1 Linguistics: strings

1.1 STRING GRAMMARS

Throughout this book, I shall be using 'grammar' in the sense given to it by mathematicians. That calls for a word of explanation, for the terms 'grammar' and 'syntax' are in process of interchanging their meanings and the new usage has not yet crystallized, with consequent occasions of confusion.

The way in which we classify linguistic expressions and the way in which which we analyse them structurally depends upon our *purposes*. People's jobs, for example, can be distinguished into manual versus 'white-collar' or, alternatively, into those concerned with production and those concerned with providing services. Sometimes we need one classification, sometimes the other. Language is no exception. The branches of learning whose subject-matter is language already exhibit at least three distinct (though not unrelated) purposes. Oldest, perhaps, is the study of meaning and of validity in argument. Then came grammar (now, more often, called 'syntax'), the description, roughly at the level of words and phrases, of those combinations which are used in a given language. More recently, phonology and phonetics have sought to classify sounds and their combinations, from several points of view, including, for example, the relation of sound production to the physiology of the throat and mouth. There is no *a priori* reason to suppose that the same structural analysis will be apposite to all of these purposes. Indeed, quite the contrary, although, to the extent to which the purposes are related, it should be possible to inter-relate the corresponding structural systems. The contrast which is of prime concern here is that between the study of meaning, on the one hand, and that of the accepted forms of expression in a particular language, on the other.

Etymologically, 'syntax' simply means 'order', 'arrangement', 'organization', so that it is precisely the study of structure. Consequently, if we distinguish more than one structural system in language, we shall correspondingly have more than one syntax, for example a syntax

relating to meaning and another relating to the forms of expression which are accepted in a particular language. But recently it has become common to contrast syntax with semantics. In the mouths of linguists, this is a contrast with the study of meaning, and syntax is roughly what grammar used to be, a classification of the accepted forms of expression for a particular language, though largely omitting the morphology of individual words.

Meanwhile, grammar has been used by some philosophers for the combination of expressions in relation to their meanings, as in 'philosophical grammar'. Moreover, its application has been extended very considerably by mathematicians to embrace almost any system of rules for the combination of elements, not necessarily linguistic, into structures. An example is a (graph) grammar to describe the development of epidermal cell layers (Lindenmayer and Rosenberg, 1979); another is a (context-free) grammar for string descriptions of submedian and telocentric chromosomes (Ledley *et al.*, 1965). In the mathematician's usage, a grammar is always a formal system, whereas philosophical grammar is usually understood to be implicit, the unwritten rules for the combination of expressions with respect to their meanings; the philosopher's task is then to make as much of it explicit as his immediate purposes may require. This slight ambiguity is unavoidable when we are dealing with everyday language, for we are not free to specify what rules we please: we are constrained by language which is already present; yet, at the same time, we want to be as explicit as possible.

It seems too late to protest successfully against this reversal of the terminology, but it is with regret that I acquiesce in it. Most important, however, is to have a clear and stable usage. We do not have this at present, as the reversal is not yet complete, so that 'grammar' and 'syntax' are still sometimes used interchangeably, for instance 'universal grammar' instead of 'universal syntax'. We do need two distinct terms here, one for the study of linguistic structures in general, of whatever type, and another for the study of forms of expression which are accepted in a particular language. So, bowing to the new custom, I shall reserve 'grammar' for the former and 'syntax' for the latter.

To the extent that linguists have concerned themselves with specifying grammars formally, most of the grammars which they have proposed for everyday language have been among those known to mathematicians as *string* grammars. String grammars, naturally, generate strings, that is, symbols concatenated in a line, a well-ordering, so that we could identify each as the first, second, third, etc., symbol in the string. Alternatively, a string is an ordered *list*. It is also possible for a member of a string itself

to be a string, so that we obtain a structure of nested strings. So, if the grammar is used as a basis for representing meaning, there is an implicit claim that meaning can be represented adequately by string structures, that nothing more complicated is needed.

