2
Linguistic background

This chapter introduces some of the terminology, ideas and methods of those aspects of linguistics which are most relevant to MT. Readers who have a good background in linguistics might like to skip parts of this chapter, or read it quickly so as to see our own, perhaps idiosyncratic, approach to familiar material. The first part of the chapter (sections 2.1 to 2.7) introduces some basic terminology and concepts; the second describes briefly some of the formal apparatus of linguistic description (section 2.8), the linguistic theories which have influenced much MT research (section 2.9), and some recent developments which are beginning to influence MT research in various ways (section 2.10). Some knowledge of the content of sections 2.1 to 2.8 is essential for the understanding of all chapters. The information in the last two sections should be regarded as supplementary, for reference when the need arises.

2.1 The study of language

The study of language takes many forms. It can concentrate on historical aspects, language change and evolution, older forms of a language, the origin of language. It can concentrate on social aspects: how language is used to communicate, to influence opinions, to persuade others, how it differs between the sexes and classes, from region to region, from one social context to another. It can concentrate on psychological aspects: language acquisition by children, second language learning by adults, the language of the mentally handicapped or mentally ill, the creative use of language, etc. It can concentrate on political and geographical aspects:

problems of language spread and death, bilingualism, dialects and their status, national languages, standardization and normalization of technical terminology. Finally, and in the opinion of many the most central concern, it can concentrate on the 'nature' of language, on the underlying 'system' of language and languages.

There is a long and successful tradition in Western linguistics that language systems can be studied in isolation in a worthwhile way. It has always been routine for writers of grammars to abstract away from the heterogeneity of actual speech communities and to present some kind of community norm. Linguists do not attempt to account for the full range of observable phenomena but rather concentrate on aspects which they believe can be handled satisfactorily in a systematic fashion. The basic assumption is that language users do not constantly refer to a vast store or memory of previously observed utterances but have available a set of abstract formulae and rules enabling them to construct and understand expressions which have never occurred before.

This tacit and unconscious 'knowledge' of the system of rules and principles is said to represent the **competence** of native speakers. It is contrasted with their actual **performance**, the use and comprehension of language on particular occasions, the directly observed spoken utterances and written texts, including any hesitations, slips of the tongue and misspellings. Although performance is founded on, and ultimately explained by, the competence of speakers, it directly reflects competence only in ideal circumstances. Actual communication is determined as much by external factors as by the 'internalised' grammar, e.g. by memory limitations, lack of concentration, nervousness, inebriation, prejudice and social constraints.

Consequently, linguistics must go beyond observation and description: it must also explain the judgements, intuitions and introspections of speakers. For example, how can ambiguity and synonymy be 'observed'? How else other than by asking directly can we discover that (la) can sometimes be equivalent to (1b) and sometimes to (lc)?

(la) Visiting relatives can be boring.

(1b) It can be boring to visit relatives.

(1c) Relatives who visit can be boring.

And how can observation help to discover that the similar sentence (2a) is not ambiguous, why it can be equivalent only to (2b) and why (2c) is anomalous? (Following the well-established practice in linguistics, unacceptable or anomalous sentences are marked with an asterisk here and throughout this book.)

(2a) Visiting hospitals can be boring.

(2b) It can be boring to visit hospitals.

(2c) *Hospitals which visit can be boring.

2.2 Grammar

It is assumed that the basic task of linguistics is to study the competence of the 'ideal' speaker-hearer, which is represented by the **grammar** of rules and principles underlying the potential ability of any native speaker to utter and comprehend any legitimate expression in the language. Theoretical linguistics is therefore concerned primarily with the rigorous and precise (i.e. formal) description of grammars, which (as is now widely accepted by most linguists) should satisfy the following basic requirements. Grammars should be **observationally adequate,** by being capable of demonstrating whether a particular string of words is or is not well-formed. They should be **descriptively adequate,** by assigning structural descriptions to strings of well-formed sentences in order to explain native speakers' judgments about relationships between utterances. And they should be **explanatorily adequate,** by representing the best available descriptively adequate grammar in the light of a general theory of what kinds of grammars are possible for human languages. The latter requirement leads linguistics to consider the principles and constraints on language as such, and hence to investigations of **universal grammar,** i.e. a description of the resources available for constructing the grammars of individual languages. The more powerful the theory of universal grammar, the less comprehensive individual grammars need to be, because the specific rule-systems of particular languages will be consequences of the general properties of universal grammar.

There are arguments about the autonomy of formal linguistics. On the one hand, some linguists maintain that 'language systems' must be described independently of particular usages of languages, and universal grammar must be studied independently of particular grammars of individual languages, just as physics describes the principles of motion independently of the particular movements of objects in particular situations. On the other hand, some linguists believe that theoretical constructs must have some psychological 'reality', in order that theories can be 'tested' against actual linguistic performance. To some extent, these arguments can be disregarded in fields of 'applied' linguistics such as computational linguistics: what matters is whether the results have practical value in suggesting methods and approaches of greater power and generality in more rigorous, internally consistent, and perhaps universally valid frameworks.

2.3 Phonology and orthography

Although almost all MT systems at present deal with (written) text input and output, research in speech MT is taking place and will no doubt increase in the near future. In this respect, phonetics and phonology obviously play an important role. Phonetics is concerned with the description of the sound units of languages in general; **phonology** is concerned with the description of the sound units (**phonemes**) available in a particular language and the way they combine with each other. For speech MT, analysis of the input sound, and generation of synthetic speech in the target language are the two problems in this area. Of these, the former is significantly the more difficult. While our writing system tends to identify individual speech sounds that make up a word (roughly one letter per sound), in reality the exact acoustic properties of the individual phonemes vary greatly depending on the surrounding context (as well as varying from speaker to speaker, and even from one occasion to another with the same speaker). For example the [t] sounds in *top, stop, bottle, pot* and *scotch* are all acoustically quite dissimilar, although sharing certain acoustic features which however are also found in other phonemes (for example [d], [s], [th]). It is the combination of distinctive acoustic features which enables the brain — or a speech recognition device — to interpret them as representing the [t] sound. Analysis of speech input by computer is usually based on statistical probability: each element in a sequence of sounds is open to various interpretations, but usually only one combination of these interpretations is statistically likely. This is where phonology, describing what combinations of sounds are permitted, plays a role.

By contrast, synthesising speech is a problem that is well on its way to being solved: much synthetic speech is almost indistinguishable from the real thing (for example the standard messages sometimes heard on the telephone are often synthesised, and are not recordings of human speech). Nevertheless, problems with **suprasegmental** features such as pitch, stress and tone remain: robot speech in TV and cinema science fiction is often portrayed as lacking these features (an accurate reflection of the state of the art in the late 1980s!).

For most MT systems, such problems are irrelevant, since they deal with written input and output. As such, the **orthography** (or spelling) is usually given *a priori,* though a system which was designed to deal with **ill-formed input** would need to know about **orthotactics** — what sequences of letters are legal in a given language — to be able to handle and correct misspellings. For example, English allows words to begin with the sequences *spr-, spl-, str-, skr-, kl-, sl-,* but not **stl-, *skl-* or **sr-;* the initial sequence *zmrzl-* is possible in Czech *(zmrzlina* ‘ice cream’) but not in English. However, many orthotactically possible words do not actually occur in a language, e.g. in English we have *split, slit, lit, slid* and *lid,* but not **splid;* and we have *gull, dull* and *dud,* but not **gud.* The knowledge of possible combinations of letters can be used to suggest corrections of misspelled words.

Orthography also covers paralinguistic problems such as the use of different type styles and punctuation, a little-studied but nevertheless important issue in multilingual tasks like translation. For example, in English, an italic type-face (or underlining) is often used to indicate emphasis, while in German the convention is to space out the words l i k e t h i s. In Japanese, emphasis is more usually indicated by a change of word order, or by a special suffix. Similarly, punctuation conventions differ. In English, we distinguish restrictive and descriptive relative clauses with the use or not of a separating comma, as in (3), whereas in German the comma is required irrespective of the type of relative clause:

(3a) Books which are imported from Japan are expensive.

(3b) Futons, which are Japanese-style beds, are very comfortable.

In other cases, English uses commas liberally for emphasis (4a), where German uses none (4b).

(4a) This year, the man, however, and his wife, too, will go on holiday.

(4b) *Dieses Jahr werden jedoch der Mann und auch seine Frau*
Urlaub machen.

Other conventions differing from language to language are those regarding quote marks (e.g. the use in French of « and »), use of capital letters (e.g. German nouns) or not (e.g. nationality adjectives in French), question marks with indirect questions, and much more.

2.4 Morphology and the lexicon

Morphology is concerned with the ways in which words are formed from basic sequences of phonemes. Two types are distinguished: inflectional morphology and derivational morphology. Words in many languages differ in form according to different functions, e.g. nouns in singular and plural *(table* and *tables),* verbs in present and past tenses *(likes* and *liked).* **Inflectional morphology** is the system defining the possible variations on a root (or base) form, which in traditional grammars were given as 'paradigms' such as the Latin *dominus, dominum, domini, domino,* etc. Here the root *domin-* is combined with various endings *(-us, -um, -i, -o,* etc.), which may also occur with other forms: *equus, servus,* etc. English is relatively poor in inflectional variation: most verbs have only *-s, -ed* and *-ing* available; languages such as Russian are much richer, cf. the verb *delat'* with present tense forms *delayu, delaeš, delaet, delaem, delaete, delayut,* and past tense forms *delal, delala, delalo, delali.* Languages can be classified according to the extent to which they use inflectional morphology. At one end of the scale are so-called **isolating** languages, like Chinese, which have almost no inflectional morphology; at the other end are **polysynthetic** languages, of which Eskimo is said to be an example, where most of the grammatical meaning of a sentence is expressed by inflections on verbs and nouns. In between are **agglutinative** languages, of which Turkish is the standard example, where inflectional suffixes can be added one after the other to a root, and **inflecting** languages like Latin, where simple affixes convey complex meanings: for example, the *-o* ending in Latin *amo* ('I love') indicates person (1st), number (singular), tense (present), voice (active) and mood (indicative).

Derivational morphology is concerned with the formation of root (inflectable) forms from other roots, often of different grammatical categories (see below). Thus, from the English noun *nation* may be formed the adjective *national;* the further addition of *-ise* gives a verb *nationalise,* adding the suffix *-ism* instead gives the noun *nationalism,* or adding *-ist* gives an agentive noun *nationalist.* And by yet further suffixation we can have *nationalisation* and *nationalistic,* or by adding prefixes *renationalise, denationalisation,* etc.

