[*Automatic Translation of Languages*: NATO Summer School…July 1962]

AN INTRODUCTION TO COMPUTATIONAL PROCEDURES IN LINGUISTIC RESEARCH †

DAVID G. HAYS

The Rand Corporation, 1700 Main Street, Santa Monica, California

PREFATORY REMARKS

EVEN at the 1962 Institute where these lectures were presented, it was hard to find much interest in linguistic research of the empirical sort. Two areas were far more attractive: the design and refinement of translation algorithms, and the establishment of mathematical theory for linguistics. Yet each algorithm either contains or presupposes a body of empirical fact which, in fact, does not presently exist, and theory is pertinent to linguistics and its applications only insofar as it guides the collection and organization of data. During the Institute, it occasionally seemed that the theoreticians were refusing this aid to the empiricists; some of the theorems stated, and some of the interpretations given, suggested that it is theoretically impossible for linguistic theory to guide the collection of data. The theorems are undoubtedly true, but the interpretations are indubitably false.

These lectures, therefore, have to maintain a double argument: that the adoption of systematic procedures for collection and organization of linguistic data is (i) necessary and (ii) possible. Necessary, in the sense that practical applications (such as automatic translation) cannot be developed to the point of usefulness without empirical studies that are unmanageable unless they follow systematic procedures. Possible, in the sense that undecidability theorems do not apply to the situations that arise in practise. Beyond this argument, these lectures are concerned with techniques, with the steps to be carried out in a real program of data collection. Convenience, economy, and avoidance or control of errors are, as they must be in large-scale operations, central questions. Finally, it will be necessary to emphasize, even here, the need for additional theory. The aspects of language that have been studied most widely and formalized most adequately heretofore are not the only aspects of language relevant to automatic translation, and systems of automatic translation that rely entirely on present-day theory have not proved satisfactory.

The written version of these lectures was prepared after the Institute, and the author took advantage, where possible, of what was said there by students and other lecturers. It will be obvious that he is especially indebted to Professor Bar-Hillel, whose work stimulated much more than the construction of counter-arguments on specific points. Insofar as the lectures are based on earlier publications of the same author, they draw most heavily from [1] and [2].

† Presented at the NATO Advanced Study Institute on Automatic Translation of Languages, Venice, 15-31 July 1962.

1. METHODOLOGY AND RESEARCH DESIGN

The courses taught in American high schools include English, History, Geography, and Mathematics. Until courses in 'Human Relations' were introduced, English and Mathematics had the special distinction of being the only courses intended to influence behavior outside the school. And, whereas Mathematics would be expected to influence behavior only in such special situations as the verification of bank accounts, English was and is expected to influence the student's behavior whenever he speaks or writes. Human Relations (and Driver Training) are also intended to influence behavior, the one universally, the other in narrowly defined circumstances. Now, everyone would agree that driving in a way that differs from the methods taught in school is dangerous (driving without using the steering wheel) or bound to be unsuccessful (driving without turning on the ignition). Likewise, doing arithmetic by nonstandard methods (as with such rules as 3+2 = 7) cannot lead to uniformly satisfactory results. These courses teach all there is to know about their subjects. On the other hand, everyone would agree that a Human Relations course does not, because it cannot, teach everything there is to know about dealing with one's fellow men; the contrary proposition is laughable, but so is the proposition that an English course teaches everything there is to know about the use of that language, and yet that proposition is often adopted in computational linguistics, admittedly only in covert versions such as 'the best dictionaries and grammars contain much useful information'.

The best dictionaries and grammars (e.g. [3]) do indeed contain enormous amounts of useful information about English, French, and other languages. To omit them from a list of sources of linguistic data would be folly and lead to regrettable waste of time and money. But they do not contain everything there is to know about English, French, or any other language except possibly some dead language of which only a few sentences remain. Some of the most striking examples of the gaps that can always be found are rules for selection of equivalent words and equivalent grammatical structures in translation; rules for prepositional usage and the kinds of structures (phrases, subordinate clauses, etc.) that particular words can govern; rules for ellipsis; rules for pronominal reference; rules for insertion, deletion, or translation of articles, moods, aspects. There is just not enough in all the dictionaries and grammars of English and French to make possible the immediate construction of a good system for automatic translation of one language into the other, or of a good system for automatic indexing or abstracting of either language, and therefore the school that would be up and doing is bound to be unsuccessful, in the view of the present author, for some time to come.

Mathematics and language resemble one another so closely, as several linguists have pointed out (e.g. [4]), the construction of a grammatically accurate sentence closely resembles the construction of a valid formula, that one may not immediately see why mathematics should be so definitively exposed in its treatises and languages so inadequately dealt with in theirs. The answer is nevertheless immediately clear: languages are invented, learned, modified, and kept unaltered by largely unconscious processes in human interaction. Like other aspects of human behavior, language is a matter of convention, but it must be clearly understood that these conventions are mostly unconscious. When a child comes to school for the first time he already knows a great deal about his language, and what he learns thereafter he learns partly in language classes but partly elsewhere. The purpose of language teaching (that is, classroom teaching of the child's native language) is only to reinforce certain conventions that, according to experience, are not adequately supported by the unconscious mechanisms. Its purpose can be called artificial, in contrast to the natural support of conventions outside the classroom.

Since the child does not learn his native language in the classroom, but only a few somewhat artificial elements of his language, it is not necessary for the sake of such teaching to have a thorough description of the language in systematic form. And the teaching of second languages also depends on practise, on unconscious learning and on knowledge of the first language. Heretofore, full-scale knowledge of natural languages has been an object of at most academic interest, and the academicians (the linguists in this case) have had special interests: phonetics and phonemics, morphology, and to a limited degree syntax. Limitation of attention to these areas avoided conflicts with neighboring disciplines, and brought the reward of quick success. In less than a century the study of speech sounds has reached the point of making automatic speech production and recognition almost realizable. Within half a century, the study of morphology and syntax has made the automatic dictionary possible, automatic parsing possible within certain clear limits and with certain prerequisites, and has made automatic methods of linguistic research almost realizable. On the other hand, areas extending toward what is generally called semantics have not been studied so closely, and the motivation to study 'usage' in detail and *in extenso* has been absent. Some of the conventions that define natural languages have been brought to light, others remain unconscious.

Thus the difference between mathematics and language: the mathematician begins with rules or conventions, explicitly formulated, and works out their consequences, the sentences of his *formal* languages. The native speaker acquires a set of conventions by overt learning, in which case the conventions are at least occasionally conscious, or by covert learning, through listening to sentences and reading them, in which case the conventions may never be conscious. The linguist’s task is to obtain a full statement of these conventions. Those that are conscious can be obtained by asking direct questions or, for well-known languages, by reading reference books. Those that are still unconscious can only be discovered by inference from observation of behavior. And there is the additional difficulty that the so-called ‘conscious rules’ may not control normal linguistic behavior; hence even these must be verified.

The linguist has several kinds of raw material at his disposition. He can use text in the language that he wants to study, and this text may have a natural origin or it may have been produced in response to his questioning. He can use parallel texts in two languages, and again these texts may have natural origins; as when a Russian book is translated into English for publication, or they may have been produced expressly for the linguist; as when the linguist asks an informant to translate a sentence from the linguist's language into the informant's. He can interrogate informants in any way he chooses, for example asking whether a sentence that he utters is correct, or asking whether there are any words in English (if that is the informant’s language) that form plurals but not with -*s* or -*es*, or asking the informant to comment, in the linguist’s own terms, on the sentences of a text. He can, in principle, collect data on the meanings of texts or fragments of text, although the techniques that he would need are not well developed.