A string grammar G_S consists of an ordered set $\langle N,\Sigma,P,S\rangle$, where N is a finite set of non-terminals, Σ of terminals, P of productions or re-writing rules, and S is the starting symbol. Intuitively, terminals are expressions of the language which the grammar generates, non-terminals are category symbols, of which the starting symbol S is one. V, the union of N and Σ, is the *alphabet* of the grammar, while V^* is the closure of V, that is, the denumerably infinite set of all finite strings composed of members of V, but including the empty string (*excluding* the empty string, it is V^+). In general, productions take the form $\alpha \Rightarrow \beta$, meaning that α may be re-written as β, where α is in V^+ and β is in V^*.

Linguists have largely confined themselves to string grammars, of which a wide variety has now been proposed. Their interest, however, has primarily been in syntax, so we need only be concerned with these grammars to the extent that they have been expected to sustain an account of meaning. To this end, some exposition of formal syntax is unavoidable; yet, at the same time, a comprehensive survey of every theory, even from this point of view alone, would call for a book to itself. I propose, therefore, to concentrate upon the most famous and the most fully developed string grammar for everyday language, *transformational* grammar, due originally to Chomsky (1957), which from the start has sought to encompass an account of meaning as well.[1] I shall not, however, discuss the syntactic arguments used to support the proposed structural analyses. Moreover, linguists who favour a different theory of syntax will have to ask for themselves whether the points of criticism which I raise carry over to their preferred theory.

Transformational grammar grew out of constituent-structure analysis (Bloomfield, 1933; Wells, 1947; Harris, 1951; Postal, 1964). Sentences were first divided into their immediate constituents, typically phrases,

[1] My exposition is based on Borsley (1991) and Radford (1988), supplemented from Radford (1981), Jacobsen (1986) and Chomsky (1977, 1982a, 1982b, 1986a and 1986b). For the most recent version of the semantic component, May (1985) is the central text. To be exact, there have been *three* transformational grammars, the second dating from Chomsky (1965) and the third from Chomsky (1981). In each case continuity of notation has masked fundamental changes to the theory. Thus from 1965 to 1981 transformations were held to be meaning-preserving, with the semantic component operating upon deep structures, while since 1981 the semantic component has been attached to shallow structures, which are almost surface structures, and transformation rules merely move items within a structure derived from the phrase-structure rules. The revisions have been prompted in large measure by challenges from logic.

and then the latter were progressively divided until their ultimate constituents, typically single words, were reached. The criterion upon which the divisions were based was initially intuitive, but later a substitution test was introduced: if, for an expression occurring in a sentence others could be substituted whilst preserving grammaticality and if, further, the same substitutions were possible in any other sentence in which the expression could occur, then that expression was accounted a constituent of any sentence in which it occurred. The test was thus also a method of classifying linguistic expressions, the members of each group belonging to the same *syntactic category*.[2]

By using a symbol for each category, it became possible to describe a series of sentence-patterns. The categories (constituting N, the finite set of non-terminals) used in most formal theories today are derived from those of traditional syntax. While there remain some differences, there is also a broad measure of agreement, starting from Noun (N), Verb (V), Adjective (A) and Preposition (P). These are known as *lexical* categories; investigations of constituent structure revealed a need to classify corresponding phrases, so four *phrasal* categories were introduced as well. Subsequently arguments were put forward for intermediate categories to cater for expressions which were more than single words or morphemes, yet smaller than the phrases already recognized. The intermediate categories are commonly indicated by a prime, the phrasal categories by a double prime, for example N', N'' (or NP for the latter).

The original start symbol was S (Sentence) and there was no phrasal category corresponding to it. Subsequently it was itself recognized as a phrasal category, the head of such phrases being an Inflexion (I) catering for variations of tense, aspect and modality, so that S, accordingly, was replaced by I''. A further category C (Complementizer) was introduced later to provide for subordinate and relative clauses as well as for mood; thus, examples of complementizers are relative pronouns and 'that' introducing a subordinate clause. As a result of these developments, the start symbol in the latest version of the theory is C'' but, as I shall only be concerned with indicative sentences, I shall omit the first few steps in each derivation and begin with I''. This will make it easier to see the essential features of derivations for the purpose in hand.