Often included under the heading of morphology is compounding, where whole words are combined into new forms. The meanings of compounds are sometimes obvious from their components *(blackberry),* sometimes slightly different (a *blackboard* is a special type of board, typically but not necessarily black), and sometimes completely opaque (a *blackleg* is a traitor or strike breaker). What makes compounding a problem in morphology is that in some languages (though not usually in English), compounds are formed by simply writing the two words together, without a space or hyphen between them. This is a problem when new or novel compounds are formed, and so do not appear in the dictionary (e.g. in German *Lufthansafrachtflüge* 'Lufthansa cargo flights').

The **lexicon** of a language lists the **lexical items** occurring in that language. In a typical traditional dictionary, entries are identified by a base (or 'canonical') form of the word. This sometimes corresponds to the uninflected root (as in English), though not always. In French dictionaries for example, verbs are listed under one of their inflected forms (the infinitive): *manger.* In Latin dictionaries, nouns are given in the nominative singular *(equus),* and verbs in the 1st person singular present tense active voice *(habeo).* Traditional dictionary entries indicate pronunciations, give grammatical categories, provide definitions and (often) supply some etymological and stylistic information.

The lexicon in a MT system and generally in linguistics is slightly different. Some MT systems have only **full-form lexicons**, i.e. lists of the words as they actually occur, with their corresponding grammatical information. So for example the lexicon might list separately the words *walks, walking* and *walked.* This option is less attractive for highly inflecting languages, where each lexical item may have ten or twenty different forms, and so the lexicon will list a root form, and there will be an interaction with a morphology component to analyse or generate the appropriate forms (see further in sections 5.1 and 7.2.1).

The MT lexicon will give the information needed for syntactic and semantic processing (see below): grammatical category (noun, verb, etc.), **subcategorisation features** (i.e. what 'subcategory' the word belongs to, e.g. transitive or intransitive verb, masculine or feminine noun), and semantic information (animate noun, verb requiring human subject). Often these last two types of information are used in conjunction with each other, as when a subcategory is defined in terms of **selection restrictions** on the words it can occur with. So for example, the verb *laugh* has a selection restriction on its subject, namely that it be animate.

A MT system must obviously also include data on correspondences between lexical items in different languages. Because this is often quite complex, many systems separate the information required for analysing or producing texts in one particular language and the information about lexical correspondences in two languages. The former is contained in **monolingual** lexicons and the latter in **bilingual** (or transfer) lexicons. As well as word-pair equivalents, MT lexicons often indicate the conditions under which equivalences can be assumed: different grammatical categories *(feed:* verb or noun), semantic categories *(board:* flat surface or group of people), or syntactic environment *(know* a fact or how to do something) and so on.

2.5 Syntax

Syntax comprises the rules or principles by which words (lexical items) may combine to form sentences. Rules apply to the **grammatical categories**. It is common to distinguish between the grammatical categories of individual lexical items such as noun, determiner (article), adjective, verb, adverb, preposition, etc., and the **constituents** indicating groupings of items, e.g. noun phrase, subordinate clause, sentence. Syntactic description also recognises **subcategories** or grammatically significant subdivisions of the categories, as mentioned in the previous section.

Linguists find it useful to distinguish major and minor categories. The **major categories** are the ones with large membership: nouns, verbs, adjectives. These are sometimes called **open class categories**, because new members are continually being added to the set (by making new derivations or compounds). The **minor** or **closed class categories** are the grammatical or function words like prepositions, conjunctions and determiners: these form small and finite sets which are rarely or never added to. This distinction is significant for MT in three ways: first, users of MT systems are normally permitted to add new words to the lexicon in the major categories, but not in the minor categories; second, the syntactic behaviour of minor category words is often more idiosyncratic, whereas the behaviour of open class words is easier to generalise; and third, an 'unknown word' in a text being processed by an MT system can (and must) be assumed to belong to one of the major categories.

Following the usual practice, grammatical categories are abbreviated in this book as 'n' for noun, 'v' for verb, 'adj' for adjective, 'det' for determiner, 'adv' for adverb, 'prep' for preposition, 'NP' for noun phrase, 'PP' for prepositional phrase, 'VP' for verb phrase, 'S' for sentence.

Syntactic descriptions are concerned with three basic types of relationships in sentences: **sequence**, e.g. in English adjectives normally precede the nouns they modify, whereas in French they normally follow; **dependency**, i.e. relations between categories, e.g. prepositions may determine the morphological form (or case) of the nouns which depend on them in many languages, and verbs often determine the syntactic form of some of the other elements in a sentence —see below); and **constituency**, for example a noun phrase may consist of a determiner, an adjective and a noun.

2.5.1 Syntactic features and functions

Relations between constituent or dependent items are sometimes indicated by the sharing of **syntactic features**. The difference between (5a) and (5b) lies in the fact that in (5a) the noun and the verb are both 'singular' and in (5b) they are both 'plural'. The extent of such kinds of agreement can be greater in other languages, as in the French example (6), where each word is marked as 'plural' and in addition the noun *rues* and the adjectives *grandes* and *embouteillées* are all 'feminine'. In the Russian examples in (7), the verb in each case agrees with its subject in both number and gender.

(5a) The boy runs.

(5b) The boys run.

(6) *Les grandes rues sont embouteillées.*
'The main roads are jammed'

(7a) *Čelovek kuril.*
'The man was smoking'

(7b) *Ženščina kurila.*
'The woman was smoking'

(7c) *Lyudi kurili.*
'The people were smoking'

The sources of agreement relations lie in the governor in a dependency relation or in the head of a phrase structure. The governor or head is the element or item which is obligatory in a structure: the verb in a sentence or verb phrase, the noun in a noun phrase, the preposition in a prepositional phrase, etc. These heads or governors determine the forms of dependent (or governed) elements: adjectives agree with nouns, prepositions ‘govern’ particular noun endings, etc.

The **grammatical functions** of words in sentences are related to syntactic roles. In many European languages, the noun (or noun phrase) in the nominative (if applicable) which agrees in number with a following verb is referred to as the grammatical ‘subject’ of the sentences *(the boy* in (5a) for example). Other functions include those of ‘direct object’ and ‘indirect object’, for example *the book* and *(to) the girl* respectively in both (8a) and (8b).

(8a) The man gave the book to the girl.

(8b) The man gave the girl the book.

Other sentential functions include ‘prepositional object’ *(on his deputy* in (9)), ‘sentential complement’ *(that he would come* in (10)) and ‘adverbials’ which (in English for example) are typically adverbs or prepositional phrases *(soon* in (11a) and *in a few minutes* in (11b)).

(9) The President relies on his deputy.

(10) He promised that he would come.

(11a) He will arrive soon.

(11b) He will arrive in a few moments.

Within the noun phrase typical functions are ‘determiner’ *(the, a* and so on), ‘quantifier’ *(all, some,* numbers) and ‘modifier’ (adjectives, relative clauses).

2.5.2 Deep and surface structure

Syntactic functions such as ‘subject’ and ‘object’ refer to the functions of the elements in the sentence largely irrespective of meaning. So for example, although *the man* is subject of (8a) and (8b), in related sentences (12a) and (12b), the subject is *the book* and *the girl* respectively.

(12a) The book was given to the man by the girl.

(12b) The girl was a given the book by the man.

However, linguists commonly make a distinction between a **surface structure** and an underlying **deep structure**, and consider passive forms like (12a) and (12b), as well as nominalisations as in (13a) to have the same underlying forms as (8a) and (13b) respectively. The syntactic functions of the elements in these hypothetical underlying forms can be recognised as ‘deep subject’ and ‘deep object’. Therefore, in (12a) *the book* is simultaneously surface subject and deep object.

(13a) the destruction of the city by the enemy...

(13b) The enemy destroyed the city.

Linguists differ as to whether the underlying form actually is the same as the most neutral surface form (for example, the active declarative sentence), or is some abstract **canonical form** from which active, passive, nominalisation and so on are all derived in some way. We will return to this question below. However the distinction between deep and surface syntactic functions is one which is generally accepted and understood.

2.5.3 Predicate-argument structure

Syntactic relationships within sentences may also be described in terms of **predicate-argument structures**. This term refers to the traditional division of propositions in logic into **predicates** and **arguments**. A sentence such as (5a) above corresponds to the proposition *run (boy),* where the predicate *run* has a single argument *boy;* a sentence such as (8a) corresponds to *gave (man, book, girl),* a proposition with three arguments. The predicate usually corresponds to the main verb in the syntactic structure, and the arguments are its dependents. A representation which focuses on this aspect of sentence structure is called a 'dependency representation', see 2.8.1 below. The assignment of different functions or roles to the arguments is part of the grammatical theories of Valency and Case, both widely used in MT, and which are discussed in sections 2.9.5 and 2.9.6 below.

2.6 Semantics

Semantics is the study of the ways in which individual words (lexical items) have meaning, either in isolation or in the context of other words, and the ways in which phrases and sentences express meanings. A common assumption is that word meanings involve **semantic features**. Words such as *man, woman, boy, girl* share a common feature 'human' in contrast to animals, and share with animals a feature 'animate' which distinguishes them both from inanimate physical objects *(rock, table, chair)* or from abstract notions *(beauty, honesty).* Furthermore, a feature such as 'male' distinguishes *man* and *boy* from *woman* and *girl,* and a feature 'young' distinguishes *boy* and *girl* from *man* and *woman.* Such features indicate not only the potential range of extra-linguistic objects to which they may refer (i.e. assuming a matching of semantic features and real-world attributes), but also the appropriate conjunction of words in texts (sentences), e.g. *girl* and *dress, chair* and *sit.* Such features are often organised into a **semantic feature hierarchy**: for example, 'humans' along with animals, birds, reptiles and insects are all 'animate'; animate things along with plants are 'living'; living beings along with artefacts are all 'physical objects'. Generalisations can be made at various different points in such a hierarchy: for example, any animate thing can be the subject of the verb *walk,* but only humans (normally) can *talk.* Such generalisations can be captured by supposing the hierarchy to be an inheritance hierarchy, so that it is sufficient to say that the word *teacher* has the feature 'human' to know that it is also 'animate', 'living', and so on. Inasmuch as semantic features tend to reflect the realities of the world around us, semantic feature hierarchies are rarely as simple these examples suggest, and semantic features are more usually arranged in 'polyhierarchies' or **semantic networks** with lots of interwoven inheritance hierarchies.

