred at constituent boundaries. They argued that the constituent was thus a 'perceptual unit' which resisted interruption. Unfortunately there were severe methodological problems with the click paradigm (see for example Clark and Clark (1977) for a discussion).

A broader problem with all these early studies is that they do not control for the fact that constituents are often semantic units as well as syntactic units. Thus, as will be discussed further in Chapter 15, *a single odd block* is a constituent (an *NP*) but also a semantic unit (an object of type BLOCK which has certain properties). Thus experiments which show that people notice the boundaries of constituents could simply be measuring a semantic rather than a syntactic fact.

Thus it is necessary to find evidence for a constituent which is *not* a semantic unit. Furthermore, since there are many non-constituent-based theories of grammar based on lexical dependencies, it is important to find evidence that cannot be interpreted as a *lexical* fact; i.e. evidence for constituency that is not based on particular words.

One suggestive series of experiments arguing for constituency has come from Kathryn Bock and her colleagues. Bock and Loebell (1990), for example, avoided all these earlier pitfalls by studying whether a subject who uses a particular syntactic constituent (for example a verb-phrase of a particular type, like *V NP PP*), is more likely to use the constituent in following sentences. In other words, they asked whether use of a constituent *primes* its use in subsequent sentences. As we saw in previous chapters, priming is a common way to test for the existence of a mental structure. Bock and Loebell relied on the English **ditransitive alternation**. A ditransitive verb is one like *give* which can take two arguments:

(9.17) The wealthy widow gave [$_{NP}$ the church] [$_{NP}$ her Mercedes].

The verb *give* allows another possible subcategorization frame, called a **prepositional dative** in which the indirect object is expressed as a prepositional phrase:

(9.18) The wealthy widow gave [$_{NP}$ her Mercedes] [$_{PP}$ to the church].

As we discussed on page 339, many verbs other than *give* have such **alternations** (*send*, *sell*, etc; see Levin (1993) for a summary of many different alternation patterns). Bock and Loebell relied on these alternations by giving subjects a picture, and asking them to describe it in one sentence. The picture was designed to elicit verbs like *give* or *sell* by showing an event such as a boy handing an apple to a teacher. Since these verbs alternate, subjects might, for example, say *The boy gave the apple to the teacher* or *The boy gave the teacher an apple.*

ALTERNATIONS

Before describing the picture, subjects were asked to read an unrelated 'priming' sentence out loud; the priming sentences either had *V NP NP* or *V NP PP* structure. Crucially, while these priming sentences had the same *constituent structure* as the dative alternation sentences, they did not have the same *semantics*. For example, the priming sentences might be prepositional *locatives*, rather than *datives*:

(9.19) IBM moved [$_{NP}$ a bigger computer] [$_{PP}$ to the Sears store].

Bock and Loebell found that subjects who had just read a *V NP PP* sentence were more like to use a *V NP PP* structure in describing the picture. This suggested that the use of a particular constituent *primed* the later use of that constituent, and hence that the constituent must be mentally represented in order to prime and be primed.

In more recent work, Bock and her colleagues have continued to find evidence for this kind of constituency structure.

There is a quite different disagreement about the human use of context-free grammars. Many researchers have suggested that natural language is unlike a formal language, and in particular that the set of possible sentences in a language cannot be described by purely syntactic context-free grammar productions. They argue that a complete model of syntactic structure will prove to be impossible unless it includes knowledge from other domains (for example like semantic, intonational, pragmatic, and social/interactional domains). Others argue that the syntax of natural language can be represented by formal languages. This second position is called **modularist**: researchers holding this position argue that human syntactic knowledge is a distinct module of the human mind. The first position, in which grammatical knowledge may incorporate semantic, pragmatic, and other constraints, is called **anti-modularist**. We will return to this debate in Chapter 15.

MODULARIST

ANTI-MODULARIST

9.13 Summary

This chapter has introduced a number of fundamental concepts in syntax via the **context-free grammar**.