Linguistic methodology has variants corresponding to choice of raw material. If the sole object of study is a text, obtained under neutral conditions, then the methodology is called ‘distributional analysis’, and its only objective is to characterize the sentences of the language. The chief advocate of this method, at least in the United States, seems to be Harris [5]. It is obvious that no other objective can be attained, since all that is known about any sequence of sounds or letters is that it does or does not occur in the text. The object studied is always a *finite* text, and there is always a finite characterization of the sentences in a finite text. In fact, there are indefinitely many such characterizations. The hope of eliminating some of them, or even of reducing the set of acceptable characterizations to a single member which could be called the (unique) grammar of the language, has led to the introduction of extrinsic principles, of which the most famous is simplicity. A simple example of distributional procedure leads to a classification of English written consonants. Let + stand for the space between words, and *v* stand for any vowel. The linguist can inquire what consonants occur in the distributional frame +_*v*, that is, after a word space and before a vowel. If the text consulted is large enough, he will find every consonant. Now he takes the frame + *c_v*, where *c* is any consonant. In this frame he finds *H, L,* and *R* very often (and after many different consonants); he finds *M, N,* and *W* less often; he finds that several consonants often occur after *S;* and he finds that a number of other consonants occur rarely, each after only one or two other consonants. He therefore asserts that there are four main frames for the classification of consonants (with respect to occurrence at the beginning of a word): *+S_v,* + *_Hv,* +*_Lv,* and +*_Rv*. In the first frame, *C, H, K, L, M, N, P, T,* and *W* occur (for example, in science, ship, skate, slit, smooth, snuff, spark, stand, swear). In the second, *C, G, P, S,* and *T* (check, ghost, philosophy, share, that); note that the appearance of *H* in the first frame is equivalent to the appearance of *S* in the second. In the third, *B, C, F, G, P,* and *S* (black, clay, flay, glass, play, slay); again, *+SLv* is either *S* in + *_Lv* or *L* in *+S_v*. In the fourth frame, *B, C, D, F, G, P, T,* and *W* (bray, craw, draw, free, grey, prey, tray, write). These observations are not the end of the description of English initial consonant clusters, and the description is not the end of the analysis, but they illustrate the kind of observations that distributional methodology permits.

If the linguist chooses to accept statements about meaning as raw material, he must of course obtain them by interrogation of an informant. Bloomfield’s definition of the morpheme, quoted by Nida [6] and widely accepted, illustrates *form-meaning* methodology: “A linguistic form which bears no partial phonetic-semantic resemblance to any other form is … a morpheme”. If a phonetic description of the language is given in advance, “partial phonetic resemblance” is clearly determinable. Thus, for example, ban-bar, can-car, fan-far, man-mar, pan-par, tan-tar are pairs bearing partial phonetic resemblance; in each pair, the two words have the same initial consonant and the same vowel (in written form). Does ban bear any partial semantic resemblance to bar? It would not be wise to answer ‘no’ too quickly, since ‘He was banned’ and ‘He was barred’ might well be said to have similar meanings. The hyphenation of ‘phonetic-semantic’ in Bloomfield’s definition means that the partial phonetic and semantic resemblances must be correlated, however, and the only evidence of such a correlation is that the same pair of resemblances occurs in several forms. If can and car, fan and far, etc., bear any partial semantic resemblances to one another, pair by pair, they are surely not the same as the resemblance of ban to bar. On the other hand, build-builder, work-worker, walk-walker, and many more such pairs, are asserted to bear a common partial phonetic-semantic resemblance. The first member of each pair does not, the second does end in *-er;* the first member of each pair names an action, the second member names a person who performs the action. Thus *-er* is identified as a morpheme with the meaning *agentive.* Hjelmslev’s commutation test is methodologically a form-meaning procedure. As quoted by Togeby, “Les éléments du contenu ne sont indépendants que si leur interchangement peut entrainer un changement d’expression" [7, pp. 7-8]. Here ‘élément du contenu’ should be understood as a morpheme, semantically defined, and ‘expression’ has to do with the phonetic representation of morphemes. Thus two supposedly distinct meaning units must have phonetically distinct representations, at least in some contexts (‘peut entrainer’).

Note that any use of parallel texts, whether in two languages or in one (in which case one is a paraphrase of the other), is a form-meaning procedure, since the assertion that the texts are parallel is a semantic assertion. Likewise, for reasons to be discussed below, the method with text and editors that will shortly be introduced is a semantic procedure.

A third methodology is *psychological.* Linguistic conventions are effective only to the degree that they are part of the cognitive systems of speakers of the language. Psychological methodology makes the speaker an explicit object of study and undertakes to determine his cognitive structure. Martinet [8] quotes Baudouin de Courtenay’s ‘phonic intentions’ as an example of a psycholinguistic concept and gives an illustration of its application. French has two l’s, one voiced /lak/ = lac, the other unvoiced /pøpl/ = peuple. These two sounds represent the same phoneme, however, because they occur in mutually exclusive distributions, or because they never serve to differentiate words with different meanings, or because, and this is the application of psychological methodology, they result from the same phonic intention. In other terms, they are represented in the cognitive structure of a speaker of French by a single element. But, as Martinet points out, evidence about cognitive structures is hard to obtain. Very few studies can be cited, but Miller’s experiment on speech-perception units is certainly one [9].

It is widely believed that the three methodologies should lead to the same results. Martinet's version of the argument is that cognitive discriminations are made only when commutation, the need to keep sounds separate because they differentiate words, for example, forces them. The distributional methodology can be tied to the others by the argument that if two phones, for example, have complementary (mutually exclusive) distributions, they cannot serve to distinguish any pair of words.

Each of the three methodologies has both advantages and disadvantages. In favor of the pure distributional methodology is the simplicity of collecting the data. Semantic and psychological data can be obtained, but in each case the theory is not well enough developed to permit the establishment of adequate controls. Against distributional methodology is the non-uniqueness of its results. The use of extrinsic principles such as economy is not fundamentally unsound, but economy or simplicity has not been formulated as yet in terms that all linguists can accept, and it cannot yet be demonstrated that any particular set of extrinsic principles is adequate to reduce the many possible distributional grammars based on any given finite text to uniqueness. The discovery that a fixed set of extrinsic principles had such an effect, and the further discovery that the resulting unique grammar corresponded to semantic and psychological findings, would be a linguistic achievement of great importance. The extrinsic principles thus supported, if they in turn could be proved unique, would have the status now sometimes claimed for the principle of economy: they would give a metapsychological characterization of the speakers of the language or languages, of the community of speakers, as they are supposedly characterized by Zipf’s principle of least effort [10].

When a digital computer is available to the linguist, many tasks are conveniently carried out that would be nearly impossible without it. For the distributionalist, every operation involves scanning text for occurrences of an item in a frame, and the number of items and frames that ought to be studied is large. Thus Harris, a decade ago, regarded the distributional method as an idealization of good practise, impossible to apply. Only the beginnings of a system for automatic distributional analysis have been created as yet, but it is no longer impossible to imagine practical use of distributional methodology.

An adaptation of the form-meaning method to the special circumstances of computer use is the ‘cyclical’ method with posteditors [11]. When this method is applied to the study of a language, a crude description of the language must first be constructed. The work then proceeds through a series of stages. At each stage, the existing description of the language is supplied to a computer program which applies it to a sample text. Here the generalizations of the description are converted into analyses of specific sentences. If the description is incomplete, some sentences may be analysed satisfactorily, but others may be given incomplete or incorrect analyses, and some sentences may not be analysed at all. The correctness of each sentence analysis is decided by the editors (posteditors, because they examine the text after the computer program has analysed it). The editors correct any errors they find, and their corrections furnish the raw material for studies that lead to an improved description of the language. The modification of the description ends one cycle and permits another to begin.

If the editors were eliminated, the cyclic procedure would be a distributional method. The data to be studied in each cycle would consist of the sentences not fully analysed. But the editors use their whole knowledge of the language and the subject matter of the text when they consider the correctness of each sentence analysis. Hence the procedure uses form-meaning methodology.