It is common to study relationships between lexical items within a semantic field or 'semantic system'. The vocabulary of kinship is one example: *father, mother, son, daughter, uncle, brother, grandfather, cousin,* etc. Another could be the verbs of motion: *walk, ride, drive, swim, run,* etc. In many cases the analysis of a semantic field or system can be formulated in terms of semantic features.

Words in the same semantic field often have similar or comparable SYNTACTIC behaviour: for example, most verbs of motion are intransitive and take a locative prepositional phrase as a prepositional object.

Along with semantic features are semantic functions, also known as 'case roles': here, the semantic relationship between the predicate and its arguments is captured. This notion is at the centre of the theory of Case grammar and is discussed in more detail below (section 2.9.6).

Whether a particular word or expression is appropriate to refer to some entity or event is determined not only by the semantic features which constitute its denotation, but also by less easily formalised aspects of connotation. Among these may be included differences of register: a *friend* may be called *pal, mate* or *guy* in colloquial usages, the informal *lavatory or loo is* likely to be referred to as a *public convenience* in official statements, etc. Differences of subject domain affect semantic usage: for the physicist the term *field* means something quite different from the farmer's *field,* and terms such as *force* and *energy* are defined in ways which common usage is unaware of.

2.7 Text relations

Links between and within sentences are conveyed by the use of pronouns, definite articles, nominalisations, etc. Consider a sentence like (14)

(14) An old soldier bought a pair of trousers.

In a subsequent sentence the old soldier may be referred to by an appropriate pronoun (e.g. *he* for male persons). Or the same kind of reference may be conveyed by a definite expression *(the man),* or by a nominalisation of his action *(the purchaser).* The term **anaphoric** reference is widely used to cover these usages. Within sentences, such reference may be expressed by reflexive pronouns (15).

(15) The man bought himself a shirt.

However, it should be noted that pronouns and definite expressions may also refer to entities not previously mentioned in a text or discourse; they may refer to persons or objects already known or easily inferable from the situation —this is **deictic** (or 'pointing') reference. For example, in the brief text in (16a), the pronoun *it* refers not to the restaurant, but to the unmentioned but inferable meal eaten there; similarly, in (16b), the *it* refers to the sea implied by the mention of a ferry boat.

(16a) We went to a restaurant last night. It was delicious.

(16b) We took the ferry from Liverpool to Dublin. It was very rough.

Less frequently, anaphoric or deictic reference points forward rather than backwards. For example, (16b) can easily be rephrased as in (17a). Yet there are some restrictions on forward anaphora, or **cataphora** as it is sometimes called. Sentence (17b) seems acceptable, with the cataphoric reference of *it* to *car,* but in (17c) we cannot be sure if *he* refers to *the old man* or to some other person.

(17a) It was very rough when we took the ferry from Liverpool to Dublin.

(17b) You can put it in the garage if you come by car.

(17c) He looked relieved when the old man heard he didn't have to pay tax.

In general, the sequence of words and ideas in sentences is determined by text relationships. The start of a sentence typically makes a link to previous sentences; the initial words refer to elements already mentioned or presumed to be already known — either by anaphoric reference or by presupposition. The remainder of the sentence conveys something new about these ‘known’ or ‘given’ elements. The first part of the sentence is what it is ‘about’, and is called its **theme**. What follows, the new information, is the **rheme**. In an alternative terminology, these are **topic** and **comment** respectively. The processes involved in the selection of theme elements is frequently referred to as **thematisation**: in English and other European languages, passivisation is one common means of changing the theme of a sentence.

In English, the distinction between old and new information is expressed primarily by the use of definite or indefinite expressions, but in languages such as Russian where there are no articles, the distinction is conveyed by word order (18). Initial (thematic) elements are assumed to have been already known to the listener or reader, and later (rhematic) elements are perhaps being mentioned for the first time in the present context.

(18a) *Ženščina vyšla iz domu.*
WOMAN-nom CAME OUT HOUSE-gen
‘The woman came out of the house’

(18b) *Iz domu vyšla ženščina.*
‘A woman came out of the house’

In (18a) *ženščina* is an old (thematic) noun, where the definite article is appropriate in English; in (18b) it is a new (rhematic) noun, where English requires an indefinite article.

2.8 Representations

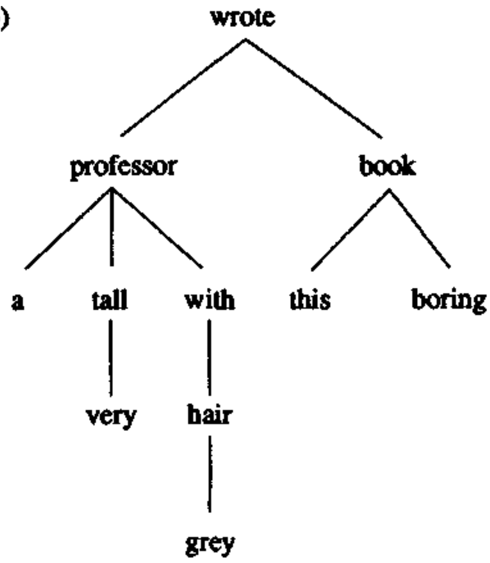
A major focus of formal linguistics is the definition of structural representations of natural language texts, and the definition of formal grammars which define the range of well-formed representations. Earlier, we mentioned three types of syntactic relationship found in linguistic descriptions: sequence, dependency and constituency. Two basic types of representation are in common use, emphasising one or other of dependency or constituency; sequence is optionally indicated on both types of representation.

2.8.1 Dependency

A traditional method of representing the dependency relations in a sentence is by a **dependency tree** where the dependents of each governor are portrayed as stemming from them on branches. In (19a) and (19b) we give a sentence and its corresponding dependency tree structure.

(19a) A very tall professor with grey hair wrote this boring book.

(19b)



The adjective *tall* is modified by the adverb *very,* and so governs it; determiners *(a, this)* and adjectives *(boring, tall, grey)* are governed by nouns *(professor, book, hair);* nouns are dependent on prepositions *(with)* or on the verb *(wrote).* The head or 'governor' of the whole sentence is the main verb.

We can indicate sequence in a dependency tree by convention: either by attaching significance to the ordering of branches or by labelling the branches with information from which the sequence can be derived. The first approach is illustrated in (19b) where the branches are ordered left-to-right in a sequence which corresponds to English word order. However, in this case we need also some general and specific rules to know where the governor fits in the sequence. In English for example, the governor generally goes in the middle: if the governor is a noun it comes after a determiner and any adjective(s) but before a prepositional modifier. Likewise the verb comes after a subject but before an object

Predicate-argument structure *(2.5.3* above) may thus be seen as an example of ordered dependency structure; in the proposition *gave (man, book, girl)* the governor is *gave* and its relationships to its dependents is given by their order, *man* before *book* before *girl.*

If we were now to ignore the left-to-right ordering of branches (and make the necessary lexical alterations), the dependency tree in (19b) could also serve equally well for an equivalent Welsh sentence (20a), where the governor typically precedes (nearly) all its dependents, or for an equivalent Japanese sentence (20b), where the governor always comes last.

(20a) *Ysgrifennodd athro tal iawn a gwallt llwyd ganddo y llyfr undonnog hwm.*
WROTE PROFESSOR TALL VERY & HAIR GREY TO-HIM THE BOOK BORING THIS

(20b) *Ichi-ban takai shiraga-de-no sensei-wa kono omoshirokunai hon-wo kaita.*
VERY TALL GREYHAIRED PROFESSOR THIS BORING BOOK WROTE.

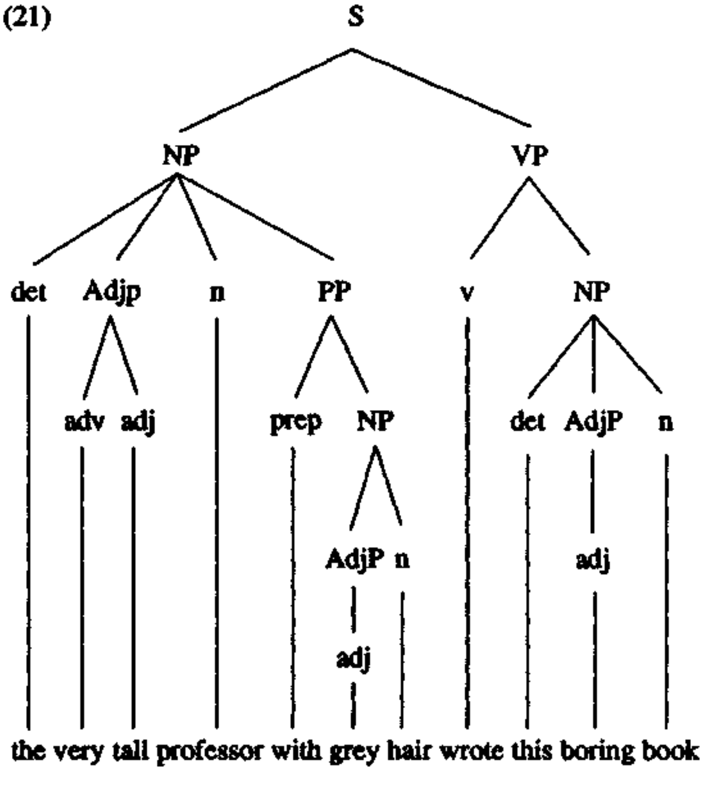
If in a dependency tree no significance is to be attached to the order of branches, then branches have to be labelled in some way if the sequence of elements is to be recorded. We could, for example, label one branch as subject and another as object while specifying no ordering of branches. In English the general rule would be for subjects to precede verbs (their governors) and for objects to follow; in Welsh the governor would precede subjects and objects, and in Japanese the governor would come last

This kind of dependency representation is often used in connection with Valency or Case grammar, which we will discuss below (sections 2.9.5 and 2.9.6).

2.8.2 Phrase structure

The traditional method of representing the structural constituency of a sentence is the **phrase structure tree,** e.g. for the same sentence (19a) the tree in (21).