Another development which can be largely ignored here is to break down the original categories into sets of *features*, each of which has a value. Thus verbs and prepositions share common behaviour which

[2] This test for syntactic constituents has now been supplemented by several more, which are clearly set out in Radford (1988, p. 90); see also Borsley (1991, ch. 2).

nouns and adjectives lack, such as being able to combine with a noun phrase, so are assumed to incorporate a feature, rather confusingly dubbed 'V'; they are $+V$, whereas nouns and adjectives are $-V$. Similarly, a feature 'N' is credited to nouns and prepositions but not to verbs and adjectives; while a further feature *BAR* (derived from an earlier notation) with 0, 1 and 2 as values can be used to differentiate lexical, intermediate and full phrasal categories respectively. The theoretical interest of this feature theory is as a basis for justifying the system of categories; but, so long as it is understood that these are syntactic categories, they are not our concern.

The form of production rules for a string grammar cited above, according to which a category symbol may be re-written as a string, combines two elements. First, there is the replacement of the category symbol by a list of symbols; second, there is the ordering of that list. So long as we use a linear notation to represent the structure of a string, these two elements are difficult to disentangle. Take, as an example, the sentence

(1) Every doctor visited at least one patient.

Let us suppose that 'every doctor' and 'at least one patient' are noun phrases, composed of an expression of a new category, *Det* (Determiner) ('every' and 'at least one') and a noun ('doctor', 'patient'), while 'visited at least one patient' is a verb phrase, composed of a verb 'visited' and the noun phrase 'at least one patient' already identified.

We can then represent a structure for (1) in a linear notation by

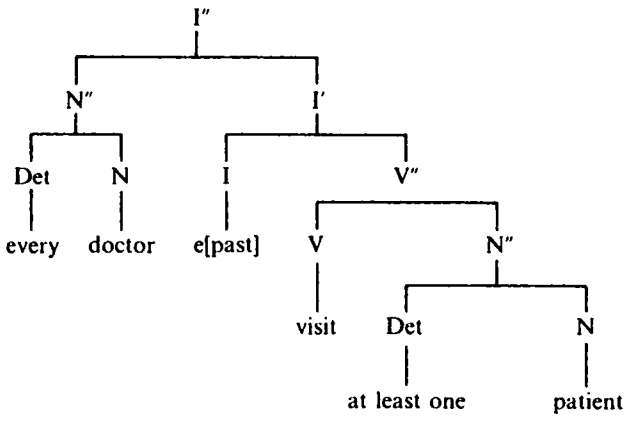
(1L) (Det N) (I (V (Det N))).

This is a string whose first member is a string of two members and whose second member is also a string of two members, with its second member being in turn a string of two members, and so once more. However, it omits any information about the categories of the sub-strings. We could supply this by placing sub-scripts on the closing brackets, as follows (intermediate categories are omitted in the interest of simplicity):

(1L') $((\text{Det N})_{N'}\ (\text{I}\ (\text{V}\ (\text{Det N})_{N'})_{V'}))_{I'}$.

This is known as a *labelled bracketing*, but it is a much less clear representation than the following tree, known as a *phrase-marker* (to which I have added the terminal symbols for (1)):

(1P)



The symbol *e* under the *I* node indicates that it is empty; tense, however, is regarded as a feature and is therefore shown in square brackets.

As we shall be dealing with trees a great deal in the sequel, a few terms will be useful. A tree consists of *nodes*, joined by *edges*. Both nodes and edges may be labelled; here only the nodes are labelled and are shown by their labels. The node at the top of the tree is called its *root*, whose label is always the starting symbol. The nodes at the bottom of the tree are called its *leaves*; here they are all terminal symbols. Phrase-markers are ordered from left to right and from top to bottom. The left-to-right ordering gives rise to a relationship of *precedence* between nodes: for any two distinct nodes X and Y, X precedes Y just in case X occurs to the left of Y. This relationship remains invariant in any given string, because edges in a phrase-marker may not cross. The top-to-bottom ordering produces a *dominance* relationship: X dominates Y just in case there is a path down the tree from X to Y; X *immediately* dominates Y just in case it dominates Y and no node occurs between them. Symbols which are immediately dominated by the same symbol are called *sisters*. Other relationships important in transformational grammar can wait until they are needed.

The first phrase-structure rules formulated by transformational grammarians combined immediate dominance and precedence, for example the following, in accordance with which we could obtain (1P):

S ⇒ N″ V″
N″ ⇒ Det N
V″ ⇒ V N″

Since then, they have undergone continuous change and refinement. First, immediate dominance (ID) and (linear) precedence (LP) have been separated; the motivation for this is that it is more economical when making cross-comparisons between different languages, which may differ

in their word-order conventions while nevertheless breaking down phrases into the same elements.