Nevertheless, the procedure avoids asking informants or editors questions of certain difficult kinds. The editor is never asked to make a general statement about the language, but only to comment on specific sentences. Second, he is never asked to provide any sentences; not the editor, but the original author, vouches for the sentencehood of each sentence in the text. All disputes about such sequences as Chomsky’s ‘colorless green ideas sleep furiously’ [12] are thus avoided. Third, the editor is never asked to state the meaning of a sentence, or to say whether two sentences have the same meaning or related meanings, *except* that he may be asked to translate or paraphrase a text, and a text and its translation or paraphrase are expected to have almost equivalent meanings. Many delicate questions, whose answers are too doubtful to provide good support for a linguistic description, do not have to be asked. Moreover, the answers given by editors to the questions that must be asked can be checked. The same text can be edited by different persons and their corrections of the machine-produced analyses compared. The method avoids or controls errors, as a good method should.

Two other characteristics of a good research procedure are economy (in the operation of the procedure, not in the results) and convenience, which leads to greater accuracy and economy. The convenience of a cyclic procedure with posteditors depends in part on the design of the posteditors’ worksheets, the forms on which they are given analysed text for correction; in part on the kinds and quantities of corrections that they must make, and the notational scheme provided for the indication of changes in the machine-produced analyses; and in part on the processes that must be carried out after postediting, the processes that reduce the raw data to an improved description of the language. Economy should be gained by the use of the computer, first to provide tentative analyses of the text, second to manipulate the raw data.

So far, almost nothing has been said about the nature of a linguistic description, or. about the places in the cyclic procedure where it is involved: the computer program that assigns analyses to sentences, the postediting that corrects these analyses, and the data-reduction that modifies the description. A text is a string of occurrences of characters from a finite alphabet. In natural languages, texts can be segmented into recurrent substrings, each indicating the occurrence of a word or morpheme. Part of the description of a language is a list of these substrings; with each must be given a statement of its linguistic properties. The list is a dictionary, and one step in sentence analysis is dictionary lookup. All the rest of the description, and all further steps in sentence analysis, uses the properties of units, not their alphabetic representations.

The structure of a sentence is a set of relationships binding all the word or morpheme occurrences in it together into a whole. Among the competing theories of sentence structure now extant, only one will be introduced and used here. The theory of immediate constituents [12], is its principal rival, and far more widely known, but the author's experience is largely confined to the theory of dependency, which has the same descriptive power in a certain formal sense [13]. According to the theory of dependency [14], [15], [16], [17], every word (or morpheme; but *word* will be used hereafter, as it can properly be used in the study of some but not all languages) depends on one other word in the same sentence. There are two exceptions: one word in every sentence is independent, and relative pronouns, adverbs, and adjectives depend on two other words simultaneously. Each word serves some function for its governor. The number of functions in any natural language is apparently small, and it seems reasonable to postulate that no word ever governs two words with the same function at the same time, again with certain exceptions. Strings of appositives ("John, the leader of the group, the strongest member, the one on whom all others relied," etc.) occur, and it is moot whether, say, all appositives depend on the first in the sequence or each on the one directly before it. Combinations of words with a conjunction ("One, two, three, ..., and *N* are integers") occur, and it seems best to attribute the same function to all conjoined elements and let them all depend on the conjunction.

The fact that one word can serve a particular function for its governor, together with the fact that a second word can govern the same function, do not mean that the first word can serve the given function for the second. Thus 'books' can serve subjective function, and 'am' can govern a word with subjective function, but 'books-am' is not a possible subject-governor combination. The properties of words that determine whether they can serve functions for one another are called agreement properties. If word *X* can serve function *F* for word *Y*, words *X* and *Y* agree with respect to function *F* with *X* as dependent.

Among the properties that must be listed in the dictionary are grammatical properties. The grammatical properties of a word are the functions it can govern, the functions it can serve (as dependent), and the agreement properties involved in any of its potential functions, plus certain other properties concerning word order and punctuation that will not be discussed here.

Given the theory of dependency and a dictionary in which grammatical properties are listed, a computer can determine the structure of any sentence provided (i) the sentence is composed of words in the dictionary, (ii) the word-occurrences in the sentence are used in accordance with the properties shown in their dictionary entries, (iii) certain word-order rules, primarily the rule of projectivity, are obeyed, and (iv) there is no ellipsis. The rule of projectivity [16] requires that all the dependents of an occurrence, all the dependents of its dependents, etc., lie between the first word to left and right that do *not* depend on it. Ellipsis raises problems that will not be discussed here. In general, the structure assigned to a sentence will not be unique; some sentences will be ambiguous.

The program that determines sentence structure, after dictionary lookup, has three parts [2]. The first selects, in accordance with the projectivity rule, a pair of occurrences that can be connected if they agree. The second, given a possibly connected pair, looks for a function that one can serve for the other with respect to which the members of the pair agree. The third changes the list of functions that can be governed by the governing member of the pair, eliminating the function served by the dependent member. The three parts of the program operate in rotation; after the third has operated on a pair of words (or after the second has failed to find a connection between a pair), the first selects a new pair. Properly designed, a program of this type can operate at high speed on a standard computer.

2. POSTEDITING

The posteditor is a linguistic technician, a subprofessional aide to the linguist. He knows the languages that are being studied and he also knows, to a limited extent, the theoretical bases of the research. The tasks that are assigned to him are exacting, but they must be adapted to his special abilities and never allowed to exceed them. They are tedious, hence fatiguing, and therefore must be designed to minimize fatigue. They are time consuming, hence expensive, and therefore must be designed for speed of performance. And the results have to be keypunched and collated with the output of the computer system for subsequent analysis. The relations between man and machine, as the current saying goes, are as complex in this system as in any now operating; the best possible design has probably not been achieved, but fairly good ones have been tried out.

Before text reaches the posteditor, it has been put through automatic dictionary lookup and automatic sentence-structure determination, and it may have been translated as well. If the text has not been processed before, i.e. if it is really new, it contains some items not in the dictionary, some items already known but in grammatical constructions previously unknown for the item. The text is also likely to contain new idiomatic combinations and new grammatical constructions. If it has been translated, the text may contain words with new equivalents or with equivalents that must be chosen according to new contextual criteria. Every new phenomenon gives work to the posteditor.

As the automatic processing of text goes by stages, postediting can likewise be divided into several steps. These steps can be made quite separate, or they can be combined. Thus, for example, worksheets can be printed after dictionary lookup. At that stage, the posteditor would merely fill out dictionary entries for unrecognized items. He can, in fact, be given an alphabetic listing of new items, with one or more examples of their use in the text. Using these examples, together with any approved reference works, he writes entries which are keypunched, added to the dictionary, and used in another dictionary-lookup operation before automatic sentence-structure determination is attempted. This plan minimizes the work to be done at this stage, but does not deal with the problem of new functions for old items. For example, a Russian text may contain noun occurrences governing dative nouns, or instrumental nouns, or particular prepositional phrases, contrary to previous experience. Observation of such phenomena adds to knowledge of the language, and must be handled at some stage. If the posteditor is required to revise all dictionary entries prior to automatic sentence-structure determination (SSD), with the goal of adding to every entry the codes necessary for proper treatment of the item's occurrences in the new text, he must actually determine the structure of every sentence himself without computer help. Thus it seems impractical to correct the dictionary completely before attempting automatic SSD, but convenient to insert entries for new items. Of course, when the rate of occurrence of new items gets small, even this step can be eliminated. Its purpose is to eliminate errors in SSD, which are costly to correct. But when the cost of correcting those errors falls below the cost of repeating dictionary lookup with a corrected dictionary, it is time to drop the preliminary step.

Fixed combinations of words, idioms, have several varieties. Some must be recognized before SSD, because the grammatical properties of the idiom are distinctly different from those of the component items. Such idioms can be recognized, easily and economically, provided that they always occur with their components adjacent in the text and in fixed order. Such idioms as English ‘inasmuch as’, Russian nesmotrya na = despite, can be listed and recognized during or just after dictionary lookup. To find new idioms of this type, the posteditor must read the text, keeping in mind the grammatical properties of individual items and the capacity of the SSD system for recognizing constructions. If the number of such idioms is small, as in English and Russian, it is probably better to identify new ones after SSD and correct the errors they cause.