(21)


S
NP
VP
det
Adjp
n
PP
v
NP
adv
adj
prep
NP
det
AdjP
n
AdjP
n
adj
adj
the very tall professor with grey hair wrote this boring book

This represents a typical analysis (though not necessarily one which would be adopted by all linguists). It shows that *this boring book* is a noun phrase (NP) containing a determiner *(this),* an adjectival phrase (AdjP) consisting of the adjective *(boring)* and a noun *(book);* that the prepositional phrase (PP) *with grey hair* is a constituent of the larger noun phrase *the very tall professor with grey hair,* that the verb *wrote* and the NP *this boring book* constitute a verb phrase (VP); and that the sentence (S) as a whole is basically in two parts: a noun phrase *(the very tall professor with grey hair)* and a verb phrase *(wrote this boring book).*

An alternative and equivalent representation of a phrase structure is a bracketed string of categories and elements (22):

```
(22) S(NP(det(the),
          AdjP(adv(very), adj(tall)),
          n(professor),
       PP(prep(with),NP(AdjP(adj(grey)),n(hair)))),
       VP(v(wrote),
       NP(det(this),
             AdjP(adj(boring)),
             n(book))))
```

To derive the tree (21) or its equivalent bracketed string (22), we have the rules in (23) and (24).

(23a) S → NP VP
(23b) NP → (det) (AdjP) n (PP)
(23c) AdjP → (adv) adj
(23d) VP → v NP
(23e) PP → prep NP
(24a) det → {*the, this*}
(24b) n → *{professor, hair, book}*
(24c) adv → *very*
(24d) adj → {*tall, grey, boring}*
(24e) prep → *with*
(24f) v → *wrote*

In each rule in (23) the category on the left can be replaced by the sequence of categories on the right, where those categories enclosed in brackets are optional. These rules are known as **phrase structure** rules. Following convention, we indicate terminal symbols in lower case. Terminal symbols are categories which do not appear on the left-hand-side of phrase structure rules, but appear in rules such as those in (24), where the surface strings are shown: curly brackets indicate alternatives. Clearly, in reality such rules would involve long lists of alternatives, so the formalism shown in (24) is rarely used: rather, terminal symbols are matched to categories listed in the lexicon.

A phrase structure grammar is said to **generate** a tree such as (21). The reader should be careful to distinguish two uses of the term 'generate': the use here — which comes from mathematics — defines a static relationship between a grammar and a representation or tree structure; there is another use of the term, which we shall use elsewhere in this book, referring to one of the procedures involved in producing a translation of a text. We shall return to this in the section 2.9.1.

Phrase structure rules capture various other grammatical relationships: **dominance** (category S dominates NP and VP in rule (23a), category VP dominates v and NP in rule (23c)) and **precedence** (NP precedes VP in rule (23a), and AdjP precedes n in rule (23b)). Linguists often use the terminology of genealogical trees to describe relationships between nodes: a **mother** node is a node which dominates another (so S is the mother of NP and VP in our example); conversely, NP and VP are the **daughters** of S; nodes which share a common mother are, of course, **sisters**. Conventionally, only female kinship terms are used.

Dominance, or motherhood, should not be confused with governorship in a dependency grammar, since the dominating category is made up of the elements it dominates, whereas in dependency, the governor is just one of the elements that make up the sentence. In fact, the dependency notion of governorship can be incorporated into a phrase structure grammar by singling out in each rule one of the daughters as being the governor. (However, since it must be a terminal symbol, the grammar in (23) would have to be adjusted if we wanted to do this.) The precedence relationship obviously corresponds to the idea of sequence mentioned earlier.

In a phrase structure grammar, syntactic functions can be defined in terms of dominance and precedence. For example, 'subject' might be defined as being the NP dominated by S and preceding a VP, while 'object' is a NP dominated by VP and preceded by a verb.

2.8.3 Feature-based representations

In several sections above we have mentioned additional features — syntactic and semantic — which play a part in linguistic representation. In general, features can be represented as **attributes** with corresponding values, and as such are often called 'attribute-value pairs'. Features can be used to represent almost everything we have discussed so far: 'category' can be an attribute with values such as 'noun', 'verb', 'noun phrase', etc; grammatical features such as 'gender', 'number', 'tense' are all possible attributes, with the expected range of values; 'surface function' might have values such as 'subject', 'modifier' and so on, while there might be a parallel attribute 'deep function', with the same range of values. Thus, the nodes of our tree representations can be labelled not with single values, but with features in the form of attribute-value pairs. Sets of such features are usually called **feature bundles**.

Linguistic theories which use feature-based representations often include a **feature theory** which, for example, defines the lists of attributes and their possible values, but also might include rules about which features can or must be combined. For example, the feature theory might say that if the value of the 'category' attribute is 'noun', then the 'gender' feature is relevant, while if it is 'verb', then 'tense' must have a value. There might also be a system of **default values**, for example we could assume that the value of 'number' for a noun is 'singular' if it is not otherwise specified.

The values of attributes need not be generally restricted to single atomic units. In the case of ‘semantic feature’ for example, we might want to stipulate a whole set of values, e.g. ‘{human, male, young}’ for *boy.* In some theories, the value of an attribute can itself be a feature bundle. In this way, an entire tree structure can be replaced by a complex **feature structure.** Traditionally, feature structures are enclosed in square brackets. Sets of values (which might also be feature bundles) are shown in curly brackets. For example, we could represent (21) above as the feature structure (25), where the constituents or daughters (‘dtr’) at each level are listed as an ordered set.

```
(25) [cat:sentence
        dtr:{[cat:np,function:subj,num:sing,
               dtr:{[cat:det,function:det,num:sing,lex:a],
                     [cat:adjp,function:mod,
                     dtr:{[cat:adv,function:mod,lex:very],
                     [cat:adj,function:head,lex:tall]}],
                     [cat:n,function:head,num:sing,lex:professor,
                     sem:{human}],
                     [cat:pp,function:mod,
                     dtr:{[cat:prep,function:head,lex:with],
                     [cat:np,function:obj,
                     dtr:{[cat:adjp,function:mod,
                           dtr:{[cat:adj,function:head,lex:grey]},
                           [cat:n,function:head,num:sing,lex:hair,
                           sem:{bodypart}]}}]},
               [cat:vp,function:pred,
               dtr:{[cat:v,function:head,tense:past,lex:write,string:wrote],
                     [cat:np,function:obj,num:sing,
                     dtr:{[cat:det,function:det,num:sing,lex:this],
                           [cat:adjp,function:mod,
                           dtr:{[cat:adj,function:head,lex:boring]}],
                           [cat:n,function:head,num:sing,
                           lex:book,sem:{ ETC}]}])]}]
```

It is also possible on the other hand to use feature structures to represent dependency trees, such as (19b), which is shown as (26). In this example, we treat syntactic function as the main feature on which to base the structure, rather than constituency (‘subj’, ‘obj’, ‘mod’ etc.) as in (25).

```
(26) [cat:v,tense:past,lex:write,string:wrote,
         subj:[cat:n,num:sing,lex:professor,sem:{human},
               det:[cat:det,num:sing,lex:a],
               mod:{[cat:adj,lex: tall,
                       mod:[cat:adv,lex:very],
                       [cat:prep,lex:with,
                       obj:[cat:n,num:sing,lex:hair,sem:{bodypart},
                       mod:[cat:adj,lex:grey]]]}
         obj:[cat:n,num:sing,lex:book,sem:{ ETC),
         det:[cat:det,num:sing,lex:this],
         mod:[cat:adj,lex:boring]]]
```

2.8.4 Canonical and logical form

Linguists often talk about **canonical form,** meaning a form of neutral representation. We have already seen this idea in section 2.4, where we talked about canonical form for lexical items, which can be the uninflected root, a conventional inflected form (e.g. infinitive of the verb in French and Spanish) or even some hypothetical or hybrid form, which might be a stem, even though this never normally occurs alone (e.g. Latin *equ-* 'horse'), or else some other made-up identifier. Although this last type of canonical representation is not usually found in dictionaries, it is quite common in other sorts of linguistic representation, where we want to talk about a word without drawing attention to its specific surface form. For example we might want a convenient way to refer to the verb *be* in all its forms *(being, been, am, are, was, were).* Lexical canonical forms may be given also for multi-word or discontinuous lexical items (e.g. *aircraft carrier, give ... a hand):* a device often used is to replace the space with an underline character '_', e.g. *aircraft_carrier.*

Canonical form is also — perhaps more often — used to describe in a neutral way a construct rather than a single lexical item. For example, just as we wish to relate the different inflected forms of a word to a single canonical lexical item, so too we might wish to have a way of referring to various closely related sentences or sentence parts (e.g. irrespective of tense, passivization, etc.). This notion of 'canonical form' is very close to the idea of deep structure, seen in 2.5.2, while a predicate-argument representation as in 2.5.3 is also sometimes used for this purpose. Often, canonical syntactic form is taken to correspond to the surface form of the least marked structure, e.g. an active declarative sentence with no special thematization effects. If this seems counter-intuitive in the light of the discussion of deep structure, it should be born in mind that in his earliest formulation of the concept, Chomsky believed that surface structures were derived from deep structures by transformations (see 2.9.2 below). It is not impossible to imagine that one possible surface structure is the one derived from deep structure without any transformations, and is thus identical to it. Since the transformations embody deviations from the most neutral form of a proposition, e.g. by introducing passive voice, then this equation of deep structure, canonical form, and unmarked surface structure is quite plausible.

A further candidate for canonical form might be **logical form,** which is a representation used to make explicit the meaning of an utterance in terms of the truth conditions it implies. This is a representation found in formal logic, in which notably the introduction and distinction of participants in the event being described is made explicit. Of particular interest to logicians is the issue of **quantifier scope,** which logical form representations make explicit. For example, a sentence such as (27a) would be represented as (27b), with the interpretation (27c).

(27a) Cats like fish.

(27b) $\forall x$ cat (x) $\forall y$ fish (y) like (x,y)

(27c) Whenever something *(x)* is a cat, then for whatever things which are fish (y), *x* likes *y.*

Logical form representations make explicit **quantifier scope ambiguities** such as the classical (28a) which might be represented as (28b) and which is to be read as (28c), or which might be represented as (28d), with the interpretation (28e).

(28a) Every man loves a woman.

(28b) $\exists y$ woman (y) ($\forall x$ man(x) love(x,y))

(28c) There is a woman whom all men love.

(28d) $\forall x$ man(x) ($\exists y$ woman(y) love(x,y))

(28e) For every man there is a woman whom he loves.

Although the second reading is pragmatically more plausible in this case, it is not always so obvious. Determining and indicating the correct quantifier scope is important because, as the following examples show, the two readings are not always equivalent, and important consequences might follow, e.g. if in some language the two readings receive different translations, or if, for some reason, we want to passivize the sentence, as in (29).

(29a) All the teachers in this room speak two languages.