- In many languages, groups of consecutive words act as a group or a **constituent**, which can be modeled by **context-free grammars** (also known as **phrase-structure grammars**.
- A context-free grammar consists of a set of **rules** or **productions**, expressed over a set of **non-terminal** symbols and a set of **terminal** symbols. Formally, a particular **context-free language** is the set of strings which can be **derived** from a particular **context-free grammar**.
- A **generative grammar** is a traditional name in linguistics for a formal language which is used to model the grammar of a natural language.
- There are many sentence-level grammatical constructions in English; **declarative**, **imperative**, **yes-no-question**, and **wh-question** are four very common types, which can be modeled with context-free rules.
- An English **noun phrase** can have **determiners**, **numbers**, **quantifiers**, and **adjective phrases** preceding the **head noun**, which can be followed by a number of **postmodifiers**; **gerundive** VPs, **infinitives** VPs, and **past participial** VPs are common possibilities.
- **Subjects** in English **agree** with the main verb in person and number.
- Verbs can be **subcategorized** by the types of **complements** they expect. Simple subcategories are **transitive** and **intransitive**; most grammars include many more categories than these.
- The correlate of **sentences** in spoken language are generally called **utterances**. Utterances may be **disfluent**, containing **filled pauses** like *um* and *uh*, **restarts**, and **repairs**.
- Any context-free grammar can be converted to **Chomsky normal form**, in which the right-hand-side of each rule has either two non-terminals or a single terminal.
- Context-free grammars are more powerful than finite-state automata, but it is nonetheless possible to **approximate** a context-free grammar with a FSA.
- There is some evidence that constituency plays a role in the human processing of language.

Bibliographical and Historical Notes

> “den sprachlichen Ausdruck für die willkürliche Gliederung einer Gesammtvorstellung in ihre in logische Beziehung zueinander gesetzten Bestandteile”
> “the linguistic expression for the arbitrary division of a total idea into its constituent parts placed in logical relations to one another”
>
> Wundt’s (1900:240) definition of the sentence; the origin of the idea of phrasal constituency, cited in Percival (1976).

The recent historical research of Percival (1976) has made it clear that this idea of breaking up a sentence into a hierarchy of constituents appeared in the *Völkerpsychologie* of the groundbreaking psychologist Wilhelm Wundt (Wundt, 1900). By contrast, traditional European grammar, dating from the Classical period, defined relations between *words* rather than constituents. Wundt’s idea of constituency was taken up into linguistics by Leonard Bloomfield in his early book *An Introduction to the Study of Language* (Bloomfield, 1914). By the time of his later book *Language* (Bloomfield, 1933), what was then called ‘immediate-constituent analysis’ was a well-established method of syntactic study in the United States. By contrast, European syntacticians retained an emphasis on word-based or **dependency** grammars; Chapter 12 discusses some of these issues in introducing dependency grammar.

American Structuralism saw a number of specific definitions of the immediate constituent, couched in terms of their search for a ‘discovery procedure’; a methodological algorithm for describing the syntax of a language. In general, these attempt to capture the intuition that “The primary criterion of the immediate constituent is the degree in which combinations behave as simple units” (Bazell, 1952, p. 284). The most well-known of the specific definitions is Harris’ idea of distributional similarity to individual units, with the *substitutability* test. Essentially, the method proceeded by breaking up a construction into constituents by attempting to substitute simple structures for possible constituents — if a substitution of a simple form, say *man*, was substitutable in a construction for a more complex set (like *intense young man*), then the form *intense young man* was probably a constituent. Harris’s test was the beginning of the intuition that a constituent is a kind of equivalence class.

The first formalization of this idea of hierarchical constituency was the **phrase-structure grammar** defined in Chomsky (1956), and further expanded upon (and argued against) in Chomsky (1957) and Chomsky (1975). From this time on, most generative linguistic theories were based at least in part on context-free grammars (such as Head-Driven Phrase Structure Grammar (Pollard and Sag, 1994), Lexical-Functional Grammar (Bresnan, 1982), Government and Binding (Chomsky, 1981), and Construction Grammar (Kay and Fillmore, 1999), *inter alia*); many of these theories used schematic context-free templates known as **X-bar schemata**.

X-BAR SCHEMATA

Shortly after Chomsky's initial work, the context-free grammar was rediscovered by Backus (1959) and independently by Naur *et al.* (1960) in their descriptions of the ALGOL programming language; Backus (1996) noted that he was influenced by the productions of Emil Post and that Naur's work was independent of his (Backus') own. After this early work, a great number of computational models of natural language processing were based on context-free grammars because of the early development of efficient algorithms to parse these grammars (see Chapter 10).