Thus the stages preceding SSD can well be combined with it in an uninterrupted sequence of machine operations leading to the production of a worksheet on which the posteditor will correct all the errors he finds. The argument can be extended still further, through translation of the text into a second language, so that the posteditor works on each passage of text only once. The system used at the Rand Corporation for analysing 250,000 running words of Russian text combined everything into one worksheet, but experience suggests that the cost of postediting twice, once for the determination of structure, once for translation, would not be much greater than the cost of a single, overall editing, and that the reduction of translation errors by correct determination of source-language structures would be worthwhile. Let us assume the two-step program.

An SSD system can yield no more than one structure, or more than one, for each sentence, according to its design. If it can yield more than one, it can yield a great many. The format of worksheets for postediting must suit the situation.

If the posteditor is to look at a single structure for each sentence and correct it, the worksheet can have a simple columnar format. Each occurrence of a source-language word occupies one line, the source-language text being printed in a vertical column. Occurrence numbers are needed for subsequent collation of different kinds of information about the text, and also for easy encoding of dependency connections. These connections are given in another column, by the computer program in the first instance, by the editor thereafter. Each occurrence has zero, one, or two governors. If zero, the program or editor writes 00 next to the occurrence. If one, the program or editor writes the occurrence number of the governor. And if an occurrence has two governors, both their occurrence numbers must be given. Each occurrence serves some function for its governor, and relatively few functions are distinguished in any language. Another column of the worksheet format is used for designation of functions, and a code assigning a one-letter designation to each function is prescribed. The program or editor writes the code symbol designating the function it serves next to each occurrence. Other information can be printed on the worksheet for the benefit of the editor: the grammatical information obtained from the glossary, one or more target-language equivalents for each item, etc. Table 1 illustrates this format; it is filled out with a Russian sentence and structural coding.

Working in this format, the editor has only to look for errors or gaps in dependency connections and function designations. Upon finding either an error or a gap, the editor

TABLE 1. POSTEDITOR'S WORKSHEET: SINGLE-STRUCTURE FORMAT

Occurrence number	Russian form	Dependency code†	Function code‡	English equivalent
1	Orkestr	02	*S*	The orchestra
2	igraet	00	*P*	plays
3	marsh	02	*C*l	the march
4	garnizona	03	*C*l	of the garrison
5	gussarov	04	*C*l	of hussars
6	malen'kogo	07	*A*	of the little
7	goroda	04	*C*2	town

† Occurrence number of the governor.

‡ *S* = subjective, *P* = predicative, *A* = adjectival, Cl = 1st complementary (direct object), C2 = 2nd complementary (indirect object).

(The author is indebted to E. Unbegaum for this example.)

must correct the structure. What more shall he do? The errors that he finds are caused by errors or gaps in the dictionary, the grammar, or the computer program. There may be similar errors that cause no mistakes in sentence-structure determination, over the text he is editing, but such other errors should not be the object of active search. A dictionary error that has no effect on the SSD operation at a particular point may be discovered at that point, but more or less by accident, and can be noted outside the normal postediting system. The dictionary errors that cause mistakes in SSD are sometimes obvious, sometimes subtle. The entry for a new item contains no information; for Russian, inserting gender, number, and case is usually easy. An idiom that has not been identified and listed should, when it occurs, cause an SSD error, and can be recognized, but giving it a complete grammatical description may not be simple. Noting that an item occurs with a new syntactic function is necessary, since its function is part of the sentence structure that the posteditor is correcting, but adding all the agreement properties necessary for that function can be difficult. In short, correcting the dictionary (and grammar) during postediting would be contrary to the principles of speed and simplicity. Moreover, these errors can be found during the analysis that is to come (see Section 3). Errors in the program, which are certainly possible if the program is intended to find at most one structure for each sentence, are still more difficult to isolate during postediting, or afterwards for that matter. Therefore it seems best to ask the editor for corrections of dependency connections and function assignments, and for *nothing* more at this stage.

If the posteditor, on the other hand, is given a list of alternative structures for each sentence, the format must be quite different. As many as a hundred different structures for one sentence may be offered by the program; the format must be designed to minimize the time consumed in finding the one desired. Listing the sentence a hundred times, each time with one structure marked, is not the answer. One possible answer follows.

One way in which the alternative structures of a sentence can differ is in the choice of the independent occurrence. The worksheet for a sentence begins with a listing of the whole sentence; each occurrence that can be independent in some structure of the sentence is marked, and a reference number is given for each. These reference numbers identify sections in the worksheet (see Table 2). The posteditor chooses the occurrence that should be independent, marks it, and turns to the section identified by the corresponding reference number, say section R-l.

TABLE 2. POSTEDITOR'S WORKSHEET: MULTIPLE-STRUCTURE FORMAT, INITIAL SECTION

```
Orkestr igraet marsh garnizona gussarov malen'kogo goroda.
        (R-1)
```

(Note: Although 'marsh' can be used as a verb, there is no complete structure for this sentence in which the occurrence of 'marsh' is independent.)

Another way in which the alternative structures of a sentence can differ is in the choice of occurrences that depend on the independent occurrence. Here, however, it is necessary to decide what part of the sentence derives from each dependent of the independent occurrence and what function each dependent has. Suppose, for example, that occurrence number 7 is independent, and that occurrences 3, 8, and 10 depend on it, occurrence number 10 being the last in the sentence. Then, by projectivity, occurrences 1, 2, and 4 through 6 all depend, directly or indirectly, on number 3. But occurrence 9 may depend on either 8 or 10. To avoid later difficulties, and the possibility of deciding, by mistake, that number 9 depends on both 8 and 10, the posteditor must decide at once whether 9 goes with 8 or with 10. And in general he must mark a point of division somewhere between any two successive dependents of the same governor.

Section R-l contains a list of alternatives. Each alternative is defined by (i) a set of occurrences dependent on the independent occurrence, (ii) the functions assigned to these dependents, and (iii) the boundaries between their derivation spans. There may, of course, be only one alternative in this section, or in any other. Each alternative is represented by a listing of the whole sentence. The independent occurrence is marked, and is constant throughout the section. The boundaries between derivation spans are also marked. Below each sentence listing, the alternative functions of each dependent of the independent occurrence are listed. These function listings are arranged so that, reading across the worksheet, a set of compatible functions are on a single line. Next to each function symbol is a reference number, identifying a section in the worksheet. The posteditor chooses one line in section R-l; it contains one or more reference numbers. He turns, one by one, to each of the corresponding sections (see Table 3).

TABLE 3. POSTEDITOR'S WORKSHEET: MULTIPLE-STRUCTURE FORMAT, TYPICAL SECTIONS†

```
Section
 R—1      Orkestr igraet marsh garnizona gussarov malen'kogo goroda
            S:7*         *C1: 8
           Cl:7*          * S:8

  8       *marsh garnizona gussarov malen'kogo goroda*
                 *Cl:17                            *
                 *C1:18   *C2:19              _____*
                 *C1:18   *C2:20*             C3:25*
```

† The section numbers are arbitrary. The function symbols are as in Table 1, except: *C*2 = 2nd complementary (indirect object), *C*3 = 3rd complementary. Double underlining marks the main occurrence in a section; single underlining marks the dependents of the main occurrence. Thus in section R-l, igraet governs orkestr and marsh, but either orkestr is subject and marsh is object, or vice versa. In section 8, marsh governs garnizona alone, or garnizona and gussarov, or garnizona and gussarov and goroda. The asterisks mark boundaries between derivation spans; if marsh governs only garnizona, the derivation span of that dependent runs to the end of the sentence, but if it governs garnizona, gussarov, and goroda, the derivation spans of the first two contain only themselves and the derivation span of goroda contains malen'kogo goroda.