(29b) Two languages are spoken by all the teachers in this room.

Only in (29b) can we be sure that there are two specific languages in question, whereas (29a) is simply a statement about multilingualism in general. Clearly, quantifier scope is closely related to differences of theme and rheme (section 2.7 above): (29a) is a statement about teachers and (29b) is about two (particular) languages.

2.9 Formal grammar and linguistic theory

2.9.1 Context free grammars and rewrite rules

The grammar in (23) is an example of a **context free grammar** (CFG). A CFG consists of a set of **rewrite rules** of the form A → α, where A belongs to a set of **non-terminal symbols** and α is a sequence of non-terminal and/or terminal symbols. The application of a sequence of rewrite rules is said to generate a representation of a sentence, and a grammar consisting of such rewrite rules is called a **generative grammar**. As we said above, we must be aware of the special meaning of the term ‘generate’ in this context. What a CFG grammar does is to define a formal relationship between a set of possible texts and their representations. This is actually a static definition, and the grammar is quite independent of the use to which it is put. The starting point can be either a text or the grammar. In the first case, the grammar is used to find out what the representation of the text should be; in the second case, the grammar is used to produce acceptable texts. It is reversible in this respect The static ‘declarative’ nature of CFGs is somewhat obscured by the perhaps misleading — but now firmly established — use of the arrow symbol ‘→’ and the term ‘generate’. CFG rules are not instructions (or ‘procedures’) to be followed, but descriptions or definitions; the rules are in a certain sense quite reversible. (For more on the distinction between ‘declarative’ and ‘procedural’ see section 3.2.)

CFGs have some very interesting properties when it comes to their implementation as computational devices (see Chapter 3). However, it has been recognised that the simple formalism seen in (23) is inadequate in several respects for describing and explaining certain natural language structures. For example, in order to account for a passive verb form, as in sentence (30) it would be necessary to add further phrase structure rules (31).

(30) This book was written by the old professor.

(31a) VP → Vpass (PP)

(31b) Vpass → aux pastpart

where Vpass represents a passive verb form, consisting typically of an auxiliary *(was)* and a past participle *(written).* But by having apparently independent sets of rules, it is difficult to make explicit the special relationship between active sentences and their passive counterparts.

In another respect, the rules in (23) are too tolerant: there is nothing to prevent a transitive verb appearing in an intransitive structure (32a) or an intransitive one in a transitive structure (32b).

(32a) *The man put.

(32b) *The man went the book.

One solution is to invoke **subcategorization features** (as mentioned above, section 2.4), which express the appropriate restrictions on the context in which a certain word may occur. So for example, we can stipulate that the v in rule (23c) must have a lexical entry specified as transitive, i.e. that it can occur followed by an NP: the lexical entry might include a subcategorization feature something like (33); the notation ‘[transitive]’ is sometimes written as ‘[— NP]’, where the bar indicates the position of the element in question.

(33) v[transitive] → *wrote*

Rule (23c) above must now be rewritten to indicate that the subcategory ‘v[transitive]’ is needed, as in (34).

(34) VP → v[transitive] NP

The distinction between categories like v and subcategories like ‘v[transitive]’ means that generalizations which apply to all v’s (irrespective of subcategorization feature) can easily be stated. Furthermore, categories are often subdivided according to various different criteria, for different purposes: these subcategorizations appear as features in the lexicon (cf. section 2.8.3), which can be evoked as necessary by different rules. For example, the pastpart in (31) can be replaced by a v with the appropriate subcategorization feature 'past-participle'.

One problem not readily addressed by simple phrase structure grammars concerns relationships between sentences such as the following, which differ according to whether the direct object is a full noun, as in (35a), or it is a '*wh*-interrogative' *(who, what,* etc.), as in (35b).

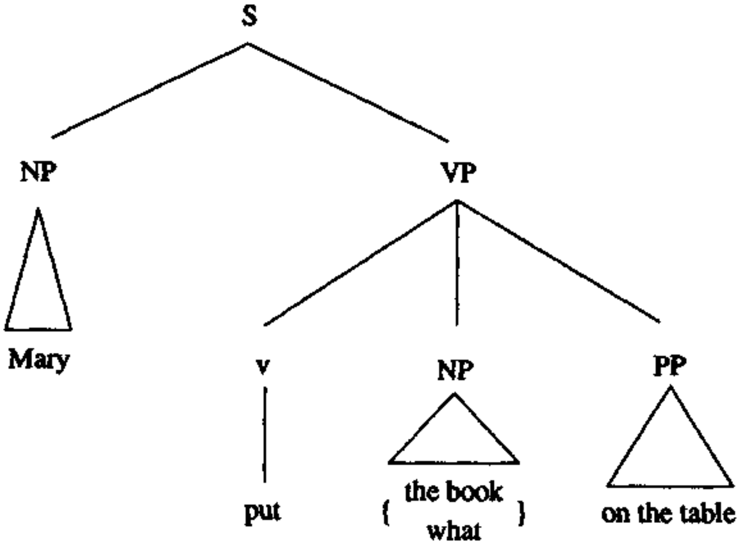
(35a) Mary put the book on the table.

(35b) What did Mary put on the table?

2.9.2 Transformational rules

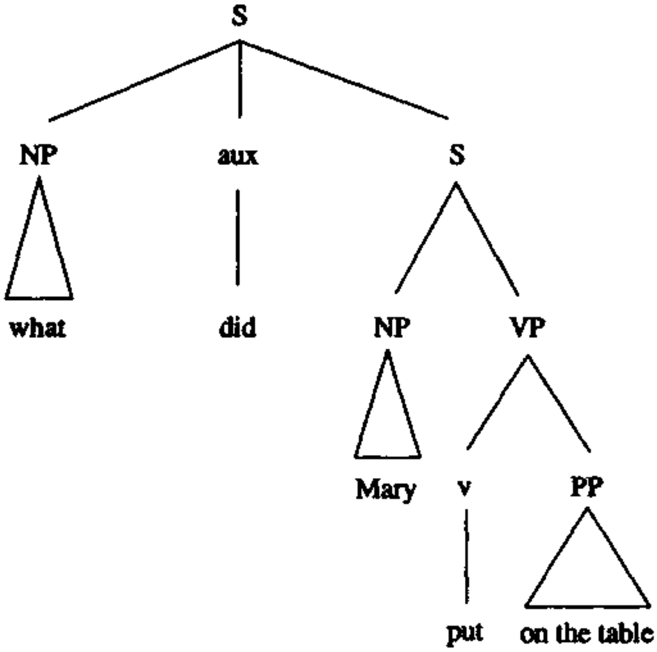
It was in an attempt to overcome such difficulties that the formal distinction was introduced between representations of surface structure and deep structure (cf. section 2.5.2 above). Both (35a) and the related (35b) are said to have similar deep structures (36) (the triangle notation in the tree structure indicates by convention that the details of this part of the tree are not of interest at the moment).

(36)



The surface form of (35a) is directly related to this deep form, but the surface form of (35b) is derived by **transformational rules** which move the NP *what* to the beginning and insert the auxiliary *did* between it and the subject NP *Mary* to give the surface structure tree (37).

(37)



Transformational rules are formulated in terms of pairs of structural descriptions. The change in structure between (36) and (37) can be captured by the rule (38).

(38) X NP v wh Y => X wh *did* NP v Y

where X and Y indicate optional unspecified structures that are unaffected by the rule. (Readers should note that this example is for illustrative purposes only; in fact this formulation would now be rejected by most linguists.)

Grammars including rules such as (38) have different and more complex properties (especially from the point of view of computational implementation) than CFGs. In particular, it is more difficult to be certain of the reversibility of such rules, since they are able to add and delete elements freely, as well as to move them around. The example (38) showed a fairly simple rule where the elements being manipulated are all single constituents, but the formalism of transformational rules can equally be used with more complex trees. In later chapters we shall see many examples of similar rules for manipulating trees in preparation for translating between languages with different sentence structures, e.g. between English and Welsh (examples (19a) and (20a) above). In this context, the process is generally referred to as **tree transduction** (see Chapter 6 in particular.)

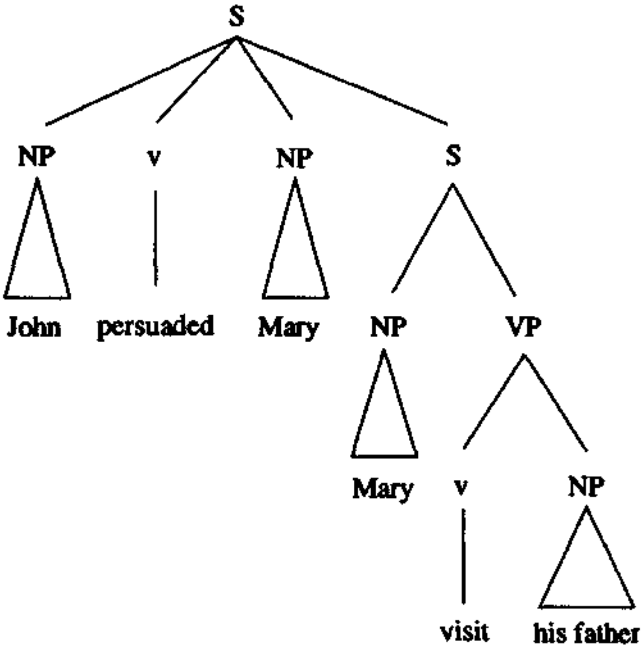
One major function of deep structure analysis is to illuminate relationships which are implicit in surface forms. Compare, for example the two sentences in (39).

(39a) John persuaded Mary to visit his father.

(39b) John promised Mary to visit his father.

We know that in (39a) the person who is (or was) to visit John's father is not John but Mary, whereas in (39b) it is not Mary but John. The deep structure for (39a) would be (40).

(40)



The derivation of the surface form (39a) from the deep structure in (40) involves the deletion of the NP *Mary* in the complement S, and the transformation of the finite verb *(visit)* into an infinitive. A similar deep structure for (39b) — with *promised* instead of *persuaded* and *John* instead of the second *Mary* — would involve the same rules to generate a surface form. Running the procedure in the opposite direction gives us the appropriate interpretation of the embedded sentence (which might be needed for translation into a language which makes the subject of the embedded sentence explicit or where knowing what the subject would be is needed for gender or number agreement): the lexical entries for *promise* and *persuade* tell us whether to copy the surface subject or to copy the surface object into the subject position in the embedded sentence.