Second, the ID rules have now been reduced to three types, which can, accordingly, be stated as rule-*schemas*, that is to say, rule-patterns which use symbols for which the category names given above may be substituted in order to yield particular rules. I shall use Greek minuscules as such symbols; on the right-hand side of the rule they will be separated by commas, to indicate that their order is indifferent. Each of the four types of rule relates to a type of *modifier*, what is modified being the *head* of the construction (which I write first), and the four modifiers belong to the same group as the traditional terms subject, object, indirect object.

The first is that of *specifier*, which takes us from a full phrasal category to an intermediate category which is the head of the new string; the rule-schema is:

(S) $\alpha'' \Rightarrow \alpha', (\beta'')$.

The parentheses indicate that β'' is an optional element, so that α'' may simply be re-written by α'. If β'' is present, however, it is the specifier of α' and the latter its head. It is not possible to substitute just *any* category name for the Greek letters and still always obtain a syntactically correct rule, but some substitutions which work out all right for English are the following:

(SC) $C'' \Rightarrow C', (N'')$
(SI) $I'' \Rightarrow I', (N'')$
(SV) $V'' \Rightarrow V', (N'')$
(SN) $N'' \Rightarrow N', Det$
(SP) $P'' \Rightarrow P', Det$
(SA) $A'' \Rightarrow A', Det$

The parentheses indicate that the symbol enclosed in them is optional. Thus C'' may simply be re-written as C', without any branching, and similarly for I'' and V''. In such cases I shall often omit the intermediate category, in the interest of keeping phrase-markers as simple as possible. So far as linear precedence is concerned, in English the head always *follows* the specifier in applications of these rules.

The second type of modifier is a *complement*; the rule-schema for introducing complements takes us from an intermediate to a lexical category, the latter again the head of the new construction, and is:

(C) $\alpha' \Rightarrow \alpha, (\beta''), (\gamma''), (\ldots)$.

Thus a rule may introduce more than one complement. Some examples are:

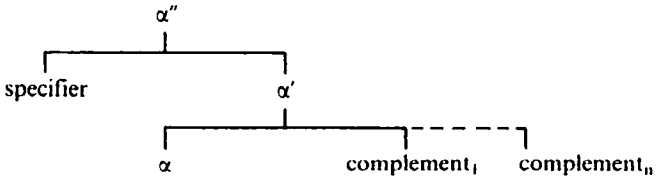
(CC) C′ ⇒ C, I″
(CI) I′ ⇒ I, V″
(CV) V′ ⇒ V, (N″), (A″), (P″), (I″)
(CN) N′ ⇒ N, (P″) / (I″)

(The slash in this rule indicates that P″ and I″ are exclusive alternatives).

(CP) P′ ⇒ P, (N″)
(CA) A′ ⇒ A, (N″), (P″)

Thus this type of rule allows us to re-introduce phrasal categories. The LP rules for English require that a lexical category or a noun phrase precedes any phrasal category which is its sister, and that a sentence (*I″*) follows all its sisters.

By using a specifier rule for a given category followed by a complement rule, we may descend from the double-barred category to the corresponding unbarred one, the derivation proceeding thus:



The third type of modifier is an *adjunct*, which is simply added to an intermediate category; thus the rule-schema is:

(A) α′ ⇒ α′, β″

The following are examples:

(AV) V′ ⇒ V′, A″ / N″ / P″
(AN) N′ ⇒ N′, N″ / A″ / P″ / I″
(AP) P′ ⇒ P′, A″ / P″
(XA) A′ ⇒ A′, A″ / P″

These rules are optional. The arguments for adjuncts from specifiers and complements are mainly syntactic and need not concern us here.

There is one type of construction for which the three rule-schemas described above do not provide, namely, that involving *coordinating conjunctions* such as 'and' and 'or'. According to transformational grammarians, coordinating conjunctions can be used to join together any two expressions of the *same* category; moreover, they can be used to form lists, so, if their category be *K*, we need a rule-schema on the following lines:

(K) $\alpha \Rightarrow K, \alpha_1, \alpha_2, (\alpha_3, \ldots, \alpha_n,)$

with the LP rule for English that *K* must occur in the penultimate position. A typical example is when a sentence is re-written as a disjunction of two sentences, i.e.