All subsequent sections of the worksheet have the same format as section R-l, except that each alternative is represented by the part of the sentence between two derivation-span boundaries in the preceding section. If the original sentence described in the example above were divided

$$* \ 12\underline{3}456*\underline{7}*\underline{8}9*\underline{10}*,$$

then one section of the worksheet lists occurrences 1 through 6, another occurrences 8 and 9, another occurrence 10. If occurrence 3, for example, can have two different functions as dependent of occurrence 7, and if its capacity to govern other occurrences depends on its own function, then two sections must be used for occurrences 1 through 6.

By selecting one alternative in each section, the editor chooses a governor and function for each occurrence in the sentence. But the posteditor need not refer to every section of the worksheet; each alternative that he selects leads to certain sections, and he must follow these leads. When he has no leads to follow, he has finished with the sentence, whether he has looked at all sections or not; if not, the untouched sections belong exclusively to false analyses.

The postediting plan just outlined requires the editor to choose the one best structure for each sentence. Another plan would require him to verify that every structure proposed by the program is syntactically valid. According to such a plan, he would have to look at every alternative in every section of the worksheet. Worse, he would have to abandon all knowledge about the language and subject matter except syntactic knowledge. But abandonment implies separation, and separating syntax from all other linguistic knowledge is evidently very difficult. If a sentence has two syntactically valid structures, one correct for that sentence and the other not, then there must exist in the language some other sentence for which the second structure *is* valid. The editor could be asked to provide that sentence, but the text can be asked also. That is to say, if the posteditor marks the one correct structure for each sentence, the text analysed will eventually provide examples of all correct structures. *Eventually* does not mean within some limited period, however long, and the whole argument is subject to questions about the limitations imposed by style and subject matter, but the point is that verifying every machine-generated structure for each sentence has no overwhelming advantages and does have some severe disadvantages.

Even if the posteditor is looking for the one best structure of each sentence, it is possible that he will not find it among those offered by the program. Suppose that in a certain section he fails to find the alternative that he requires. He must add a line to that section showing the occurrences that he wants to depend on the main element in that section, the function of each dependent, and the boundaries between their derivation spans. Next he must look for appropriate reference numbers for the dependents he has indicated. He may find all of them in other alternatives in the same section, he may find some of them there and some of them in other sections, or he may not find all of them anywhere. The sections can be arranged in order by occurrence number of the main element to simplify the search procedure. The editor can even use a section with erroneous boundaries, but then at some point he will be obliged to correct himself. Suppose, for example, that the span $* \ 1 \ 2 \ \underline{3} \ 4 \ 5 \ 6 \ *$ of the sentence cited above should actually end after the fifth occurrence. In the section devoted to this span, there may be an alternative in which occurrence 3 has the correct dependents—say, 2 and 4—and the only span error concerns occurrence 6. This alternative has the form $*1\underline{2}*\underline{3}*\underline{4}56*$. The analysis of the first span can continue normally. The section devoted to the span $*\underline{4}56*$ may contain a semi-correct alternative, say *$\underline{\underline{4}}$*$\underline{5}$6*. Choosing this alternative, the posteditor would normally continue by selecting an analysis of the span *$\underline{5}$6*, but since occurrence 6 does not properly belong in the original span, in this case the posteditor must stop. In other cases, he would have to add a section to bring an occurrence into a span where it belonged. Remembering that the posteditor's object is simply to indicate the correct governor and the correct function for each occurrence, and that derivation spans were introduced solely to reduce the opportunity of error, will make finding the simplest procedure for adding structures perfectly obvious.

When postediting has been completed, the new information is keypunched and collated with the original text, the information obtained by dictionary lookup, and that produced by SSD. Whatever the form of the worksheet, the keypunched information can be reduced to a set of specifications which, added to SSD output, determines exactly one structure for each sentence in text.

The next stage in automatic processing is an idiom lookup. This time the fixed combinations of words that must be recognized need not occur in fixed order, but they must be connected in the structure of the sentence. It is not yet possible to describe in detail the kinds of properties that must be ascribed to these idioms, and to individual words when they occur outside these idioms, but it is certain that properties similar to syntactic properties will have to be assigned to them, and that an operation similar to SSD will have to be carried out on them. An old-fashioned term for this process is the determination of semantic compatibility. Its purpose is to eliminate the ambiguity remaining after SSD, reducing the alternative structures of sentences from dozens or hundreds to one or two each, and to avoid ambiguity in the selection of target-language equivalents. It will have to be postedited, but first it must be defined in more detail than seems possible for the moment. In Section 3, it will be assumed that there exist certain source-language units (and these may be idioms or words occurring not in idioms) to which correspond sets of target-language equivalents (idioms or single words), and that the posteditor, at some stage, chooses the best equivalent for each occurrence of a source-language unit. More circumstantiality would be premature.

3. ANALYSIS OF POSTEDITED TEXT

Computational systems for linguistic analysis are still no more than semiautomatic. They rearrange and summarize data in accordance with rather simple, specific instructions from the linguist, who must draw his own conclusions from the results. Hence, regrettably, the design of analytic procedures must take into account the linguist's momentary state of knowledge. The more he knows about the language he is studying, the more elegant and powerful can his analyses be. The start that has been made on fully automatic analysis is described in Section 4; that work is for the future, however, whereas the methods introduced in the present section are ready, at least in principle, for immediate use.

The classic tool of linguistics is the concordance. Each entry in a concordance consists of a key occurrence together with part of its context. Ordinarily, every occurrence in a text appears as the key occurrence in one entry in the concordance of the text; a given occurrence may also appear in other entries, as part of the context of other occurrences. If the context included in each entry consists of the occurrences immediately preceding and following the key occurrence, the concordance of a test is three times as large as the text itself, not counting the location indicators that are usually added to each entry in a concordance. Sometimes selective concordances are prepared, omitting, for example, all entries with function words (prepositions, conjunctions, articles, etc.) as key occurrences.

When a concordance is prepared from postedited text, the context of an occurrence can be defined in terms of structural connections instead of linear order. Thus an entry might consist of a key occurrence together with its governor and all its dependents. The function served by the key occurrence and those served by its dependents can also be included. When a dictionary is available, other information can be added to each concordance entry. The exact form of the key occurrence can be replaced with a canonical form: plural nouns with singular, all verb forms with infinitives, and so on. Grammatical information can be used as well.

The arrangement of a concordance is always systematic, since the text itself could otherwise serve as its own concordance. The system chosen depends on the information available in each entry, of course, and on the purpose the concordance is to serve. Using terminology that is familiar in computing manuals, one can speak of major, intermediate, and minor sorting variables. In a telephone book, the major variable is family name, the intermediate variable is first name, and the minor variable is middle name; in the so-called 'Yellow Pages', the major variable is name of product or service, the intermediate variable is firm name, and the minor variable, used only occasionally, is branch or dealer name. In a concordance, the major variable may be the form of the key occurrence, intermediate the form of the preceding occurrence, minor the form of the following occurrence. Some other arrangements are:

Form of key occurrence—form of governor of key occurrence

Form of key occurrence—form of a dependent of key occurrence

Grammatical characteristics of key occurrence—grammatical characteristics of governor of key occurrence

Grammatical characteristics of key occurrence—grammatical characteristics of a dependent of key occurrence

Function of key occurrence (as dependent)—grammatical characteristics of key occurrence—grammatical characteristics of governor of key occurrence

(each arrangement, of course, names the major sorting variable first, then the intermediate variable, then the minor variable, if any). This list is not complete, and a whole series of other arrangements could be defined if such characteristics as the target-language equivalent of the key occurrence were included in each entry. Each arrangement has some use. Unfortunately, the arrangement best adapted to the study of any difficult problem is least adapted, in general, to the study of diverse problems. Concordances have been published in the past [18], each the product of immense manual effort, and concordances are still being published, now often with the aid of a computer. The usual arrangement of a published concordance is by form of key occurrence as major variable and occurrence order as intermediate. This arrangement, most readily understood and used by everyone, is ill adapted to almost any particular problem that comes to mind. For example, a study of grammatical agreement rules for syntactic functions would be most tedious with such an arrangement. Since standard computer programs can make concordances quickly and with any list of sorting variables desired, it seems that the publication of concordances will have little influence on research in the future. Once a text has been put on magnetic tape for computer input, and especially if postediting data is included with it, a scholar with a new research idea can obtain the tape, name his sorting variables in accordance with his plans, and obtain a concordance (very likely a selective listing instead of a complete survey of the text) with little effort, expense, or delay.