As another example, consider the sentence (41 a). It is clear that the deep subject of *like* must be *John* and that *seems* expresses the speaker’s attitude to the proposition (41b). Consequently, a deep structure is proposed which corresponds roughly to the ‘equivalent’ expression (41c), where the sentence (41b) is the complement of *seems* in the whole structure. (Differences in meaning between (41a) and (41c) are attributable to differences of theme and rheme: cf. section 2.7 above.)

(41a) John seems to like easy solutions.

(41b) John likes easy solutions.

(41c) It seems that John likes easy solutions.

The generation of (41a) from such a deep structure involves the **raising** of the NP *John* from the embedded sentence to be the grammatical subject of *seem* in the sentence as a whole. Similar transformational rules involving the raising of elements occur in sentences such as (42)

(42) Mary is believed to be a good cook.

where *Mary* is the subject of *be a good cook* and the entire phrase *Mary is a good cook* is the complement of *believe.* Verbs that have this syntactic behaviour are often called 'raising verbs'.

The addition of transformational rules to the phrase structure rules gave the **transformational-generative** (TG) grammars associated with Chomsky, the first substantial attempt to overcome the limitations of simple phrase structure rules such as those in (23) and to develop a grammar with some descriptive adequacy. In this model there is a 'base component' containing a lexicon and phrase structure rules, and generating deep structures, a 'semantic component' which interprets deep structures, a 'transformational component' consisting of transformational rules which convert deep structures into surface structures, and a 'phonological component' which produces phonetic representations of sentences from surface structures. The core notion is that of abstract 'deep structure' representations, in which surface ambiguities are eliminated, implicit relationships are made explicit, and synonymous sentences have the same representation. From a multilingual point of view, the idea that translationally equivalent sentences share a common deep representation, but have different surface structures due to different transformational and phonological components, had some appeal.

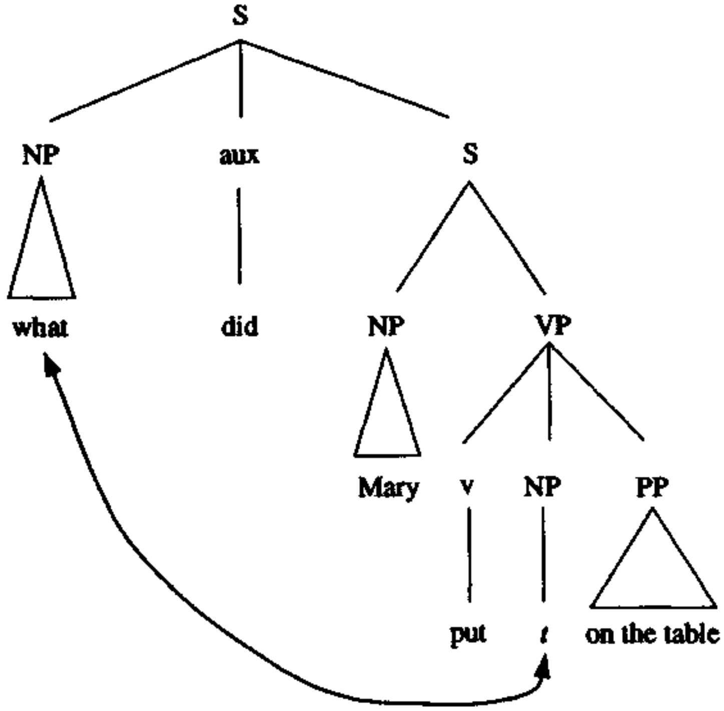
However, in the course of time the deficiencies of the TG model emerged. Briefly, two such deficiencies are, first, the excessive 'power' of transformational rules: in theory it is possible to devise a rule to do almost anything with a phrase structure which makes it virtually impossible for a parser (section 3.8) to recognise which particular transformational rule has applied; and, second, the recognition that semantic interpretation could not be restricted to deep structures and has to include information from surface structures, such as the scope of quantifiers (cf. 2.8.4 above). In the last twenty years various models of generative grammar have been developed in response to such difficulties, mainly by exploiting the full power of phrase structure rules and/or by restricting or eliminating transformational rules (sections 2.10.1 to 2.10.4 below). Nevertheless, although the theory itself is now superseded, the concepts of 'deep structure' and of 'transformation' are frequently referred to in many discussions of linguistic representations, and have influenced the design of many MT systems.

Certain concepts and formalizations are common to most of the models of generative grammar that have been developed since Chomsky's original formulation. Among them are the use of subcategorisation features (already discussed above), the treatment of traces and gaps, the $\bar{X}$ notation and the notion of valency and case (or thematic) roles. These concepts are discussed in the next three sections.

2.9.3 Traces and gaps

The notion of a trace may be illustrated by (43), a modification of (37) above.

(43)



Whereas in (37) the movement of *what* from a direct object NP position to an initial position left no object for *put* in the surface form, we now have a trace *(t)* of the NP. By this means, the subcategorisation requirements for *put* are intact and the semantic relationship between the verb and *what* can be read directly from the surface structure (43), thus eliminating the need to refer to deep structure in this respect. However, there are various limitations on the types of structural configurations in which a lexical element can be legitimately related (**bound**) to a trace element, and these have been discussed at length in the linguistics literature.

In contrast to a trace, a **gap** in a phrase structure represents lexical items or strings which have been completely deleted (and not just moved elsewhere). For example, in (44a) a second occurrence of *likes* has been omitted, and in (44b) the fragment *gone swimming* has been elided. The 'reconstruction' of such ellipses presents considerable problems both for theoretical linguistics and for computational analysis.

(44a) John likes fish, and Bill meat.

(44b) John has gone swimming and Bill has too.

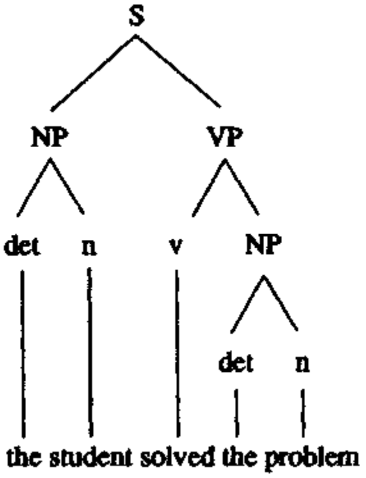
2.9.4 $\bar{X}$ theory

The formalism known as $\bar{X}$ theory ('$\bar{X}$' is pronounced — and sometimes written — as 'X-bar') developed from the recognition of similarities in the phrase structures of nominalisations and of sentences. Compare the structures in (46) for (45a-b).

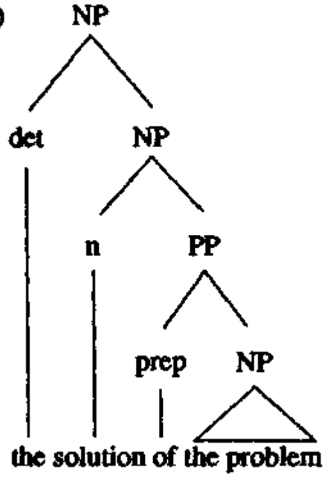
(45a) The student solved the problem.

(45b) the solution of the *problem*

(46a)

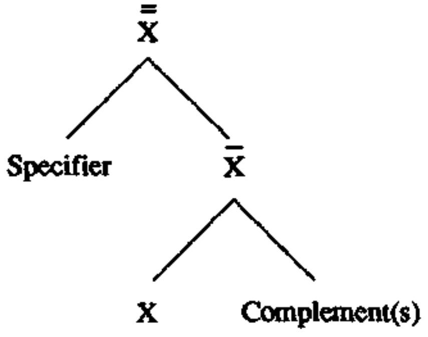


(46b)



In particular, (46b) contains a hierarchy of head categories: n *(solution)* is embedded as head item in an NP *(solution of the problem),* and this NP is in turn the head of the whole NP *(the solution of the problem).* To express such hierarchies of categories the bar notation was developed: a NP is thus a higher order n: $\bar{N}$, and a higher order NP is $\bar{\bar{N}}$ (pronounced 'N double bar'). The further generalisation of the notation and its extention to other configurations resulted in a uniform $\bar{X}$ structure:

(47)



The X can represent any category (noun, verb, adjective); the specifier can be filled by any appropriate preceding modifier (determiner, adverb); and the complement(s) by any succeeding modifier (noun phrase, verb phrase, prepositional phrase). The following are examples of structures dominated by $\overline{\overline{\text{N}}}$, $\overline{\overline{\text{Adj}}}$, and $\overline{\overline{\text{Prep}}}$, respectively:

(48a) this objection to the theory

(48b) too hot to handle

(48c) away from her problems

The specifiers are *this* (48a), *too* (48b) and *away* (48c). The $\overline{\text{X}}$ categories correspond to the dominating categories: in (48a) it is the NP (i.e. $\overline{\text{N}}$) *objection to the theory*, in (48b) it is an AdjP (i.e. $\overline{\text{Adj}}$) *hot to handle*, and in (48c) it is a PP (i.e. $\overline{\text{Prep}}$) *from her problems*. The complements are thus a PP (*to the theory*) in (48a), a VP (*to handle*) in (48b), and an NP (*her problems*) in (48c).

The basic idea of $\overline{\text{X}}$ theory is found in a number of current linguistic theories and appears also in computational linguistics, although direct applications of the notation in MT are rare and relatively recent (for one example see the representation in example (9) in Chapter 14).

2.9.5 Valency grammar

In section 2.8.1 we introduced the notion of a dependency representation, which differs from a phrase-structure representation in focusing on governor-dependent relations, corresponding to the predicate-argument structure mentioned in section 2.5.3.

In such a structure, a distinction is often made between complements or arguments which are closely associated with the verb, inasmuch as their syntactic form and function is predictable from the verb, and adjuncts which act as sentential modifiers. The number and nature of the complements which a given verb takes are given by its **valency**. Typically, verbs have a valency value between 1 and 3 (though verbs like *rain,* with a dummy subject, might be said to have a valency of 0). Monovalent verbs are intransitives like *fall* taking a single argument, the subject; divalent verbs take two arguments and may be transitive, taking a subject and object, or a subject and a prepositional phrase (e.g. *look* which has a prepositional complement with *at);* trivalent verbs include verbs taking a subject, direct and indirect object (e.g. *give),* or subject, object and prepositional complement (e.g. *supply someone with something),* or even two prepositional complements (e.g. *rely on someone for something).* Complements may also take the form of clauses introduced by *that* (e.g. *say that...),* infinitival clauses introduced by *to* (e.g. *expect to...)* or participial clauses (e.g. *like going).*

While the complements can be predicted by (or are controlled by) the verb, adjuncts are less predictable and can occur more freely with almost any verbs. The predicate and its complements express the central core of a sentence, while the adjuncts express the circumstances of time, location, reason, purpose, and so on, which accompany the central proposition.