As we have already noted, grammars based on context-free rules are not ubiquitous. One extended formalism is Tree Adjoining Grammar (TAG) (Joshi, 1985). The primary data structure in Tree Adjoining Grammar is the tree, rather than the rule. Trees come in two kinds; **initial trees** and **auxiliary trees**. Initial trees might, for example, represent simple sentential structures, while auxiliary trees are used to add recursion into a tree. Trees are combined by two operations called **substitution** and **adjunction**. See Joshi (1985) for more details. An extension of Tree Adjoining Grammar called Lexicalized Tree Adjoining Grammars will be discussed in Chapter 12.

Another class of grammatical theories that are not based on context-free grammars are instead based on the relation between words rather than constituents. Various such theories have come to be known as **dependency grammars**; representative examples include the dependency grammar of Mel'čuk (1979), the Word Grammar of Hudson (1984), or the Constraint Grammar of Karlsson *et al.* (1995). Dependency-based grammars have returned to popularity in modern statistical parsers, as the field have come to understand the crucial role of word-to-word relations; see Chapter 12 for further discussion.

Readers interested in general references grammars of English should waste no time in getting hold of Quirk *et al.* (1985a). Other useful treatments include McCawley (1998).

There are many good introductory textbook on syntax. Sag and Wasow (1999) is an introduction to **formal syntax**, focusing on the use of phrase-structure, unification, and the type-hierarchy in Head-Driven Phrase Structure Grammar. van Valin (1999) is an introduction from a less formal, more functional perspective, focusing on cross-linguistic data and on the functional motivation for syntactic structures.

FORMAL SYNTAX

EXERCISES

9.1 Draw tree structures for the following ATIS phrases:

a. Dallas
b. from Denver
c. after five p.m.
d. arriving in Washington
e. early flights
f. all redeye flights
g. on Thursday
h. a one-way fare
i. any delays in Denver

9.2 Draw tree structures for the following ATIS sentences:

a. Does American airlines have a flight between five a.m. and six a.m.
b. I would like to fly on American airlines.
c. Please repeat that.
d. Does American 487 have a first class section?
e. I need to fly between Philadelphia and Atlanta.
f. What is the fare from Atlanta to Denver?
g. Is there an American airlines flight from Philadelphia to Dallas?

9.3 Augment the grammar rules on page 337 to handle pronouns. Deal properly with person and case.

9.4 Modify the noun phrase grammar of Sections 9.4–9.6 to correctly model mass nouns and their agreement properties

9.5 How many types of NPs would rule (9.10) on page 332 expand to if we didn't allow parentheses in our grammar formalism?

9.6 Assume a grammar that has many VPs rules for different subcategorization, as expressed in Section 9.7, and differently subcategorized verb rules like *Verb-with-NP-complement*. How would the rule for post-nominal relative clauses (9.12) need to be modified if we wanted to deal properly with examples like *the earliest flight that you have*? Recall that in such examples the pronoun *that* is the object of the verb *get*. Your rules should allow this noun phrase but should correctly rule out the ungrammatical S **I get.*

9.7 Does your solution to the previous problem correctly model the NP *the earliest flight that I can get*? How about *the earliest flight that I think my mother wants me to book for her*? Hint: this phenomenon is called **long-distance dependency**.

9.8 Write rules expressing the verbal subcategory of English auxiliaries; for example you might have a rule *can → verb-with-bare-stem-VP-complement*.

POSSESSIVE
GENITIVE

9.9 NPs like *Fortune's office* or *my uncle's marks* are called **possessive** or **genitive** noun phrases. A possessive noun phrase can be modeled by treated the sub-NP like *Fortune's* or *my uncle's* as a determiner of the following head noun. Write grammar rules for English possessives. You may treat *'s* as if it were a separate word (i.e. as if there were always a space before *'s*).

9.10 Page 330 discussed the need for a *Wh-NP* constituent. The simplest Wh-NP is one of the wh-pronouns (*who, whom, whose, which*). The Wh-words , *what* and *which* can be determiners: *which four will you have?*, *what credit do you have with the Duke?*. Write rules for the different types of Wh-NPs.

9.11 Write an algorithm for converting an arbitrary context-free grammar into Chomsky normal form.