The concordance, although a classic tool of the linguist, is not the most powerful. Given postedited text and a computer, the linguist can call for crosstabulation by categories of many useful kinds. To make a crosstabulation involves the selection of two or more variables and the definition of units to be listed or counted. The result is a matrix in which each row represents a value of one variable, each column represents a value of another variable, and each cell contains the number of units characterized by the values of the corresponding row and column.

As a concrete example, let us consider the study of grammatical agreement with respect to one syntactic function, say the subjective function in Russian. We may suppose, for the purposes of the example, that the linguist has already analysed each Russian form (occurring between spaces in text) as composed of a stem and an inflectional suffix; he suspects that the endings are involved in agreement. Oversimplifying for clarity, let us suppose further that each form contains at most one suffix; every form that contains no suffix will be treated, temporarily, as if it contained a zero suffix, and every form will be said to contain some stem. The units to be counted are pairs of occurrences, each consisting of a governor and a dependent related by subjective function. The two variables of the cross tabulation are (i) inflectional suffix of the dependent and (ii) inflectional suffix of the governor. The matrix will have one row for each suffix that occurs in a subjective dependent and one column for each suffix that occurs in the governor of a subject. The counts are based on a certain text; each cell contains the number of occurrences in that text of subject-governor pairs with a certain suffix in the dependent, a certain suffix 'in the governor. Every such pair is counted just once in some cell of the matrix. Since the counts in the matrix are based on a finite text, a sample of all the Russian text ever written or to be written, they are best regarded as estimates of the counts that would be obtained from an indefinitely long text. As estimates, they are subject to sampling error, deviations from the counts that would truly describe the language but can never be obtained. The treatment of sampling error is a statistical problem that will not be discussed here. A substantial literature has been devoted to the analysis of cross tabulation matrices with sampling error, but linguistic applications are just beginning [19]. In the following paragraphs, some of the possible analyses will be discussed in terms that presuppose errorless data. The linguist who intends to perform such analyses should consult the statistical literature, or a statistician, before proceeding.

The linguist who analyses the Russian data of our example begins with a hypothesis drawn from experience with many languages concerning the relation between morphology (the suffixes, the classes of stems with which they occur, etc.) and syntax. There exist, according to this hypothesis, syntactic categories in terms of which the agreement rules for the subjective function are relatively simple. He does not suppose, however, that each suffix belongs to exactly one syntactic category, nor that each distinct suffix belongs to a different category. The possible complexities of the relations between morphology and syntax guide his analysis.

First, there may be two or more suffixes that are syntactically equivalent. Looking among the dependents first, such suffixes would be represented in the matrix by two identical rows. More precisely, since two suffixes can be syntactically equivalent even if one is more frequently used than the other, the rows should be proportional. If each entry in the matrix is divided by the sum of all entries in its row, then rows corresponding to equivalent suffixes should be identical. For simplicity in further analyses, identical rows can be combined by adding together the entries in each column; one row, representing a set of syntactically equivalent suffixes, replaces several in the matrix. The same analysis, leading to a similar reduction of the matrix, is performed on the columns. In the example, the singular suffixes of different declensions would be equivalent, and so would the plural suffixes.

Second, there may be some suffix that is used in two syntactically different ways, each corresponding to the use of some other suffix. The ambiguous suffix would be represented by a row equal to the sum of two other rows (or of three or more other rows). In the example, an ending that is singular in one declension and plural in another would have this property. Since the suffix may be singular in a high-frequency declension and plural in one of low frequency, or vice versa, its row need not be equal to the sum of two others, even after division of every entry by row sums, but only to some linear combination. Upon finding such a row, which corresponds either to a single suffix or to a set of syntactically equivalent suffixes, the linguist must call for a supplementary analysis. Let *X* be the ambiguous suffix, *Y* and *Z* the two suffixes whose range it covers. The subject-governor pairs in which *X* appears are sorted into two groups: those with governors that also govern *Y,* those with governors that also govern *Z.* The question is whether *X* occurs with the same or different stems in the two groups of occurrences. If the stems are different, they can be assigned to different classes, and indeed they may well belong to different morphological classes already because they take different sets of suffixes. With a stem of one class, *X* belongs to one syntactic category, that of *Y,* and with a stem of the other class *X* is equivalent to *Z.* The ambiguity of suffix *X* is reduced morphologically and its row in the matrix can be combined with the rows of *Y* and Z, reducing three rows to two. On the other hand, if the stems in the two groups of occurrences are the same, the ambiguity is not eliminated morphologically although it can be eliminated syntactically. The same reduction of the matrix can nevertheless be performed. Naturally, a similar analysis and reduction is carried out on the columns of the matrix.

A third possibility is that some suffix has one syntactic use that is unique to itself, another use equivalent to the use of some other suffix. The zero suffix appears in some Russian declensions, where it is syntactically equivalent to other suffixes; it also appears, for example, in the personal pronouns ya = I, my = we, etc., making it the unique first-person suffix. It follows that the corresponding row in the matrix is equal to a linear combination of other rows plus a remainder. As in the previous situation, a supplementary analysis is required, and in the example it will lead to recognition of several morphologically resolvable uses for the zero suffix.

The linguist would like to continue the analysis until the matrix contained only one nonzero entry in each row and each column; he could then call each row a simple syntactic category. In the Russian example, the matrix has 14 rows and 14 columns at that stage. Although the analyst cannot label his matrix in this fashion, we can name them by specifying person, number, and gender. Writing 1, 2, and 3 for first, second, and third persons, m, f, and n for masculine, feminine, and neuter genders, and s, p for singular and plural numbers, the columns (and rows) are labeled 1ms, lmp, lfs, lfp, 2ms, 2mp, 2fs, 2fp, 3ms, 3mp, 3fs, 3fp, 3ns, and 3np. To reach this point, the linguist may be forced to consider a row or column which contains two nonzero entries, neither corresponding to the unique nonzero entry in any other row or column, but such considerations should wait to the end of the analysis. Having reached this point, the linguist can reconsider. He should discover that gender in Russian nouns is determined by the stem, not by the suffix; that no Russian verb has a suffix exactly identifying person, number, and gender; and so on. He will probably not retain the separate syntactic categories 1ms, 2ms, and 3ms for verb suffixes, since every suffix that belongs to any one of these categories either belongs to all three of them or to one of them and also to one or two others.

The final stages of the analysis belong to the linguist, who can call on many different criteria in sharpening and systematizing the classification. The earlier stages, beginning with a large matrix with entries that always, in practise, must be subject to sampling error, can be programmed and carried out on a computer. From the preparation of concordances to the construction of crosstabulations to their analysis, the computer has taken a larger and more sophisticated part in the research process. With such a perspective, the scholar who limits its role to sorting and listing appears to be wasting his resources.

The illustration of crosstabulation analysis just presented was drawn from the classic domain of the relations between morphology and syntax. Since classic methods have been highly productive in this domain, new ones are not likely to add much, and the illustration is to that extent misleading, but it was chosen for clarity, not as representative of the problems to which the crosstabulation method should be applied. This method, better, this family of methods, is perfectly general for the class of problems involving relations among two or more variables. It is not surprising, therefore, to find many possible applications for it in linguistics.