Besides describing the syntactic form of its arguments, a valency description might also express semantic feature restrictions on the arguments. For example, the entry for *supply* might stipulate that the subject and object should be human or have human connections (e.g. an institution), while the with-complement must be a commodity.

Valency grammar is useful in translation in two ways: first, although corresponding verbs in different language often have different syntactic structures, their numerical valency is usually the same. By juxtaposing valency information, the correspondences can be identified. For example, while *look* takes a prepositional complement with *at,* the corresponding French verb *regarder* takes a direct object. An extreme example of this is *like* and *plaire:* compare (46).

(46a) The audience liked the film.

(46b) *Le film a plu aux spectateurs.*

The subject of *like* corresponds to the indirect object of *plaire,* while the direct object becomes the subject in the French.

The second benefit of Valency grammar is in the translation of prepositions: those marking complements are rarely translated directly, unless by coincidence the valency patterns coincide (e.g. *count on* and *compter sur):* more commonly there is an arbitrary correspondence of prepositions, if indeed both verbs take prepositional complements (cf. *look at / regarder).* Table 2.1 shows a range of German and English verb pairs and the corresponding prepositional complements, and indicates the wide range of possibilities: for example, although *mit* usually translates as *with, rechnen mit* is *count* ON, *prahlen mit* is *boast* OF and so on.

	at	in	of	on	to
an	*arbeiten* work	*glauben* believe	*denken* think	*stecken* stick	*gewöhnen* accustom
auf	*blicken* look	*trauen* trust	*zutreffen* be true	*rechnen* count	*weisen* point
für		*s.interessieren* be interested	*zutreffen* be true		*sorgen* see
in	*ankommen* arrive	*kleiden* dress		*einsteigen* get	*einwilligen* agree
mit		*handeln* trade	*prahlen* boast	*rechnen* count	*geschehen* happen
zu			*gehören* be part	*gratulieren* congrtulate	*führen* lead

Table 2.1 Prepositional complements in English and German

If a prepositional phrase is a complement, we must look to the verb to see how to translate it. On the other hand, adjunct prepositional phrases can usually be translated independently of the verb.

Valency does not only apply to verbs and their arguments. Adjectives too have valency patterns, especially when they are used predicatively (e.g. *proud* OF, *interested* IN, *concerned* WITH). Within noun phrases too, valency patterns can be predicted: where the noun is derived from a verb, or is semantically cognate, the complement structure is often derived in parallel: cf. (13) above and (47)-(50).

(47) The people are fighting for freedom.
the people's fight for freedom...

(48) The boy gave his brother a book.
the boy's gift of a book to his brother...

(49) The government raised £10m by indirect taxation.
the raising by the government of £10m by indirect taxation...

(50) Coventry dramatically beat Spurs by 3-2 after extra-time in the Final
at Wembley in 1987 for the first time ever.
Coventry's dramatic first-ever extra-time 3-2 victory against Spurs
in the Final at Wembley in 1987...

For other types of nouns the type of modification can also be predicted, and hence they can be said to have a valency structure. For example a *book* is often *of* some type of material, *by* someone, *about* something, and so on.

2.9.6 Case grammar

An elaboration of dependency, valency, and predicate-argument structure is found in Case grammar, in which the semantic functions of the complements (and in some versions, the adjuncts) are given. The nature of the relationship of a dependent noun or noun phrase (argument) to its governing verb (predicate) in a sentence may be expressed in terms of roles such as 'agent' (instigator of an action), 'patient' (entity affected), 'instrument' (means of achieving action), etc. Across languages, these semantic functions remain constant while the syntactic functions (subject, object and so on) differ. The semantic function label is thus a convenient point of contact for the syntactic representations for the corresponding sentences. The name 'Case grammar' derives from the term 'deep case', an unfortunate term intended to underline the distinction between surface syntactic functions (or 'surface cases' like nominative, accusative, dative) and deeper semantic functions. Other terms indicating roughly the same concept include 'semantic role', 'thematic role' and so on.

It is postulated that the range of possible semantic functions is relatively small: typically Case grammarians have identified a set of about ten. As well as those mentioned above, they include 'experiencer' (entity undergoing a non-physical action), 'recipient', 'source' (initial state or cause of action), 'goal' (aim or result of action), 'location' and (a few) others. So for example in (51) the case roles might be agent *(the teacher),* patient *(the children),* recipient *(to their parents),* instrument *(minibus),* and time *(on Thursday).*

(51) The teacher brought the children back to their parents by minibus
on Thursday.

A major problem in Case theory is precisely the identification of an appropriate set of roles. In addition, it is sometimes not clear whether or not each role can occur more than once in a given sentence: the Valency distinction between complements and adjuncts is useful here, since a sentence involving a verb which takes, for example, a locative complement (e.g. *park)* might also have a locative adjunct, as in (52).

(52) In the village, you can park your car on any street
locative adjunct locative complement

It should be noted that, like Valency, Case relations are not necessarily confined to sentences with finite verbs: they may also be applied in noun phrases to label modifiers of nouns derived from verbs. For example (53) has patient *(water),* instrument *(by pesticides)* and location *(in Europe).*

(53) pollution of water by pesticides in Europe

Semantic functions or Case roles are linked to semantic features (see section 2.6) in that certain functions are usually filled by elements bearing certain features. This can either be general (e.g. agents are usually animate, locations are usually places), or specific to verbs or verb classes (e.g. *eat* requires its patient to be edible, verbs of giving require the recipient to be animate). This link is used to describe both the syntactic and semantic behaviour of words in a sentence, and to make choices between alternative interpretations of individual words. For example, *enter* requires an enclosed space as its direct object: because of this we are able to assume in (54) that the interpretation of *bank* is the building rather than the edge of a river.

(54) The man entered the bank.

2.9.7 Unification grammar

Unification grammar is the name for a number of linguistic approaches which have recently emerged, including GPSG, LFG and Categorial grammar, all of which are discussed below. What they have in common is the use of feature representations (cf. 2.8.3. above), and the formal device of feature unification. The basic idea is that feature structures can be merged if the values of the features are compatible. In this way, structures are built up until representations such as those illustrated in (25) and (26) above emerge. Two feature structures are defined as being compatible either where the two structures have no feature attributes in common, as in (55a) and (55b); or where those feature attributes which are common to both structures share the same values, as in (56a) and (56b). (Unification is Symbolized by the logical 'set union' operator U.)

(55a) [def=yes] U [num=plur,sem=animate] =
the *mice*
[def=yes,num=plur,sem=animate]
the mice

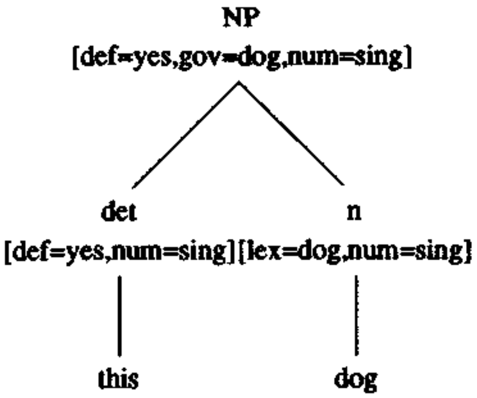
(55b) [def=yes,num=sing] U [sem=animate] =
this sheep
[def=yes,num=sing,sem=animate]
this sheep

(56a) [def=yes,num=plur] U [num=plur,sem=animate] =
those *cats*
[def=yes,num=plur,sem=animate]
those cats
(56b) [def=no,num=sing,gen=masc] U [num=sing,gen=masc] =
un *homme*
[def=no,num=sing,gen=masc]
un homme

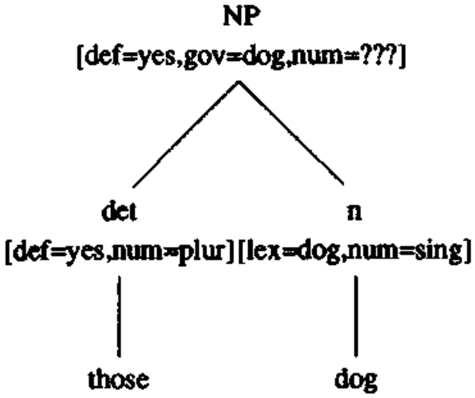
The formalism allows grammars to construct higher-order constituents, for example by merging the structures, as by a rule such as (S7a). This rule permits unification in the case of (57b), i.e. the structure is accepted as well-formed; but in (S7c) unification is blocked, since the values of 'num' are in conflict, i.e. the structure is considered ungrammatical.

(57a) NP[def=X,gov=Y,num=Z] → det[def=X,num=Z] n[lex=Y,num=Z]

(57b)


NP
[def=yes,gov=dog,num=sing]
det
n
[def=yes,num=sing][lex=dog,num=sing]
this
dog

(57c)


NP
[def=yes,gov=dog,num=???]
det
n
[def=yes,num=plur][lex=dog,num=sing]
those
dog

It should be noted that ‘unification’ as understood by linguists is slightly different from the concept of unification found in certain programming styles (see section 3.9). However, unification-based programming languages are highly suitable for implementing (linguistic) unification-based grammars. The main difference is that in linguistic unification, the exact order of attribute-value pairs may be ignored. Other differences are less important.

2.10 Influential theories and formalisms

We conclude this chapter with brief outlines of some linguistic theories which have influenced some MT researchers and may be expected to influence future research in various ways.

2.10.1 Lexical Functional Grammar

Lexical Functional Grammar (LFG) offers a formalism for expressing exactly the kind of feature structure building outlined in the previous section. In LFG there is a sequence of representations: constituent-structure (c-structure), which closely resembles the standard phrase structure representation, and feature-structure (f-structure) which is a feature-based dependency structure representation. The c-structure is given by a standard Phrase Structure Grammar (section 2.8.2 above), except that the rules also stipulate feature equations which must be applied. The application of these equations serves to build up the f-structure. It is achieved by the use of two ‘metavariables: ‘↑’ (pronounced ‘up’) and ‘↓’ (pronounced ‘down’). ↑ refers to the feature structure of the mother node in the tree, while ↓ refers to the daughter node to which it is attached, i.e. to itself. An example of a rule may help to clarify.