I″ ⇒ I″ or I″,

and another occurs when a complex subject is formed, such as 'Jack and Jill', which requires:

N″ ⇒ N″ and N″.

(I have ignored, here, the complication which arises when the conjunction as shown is the second part of a two-part expression, as in 'either . . . or' and 'both . . . and'.)

1.2 SEMANTIC RÔLES

So far, these rules will yield phrase-markers whose leaves are lexical category symbols, but they do not provide for the introduction of terminal symbols; hence the resulting structure is known as a *pre-terminal string*. In order to obtain a deep structure from a pre-terminal string, the non-terminals must be replaced by terminals. This is effected by *lexical insertion*, the substitution of words, phrases or other signs having a fixed meaning for the category symbols (including an empty or null symbol, ∅). At its simplest, this could be done by providing, for each category symbol, a list of linguistic expressions which might be substituted for it.

In practice, a more complicated arrangement is needed. Suppose, for example, that we had the pre-terminal string (1L′); clearly, a *transitive* verb must be substituted for *V*, because of the second *(Det N)*. By contrast, if the pre-terminal string had been

(Det N) (I V)

instead, an *intransitive* verb would have to be substituted for *V*. In view of this and many similar cases with other categories, linguistic expressions were *sub-categorized*, an idea due originally to Matthews and Chomsky (see Chomsky, 1965, p. 79), which was effected by showing what kinds of string they may fit into. These strings, in turn, were specified by recourse to the phrase-structure rules, since the pre-terminal strings may contain more detail (often variable) than is needed for the purpose. Thus a transitive verb like *visit* could be listed as V, +[— N″], where the brackets enclose a *sub-categorization frame* into which the expression would fit at the horizontal line, and the plus-sign indicates that the specified frame *must* be present.

A subsequent development, however, has now made it possible to eliminate sub-categorization in the lexicon (though some linguists prefer to retain it). This is *case grammar*,[3] which is concerned with the *semantic* relationships between verbs and their subjects, direct and indirect objects, etc. Borrowing from mathematics *via* logic, Fillmore calls these the *arguments* of a verb or predicate. Neither he nor other transformational grammarians, however, use the term precisely in its mathematico-logical sense, so it is better to define it, as they do, in terms of phrase-markers: an argument is any N'' which is dominated either by another N'' or by an I''.

These relationships between verbs and their arguments are expressed in everyday language, according to case-grammar, by cases, prepositions or postpositions, and may be characterized as *rôle-types*:

> human languages are constrained in such a way that the relations between arguments and predicates fall into a small number of types . . . these rôle types can be identified with certain quite elementary judgments about the things that go on around us: judgments about who does something, who experiences something, who benefits from something, where something happens, what it is that changes, what it is that moves, where it starts out, and where it ends up. (1968b, p. 38)

Fillmore eventually settled for nine of these rôles (1971, pp. 42, 50–1). He does not suppose a one-to-one correspondence between rôles and (in English) prepositions; a single rôle can be indicated, in different sentences, by more than one preposition and the same preposition may be used for more than one rôle. Typical examples of prepositions for each rôle, however, are:

Agent	'by'	Source	'from'	Location	'in'
Experiencer	'by'	Goal	'to'	Path	'along', 'through'
Object	–	Instrument	'with'	Time	'at', 'during'

There is no preposition for the object-rôle in English.[4]

These rôles are most easily understood from examples. It would be difficult to construct a single example exhibiting all nine, but

(2) Margot opened the cupboard outside her sister's bedroom with her key at 9:15 p.m. on her way from the kitchen to the attic along the landing

crams in eight of them. Margot is the Agent, the cupboard is the Object, outside her sister's bedroom is the Location, her key is the Instrument,

[3] Due originally to Gruber (1965) but largely developed by Fillmore (1966, 1968a, 1968b, 1971, 1975a) and Anderson (1971, 1977).

[4] A more recent addition to this list is Benefactive, the rôle of something which benefits from an action. Indeed, some authors posit as many as twenty-five distinct rôles, whereas others restrict them to four.