There is, for example, the problem of syntactic classification of stems. Given a syntactic function for which the morphological agreement rules are known, are there further rules determining the classes of stems that can occur in words connected by this function? Again the rows of the matrix correspond to dependents, the columns to governors, and the entries in the cells are occurrence counts. But this time the stems, rather than the endings, appear as row and column labels. Since the number of stems is ordinarily much larger than the number of suffixes in a language, the size of the matrix will be larger for this analysis, and the entries in the cells will consequently be smaller on the average. In fact, unless the text in which occurrences are counted is extremely large, almost all the cells will contain zeros and almost all the nonzero counts will be ones. The analysis is more sensitive, but not impossible. Unlike the determination of morphological agreement rules, the study of stem-stem agreement rules in syntax is just beginning, and it may be hoped that new analytic methods will accelerate it.

The classification of texts according to subject matter, and the corresponding classification of vocabulary items, although it is perhaps of more interest in information retrieval than in machine translation, is another example of a problem to which crosstabulation analysis can be applied. The first stage of the analysis uses a matrix in which rows and columns are labeled with the titles (or identification numbers) of books, articles, or abstracts; here the set of row labels is identical to the set of column labels. Each cell contains, e.g. the number of words that occur in each of two documents. Analysis of this matrix yields a classification of the documents. The next stage is a consideration of the vocabulary of the whole library, using as a description of each word the number of times it occurs in each class of documents. Each word is characterized as specific to a class of documents, hence by inference, to a subject of discourse, or as specific to a range of document classes, or as general. A few such studies have been performed, although so far only on a limited scale [20]. One current difficulty is that, for linguistic reasons, the word is most unlikely to be a suitable unit, yet the units that ought to be used cannot now be identified and listed.

The same difficulty unfortunately applies to the study of rules for choice of equivalents in machine translation; the units that ought to have stable equivalents are still unlisted. On the other hand, the search for rules of equivalent choice may lead to the identification of appropriate units. Taking the word as the starting unit, and one word at a time, the analysis can consider several variables: the functions the word serves in its various occurrences, and for what governors; the functions served by dependents of the word, and the words that serve them; and, of course, the translations of the word itself as well as the translations of related occurrences. (It is assumed here that posteditors have supplied at least a provisional translation of the text being studied, including specific translations for individual words or word groups, in addition to indications of the structure of source-language sentences.)

The first step in analysis of a particular word might be the construction of a matrix in which each row represents one equivalent of the word and each column one function it serves. If each column contains exactly one nonzero cell, the equivalent of the word is determined by its syntactic function and the analysis is complete. Otherwise, the next step is to take each function served by the word and construct a matrix for that function with rows again representing equivalents and columns representing all the words that govern that function. If each column of this matrix contains exactly one nonzero cell, the governors of the word can be classified according to the equivalents that they determine, and again the analysis is complete. However, other interesting situations are possible. If one equivalent is limited to a few governors while the others appear with many different governors, the word under study can be considered to form a fixed combination with each of the few governors that determines an equivalent, and the fixed combinations can be taken as translation units. On the other hand, if most equivalents are determined by particular governors whereas one equivalent appears with many different governors, the latter equivalent can be set aside and the governors classified as before. The diffuse equivalent may occur, for example, when the word under study appears with a particular dependent or with one of a particular class of dependents. The analysis continues as necessary, with matrices having, as column labels, words that occur as dependents, translations of related words, etc. At each stage, fixed combinations, in the target language as well as in the source, can appear and be noted.

In the selection of equivalents, several hypotheses always have to be considered and the research procedure should take all of them into account. One is that the individual word is too small a unit, i.e. that it must be translated as part of a fixed combination, at least in some of its occurrences. Another is that local conditions (syntactic function, type of governor or dependent) determine equivalent choice. A third hypothesis is stylistic or accidental variation; posteditors can choose different equivalents for different occurrences of a word without having any clear or explicable reason. Fourth, the hypothesis of subject field has to be remembered; a word can have different translations in different articles, even if there are not distinctive differences in local context, because of differences in subject matter and in the habits of authors in different disciplines. And a fifth hypothesis, although probably not the last that could be found, is that the word, in some of its occurrences, is an abbreviation of a fixed combination. If a word sometimes appears in one or more fixed combinations, and if one of those combinations occurs near the beginning of an article, it is possible that subsequent occurrences of the word stand for the whole combination and must be translated with the target-language abridgment of the combination. Harris's paper on 'discourse analysis' [21], was concerned with such problems, and K. E. Harper's (unpublished) observation that Russian nouns are modified more often near the beginning of an article than further on suggests that the phenomenon is widespread. Unfortunately, systematic procedures for analysis of linguistic relations that span more than one sentence are still undeveloped and cannot serve machine translation, but research procedures aimed at discovery of equivalent-determination rules can take this hypothesis into account.

4. AUTOMATIC LINGUISTIC ANALYSIS

The object of linguistic analysis is to characterize the sentences of a language. The language is not a finite text, but a finite text is all that the linguist can ever study. Before his analysis, he can be sure of nothing about the language, but he can only begin if he is willing to hypothesize some properties for it, and he naturally chooses properties that are universal, or at least widespread, in languages already known. These properties constitute his theory of the structure of natural languages; from the theory, he would like to derive a set of procedures that will yield a concrete description of the finite text that he is able to study and, hopefully, characterize the language beyond that text. Certain linguists, adopting the distributional methodology that excludes semantic and psycholinguistic data, have raised the problem of purely automatic 'discovery procedures'. Since such procedures could be programmed and applied, by means of a computer, to very large quantities of text, their importance for the future of linguistics seems great. The speculation that distributional, form-meaning, and psycholinguistic methodologies would yield virtually equivalent structures for any natural language makes the possibility of automatic linguistic analysis even more attractive. The beginning that has been made and the prospects for further work are the subject of this section.

The first step in the analysis of a new language, after texts have been recorded, is the reconstruction of its vocabulary. (A certain normalization of its alphabet, whether of letters or of phones, sound units, may be needed, but can be passed over here.) A universal feature of natural languages is that groups of alphabetic characters form units with which the rest of the language is constructed. Each such group is a *morph* and represents a *morpheme.* In general, morphs occur one after another in text; there are many exceptions to this rule, as in languages where the consonants of a word belong to one morph and the intervening vowels to another, but it is better to oversimplify than to introduce all the important but complicated qualifications that a useful discovery procedure would have to accept. The problem, then, is to segment a text into morph occurrences. In some texts the segmentation is marked by the author, who spaces after each morph. More often, short strings of morphs are bounded by spaces and have to be segmented internally (as in printed English, French, German, Russian, etc.). In many spoken languages and some written ones, the strings of morphs between spaces or silences are long. The silence or blank space can be taken as an absolute morph boundary (omitting qualifications as usual), and the sequences between can be sorted out. In printed English or Russian, for example, there will only be a few thousand different sequences between blanks in a text of fifty or a hundred thousand running words, whereas in spoken French an equivalent text might contain only a few silence-to-silence sequences that occurred more than once each.

A procedure has been proposed by Harris [22] for segmentation of morphs. Take one unit from silence to silence or from space to space, say $x_1\, x_2\, x_3 \ldots x_n$. Here each x_i is some character of the alphabet. Consider the list of all silence-to-silence units that begin with the same character x_1, and determine the variability of second-character choice among these units. If the next character in every unit is the same, variability is nil; if all characters of the alphabet occur as second character following x_1 each equally often, variability is maximum. The observed variability, say V_1 is noted. Next V_2 is determined; it is the variability of third-character choice among all units that begin with the sequence x_1x_2, then V_3, V_4, and so on. Plotting V_i against i, the analyst expects a declining curve, because there are relatively few morphs in a natural language as compared with the number that could be constructed using its alphabet. English, for example, could have $26^5 = 11{,}881{,}376$ different five-letter words, about twenty times as many words as in its entire vocabulary. Some of the words that do not occur are forbidden for phonological reasons, e.g. * *mxzntzz* or * *qqq*. Others are phonologically possible but simply not used, e.g. **maser* (until recently) and **thaser* (until it becomes acronymic, or otherwise enters the language). There are many morphs that begin with any x_1 fewer that begin with the same x_1 and any x_2, and so on, so that *Vi* falls until the end of the morph is reached. If the morph x_1x_2... x_k can be followed by relatively many other morphs, V_k is *larger* than either V_{k-1}, or V_{k+1} Hence a relative maximum in V_i often marks the boundary of a morph. Exactly the same calculation can be performed from right to left, this time giving variability of next-to-last character among units ending with x_n. The relative maxima given by the two calculations should mark the same boundaries, but in a language where each space-to-space unit consists of one stem morph followed by zero or one suffix morphs the right-to-left calculation should give more obvious results.