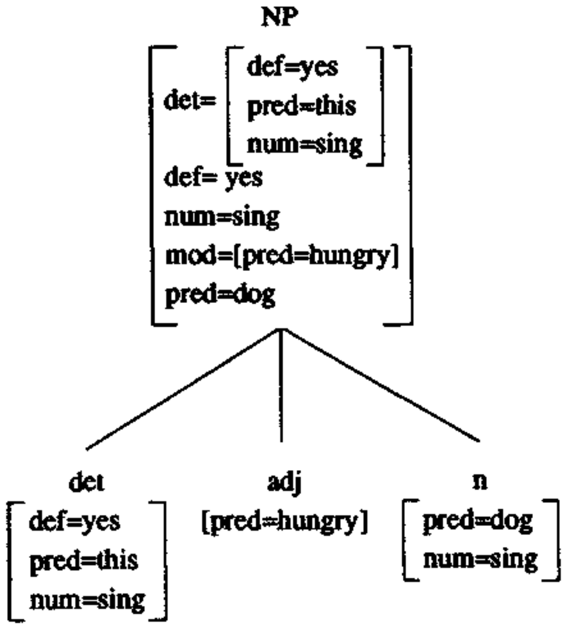
		det		
(58)	NP →	(↑ det = ↓ ↑ def = ↓ def ↑ num = ↓ num)	adj (↑ mod = ↓)	n (↑ = ↓)

This rule states that when building the f-structure for the NP, the values for the features ‘det’, ‘def’ and ‘num’ are taken from the det, and the values for ‘mod’ from the adj. The annotation (↑ = ↓) attached to n means that the entire structure is copied into NP. Notice that the values for ‘det’ and ‘mod’ will themselves be entire feature structures. In constructing the f-structure, a major part is played by the unification of features (as described above): for example, on the assumption that we have lexical entries such as (59), the value for ‘num’ is given both by the feature structure of the det and by the feature structure of the n.

(59) n → *dog* (↑pred=dog ↑ num=sing)
det → *this (*↑ def=yes ↑ pred=this ↑ num=sing)
adj → *hungry* (↑ pred=hungry)

If the equations resulting from the instantiation of the metavariables ↑ and ↓ can be unified then the f-structure for the NP can be built The structure for the NP *this hungry dog* is shown in (60). Notice that all the information we really

(60)



want has found its way up to the mother node, so that the substructures (for det, adj and n) can effectively be discarded.

The LFG formalism for mapping from c-structure to f-structure, coupled with the mechanism of feature unification, has proved a very powerful tool. The facility to build dependency structures from phrase-structure rules appears to give the LFG formalism the best of both approaches, and it has proved very popular for experimental MT systems.

2.10.2 Categorial grammar

An alternative to the constituency representation of category groups (section 2.8.2) is provided by the formalism of **categorial grammar.** In a categorial grammar there are just two or three fundamental categories, e.g. sentence S and nominal N or S, NP and VP; and there are just two functor symbols, a slash ‘/’ and a back-slash ‘\’ indicating what is expected to the right and left respectively. Other grammatical categories (adjective, adverb, etc.) are thus defined in terms of their potential to combine with one another or with one of the fundamental categories in constituency structures. Thus a transitive verb might be defined as ‘VP/NP’ because it combines with a noun phrase (to its right) to form a verb phrase; and an adjective might be defined as ‘NP/NP’ because in combination with a noun phrase to its right it forms a (higher order) noun phrase ($\overline{\overline{N}}$, in $\overline{X}$ theory). In other words, category symbols themselves define how they are to combine with other categories. Combinatory rules are those of simple ‘cancellation’ (or ‘application’) and ‘composition’, as in (61).

(61a) A/B B ⇒ A
(61b) B B\A ⇒ A

(61c) A/B B/C ⇒ A/C
(61d) C\B B\A ⇒ C\A

Categorial grammar was proposed in the early days of MT research for the treatement of syntactic structures. In recent years it has been revived, in part because of parallels with other approaches (e.g. X-bar theory), its compatibility with unification-based formalisms and with principles of semantic compositionality (section 2.10.6 below.)

2.103 Government and Binding

The initial response to the problems of the excessive 'power' of transformational grammar (section 2.9.2) was to impose conditions on the application of rules, e.g. by constraints on transformations, subcategorisation features, selection restrictions, etc. This was the approach in the model known as the **Extended Standard Theory.** The alternative was to develop a theory of conditions on the forms of structures themselves, i.e. a theory of the general principles underlying the construction of grammatical representations. This is the basic approach of what is known as **Government and Binding theory** (GB theory).

GB theory is concerned less with the elaboration of rules for particular languages than with establishing the abstract principles of language itself, with the constraints on individual languages deriving from a **universal grammar.** There is now just one basic transformational rule, expressed as 'Move α'. Its operation in a particular language depends on the specific values attached to α in that language. The particular syntactic structures of a language depend on the values or parameters attached to various 'subtheories' of the model. These subtheories Include (very briefly): the X-bar theory sketched above (section 2.9.4); a theory for assigning 'thematic roles' (cf. Section 2.9.6), θ-theory; a case theory for assigning surface case roles (e.g. nominative, genitive) regulated essentially by government and agreement restrictions; a Binding theory concerning the co-reference conditions for NPs, i.e. involving anaphora, pronominalisation, etc. (sections 2.7 and 2.9.3 above); a Bounding theory defining limitations placed on the movement rule; a Control theory for dealing with the indexing of empty categories and traces (cf. section 2.9.3 above); a Government theory defining the sets of items which may be governed by a category (cf. section 2.8.1 above).

The attraction of GB theory for some MT researchers lies in the postulation of universal principles which could provide the framework for MT systems and it has been the basis for some small-scale projects.

2.10.4 Generalized Phrase Structure Grammar

As its name suggests, this model is an attempt to provide descriptively adequate grammars without using transformational rules of any kind, i.e. be 'strongly equivalent' to a context-free grammar. Since there are computationally effective parsers for context-free grammars (section 3.8), the model of **Generalized Phrase Structure Grammar** (GPSG) has a considerable attraction for computational linguistics.

The characteristic features of the model may be briefly summarised. GPSG makes a formal distinction between dominance and precedence relations; instead of a single rule, such as (62a), there is one rule specifying constituency only (62b), (where the items are unordered) and a general rule for precedence (62c) (where H indicates any head category).

(62a) VP → V NP PP

(62b) VP → V, NP, PP

(62c) H < NP < PP

In this way the parallelism in the category orders of sentences, noun phrases, verb phrases, etc. (cf. the motivation for X theory, section 2.9.4.) is captured in a single generalisation. In GPSG there are general procedures for moving categories and features up, down and across trees, and for ensuring various types of agreement between governors and dependents; these are the 'Head Feature Convention', the 'Foot Feature Principle', and the 'Control Agreement Principle'. Instead of transformational rules GPSG provides metarules to define relations between sets of rules (e.g. those producing active structures and corresponding passive structures), and a 'slash' feature to account for gapping and preposing of *wh* elements (cf. 2.9.3 above), e.g. the formalism X/Y refers to a category X lacking feature (or category) Y, thus VP/NP refers to a VP lacking an expected direct object NP, cf. Categorial grammar (above). Finally, for semantic interpretation the model integrates a version of Montague grammar (see next section) rather than deriving a Logical Form representation as in transformational grammar and GB theory.

2.10.5 Semantic compositionality

The attraction of the semantic theory put forward by Richard Montague has been that it provides a means for integrating syntactic theories and well-established methods of formal logic. The basic principle is that of semantic **compositionality**, i.e. that the meanings of complex expressions (e.g. sentences) are functions of the meanings of their component basic expressions (e.g. words). It is a principle which has now been adopted in a number of linguistic formalisms and theories. A brief outline will indicate the general idea; fuller details are found in Chapter 16 on the Rosetta project, the most significant application of Montague grammar in MT research.

A Montague grammar specifies a set of basic expressions (meaningful units, e.g. words) and a set of syntactic rules prescribing the construction of larger expressions (e.g. sentences) from basic expressions. Expressions are assigned semantic interpretations in relation to the semantic domain of a 'possible world': each basic expression has a basic meaning (a direct association with an 'object' in the domain), and each syntactic rule has a meaning function reflecting the logical (truth) value of the combination concerned. The meanings of expressions (sentences) can thus be related to propositions of an intensional logic formalism and given an interpretation with respect to a 'model' of possible states of affairs, i.e. determining the conditions for a given sentence to be true. Montague grammar assumes that semantic interpretation can be based on relatively simple 'surface' structural information, and this obvious attraction has given it almost paradigm status in contemporary formal semantic theory.

Nevertheless, Montague semantics has been criticised for inadequacies in the treatment of certain linguistic phenomena, notably interrogatives and imperatives, and a number of innovations have appeared in recent years. One of these is **Situation Semantics**, which introduces formal mechanisms for taking discourse and speaker environments into account for semantic interpretation and for providing richer representations of content going beyond truth conditions.

2.11 Further reading

General introductions to linguistics are numerous, from among which we would recommend Lyons (1981a). There are also standard text books dealing with most of the individual topics mentioned here.

For morphology and the lexicon see Bauer (1988), and for morphology and syntax see Allerton (1979). For a general treatment of semantics, see Lyons (1981b), and for text relations, anaphora, etc. Halliday and Hasan (1976).

Dependency grammar is discussed in Herbst *et al.* (1979:32-46), while Schubert (1987) goes into more detail, and is specifically concerned with its application to MT.

For a general and comparative introduction to the various syntactic theories, see Sells (1985) or Horrocks (1987) on LFG, GB and GPSG, and Radford (1988) on TG, GB and X-bar theory

A detailed treatment of Valency and Case with particular reference to Computational Linguistics is given by Somers (1987a). This is the source of Table 2.1.

For an introduction to unification-based grammar formalisms, see Shieber (1986).

The standard reference for LFG is Bresnan (1982), for GB Chomsky (1982), and for GPSG Gazdar *et al.* (1985), though readers may find the secondary references cited above more accessible. Categorial grammar has its origins in the work of mathematical logic; it was first proposed in the MT context by Bar-Hillel (1953); for a recent substantial treatment see Oehrle *et al.* (1988).

For Montague grammar the basic introduction is Dowty *et al.* (1981); its application in MT is described in Chapter 16. For Situation Semantics see Barwise and Perry (1983); its potential for MT has been outlined by Rupp (1989).