The Harris procedure cannot work in a language that uses every phonologically allowable sequence to represent a morph, but no such language is known. There may be a few languages with so few phonemes that a large proportion of the allowable sequences are used, and if there are the procedure would be inefficient for them. Even in languages where it is most efficient, the procedure is not likely to find all the morph boundaries in a text, and it is likely to mark some that subsequent analyses do not retain. On the one hand, if vowel sequences are narrowly restricted by phonological rules, and consonant sequences likewise, but vowel-consonant sequences relatively unrestricted, there will be a relative maximum in V_i at each transition from vowel to consonant or vice versa, whenever there are two phones of the same class before the transition. These phonological maxima will be large only in peculiar cases, however, and can therefore, perhaps, be disregarded. They can be eliminated if adequate phonological analyses are performed in advance of the Harris procedure; V_i can then be calculated as the ratio of observed variability to phonologically allowable variability in each position. On the other hand, if a morph can be followed only by a few other morphs (for example, if every verb stem must be followed by one of two or three suffixes), V_i will not have a relative maximum at its boundary. In spite of these qualifications, an experiment with English showed the procedure to be more than 80 per cent effective,† and that figure could be improved by the use of refined technique and a larger sample.

The object of a grouping procedure like Harris's is not to find the morphs in a language, but to find a set of units with which analysis can proceed. If the procedure is 80 to 90 per cent accurate, as measured against the results of a psycholinguistic or form-meaning analysis, the tentative morphs that it produces can be submitted to further distributional study. Procedures lately suggested by Lamb and Garvin require a text with marked morph boundaries as input, but it would be easy to adjust their methods and other methods of the same general kind, so that one output would be a revision of those boundaries.

Given a tentative list of morphs, the next problem is to determine their mutual relations, i.e. their distributions relative to one another. One aspect of the problem is classification

† Unpublished seminar paper by C. Chomsky, cited in [23].

of the morphs; two morphs belong to the same class if they have identical distributions. The other aspect of the problem is the listing of constructions, i.e. of admissible combinations of morphs. In dependency terms, constructions are characterized by functions and agreement requirements. In any terms, the initial difficulty is that there are too many morphs and too few occurrences, even in a large text. The distributional regularities that will be summarized in a construction list do not appear until equivalent morphs are classed together, since there are not enough occurrences of individual morphs to bring out the regularities. The judgment that two morphs have equivalent distributions, hence can be assigned to a single class, cannot well be based on their few occurrences in a text, since no two morphs have similar distributions in terms of linear context and individual morphs. The judgment would be easier if it could be made in terms of classes of morphs and constructional context, but at first neither constructions nor classes are known. The deadlock can only be broken by an iterative approach that starts with a crude classification and tentative list of constructions, gradually refining the two together.

Garvin [24] begins with a rough classification of morphs based on gross features of their distributions. He requires that two kinds of boundaries be marked in the text; roughly speaking, these are word and sentence boundaries or morph and utterance boundaries. The small-unit boundaries determine the items to be classified, and the large-unit boundaries furnish the distributional criteria for an initial classification. Each occurrence of a morph is characterized as adjacent to and preceding a major boundary, hence *final;* as adjacent to and following a major boundary, hence *initial;* or as not adjacent to a boundary, hence *medial.* Considering all occurrences of a morph simultaneously, it can be assigned to one of seven categories accordingly as it occurs in all three positions (class IMF), only two (classes IM, IF, and MF), or one (classes I, M, F). Since counts of occurrence in each position can be made, subtler classification is possible. But now a category symbol can be assigned to each morph occurrence in the text and a search for constructions started. Garvin has suggested some techniques for the search and is continuing his investigation of this problem.

Lamb's procedure [25] is to form tentative constructions first, deriving tentative classes from them. He argues that a syntactic relation is a limitation on the variety of morph sequences that occur; hence local restrictions reveal (or may reveal) relations. For each morph in a text, he determines the variation in its neighbors, taking those to the right and those to the left separately. The morph with least variation is temporarily assumed to form a construction with its neighbor. Thus, if morph M_i is almost always *followed* by some morph M_j, every occurrence of M_i is assumed to be in construction with the following occurrence, whether M_j or some other morph. (If M_1 had regularly been preceded by some M_j, the construction would consist of M_i and its left neighbor). Not just the single morph with least variation among its neighbors, but all the morphs with variation below a threshold are treated in this way. Each such tentative construction is given a name, and the calculation of variation coefficients is repeated, with different results because the constructions now appear in the text instead of their constituents.

Classes are not formed until second-order constructions appear, in which one of the constituents is itself a construction. Then all the morphs that occur as partners of M_i in the first-order construction when the construction is in turn the partner of M_j are classed together. The rationale is that these morphs have the same distribution to a second degree of approximation, and that cases of third-degree differences in morph distribution are rare. The morphs that are classed together take the same partner and with it form constructions that take the same partner; the partners of the second-order construction could vary with the morph contained in the first-order construction, but experience says that such variation is unlikely in natural languages.

Since the criterion by which constructions are formed is approximate, the procedure must allow for dissolution of constructions. Lamb recalculates variation coefficients whenever a construction or class is formed. Every occurrence in text of a tentative construction is replaced with an arbitrary symbol standing for the construction, and every occurrence of a morph in a tentative class is replaced with an arbitrary symbol standing for the class. Using this text, the variation coefficients are recalculated for individual morphs, for constructions, and so on, and it may happen that the constructions originally established, when variation coefficients had to be calculated on the basis of individual morphs, will now be replaced by others. Lamb's example is that a preposition, whose following neighbors include articles, nouns, and adjectives (in English), may at first be put in construction with that neighbor so that, e.g. (in + the) is marked as a construction. Later, when a class of nouns begins to develop, the coefficient of variation among following neighbors of 'the' will decrease until (the+noun) is formed and replaces 'the' as partner of 'in'.

The units that have been called morphs in the description of Lamb's procedure might be the product of Harris's procedure, they might be the units occurring between blanks in printed text, or they might have been obtained in some other way. If they are forms, as found between blanks, they can be segmented into stems and endings by some procedure like Harris's and submitted to a morphological-agreement analysis by crosstabulation of the kind discussed in Section 3. If they are tentative morphs, from Harris's procedure, they need to be checked. One step is to look for constructions that do not involve classes when Lamb's procedure ceases to be productive. Those constructions can be taken to reassemble morphs that the Harris procedure erroneously dissected. Again, any class of tentative morphs can be inspected for phonic or graphic similarity. If Lamb's procedure gives a class in which every morph ends with a particular letter or group of letters, that ending is possibly a morph; if the same ending occurs in several distributional classes, it should certainly be recognized. Thus syntactic criteria can be applied to readjust morphological findings, and the 80 to 90 per cent accuracy of Harris's method is not a final result by any means.

Lamb is not concerned with dependency theory, but if his procedure is accepted up to this point it can be extended to the determination of dependency connections. Consider a construction, say $X = Y_iZ_j$. In each occurrence of the construction, Y_i and Z_j are either morphs or constructions. The identity of the construction over all of its occurrences is established by two facts: the construction has a homogeneous distribution, and all the Y_i (or, instead, all the Z_j) belong to a single class. If all the Y_i belong to one class, the Z_j may belong to one class or to several classes; we can consider the case in which they belong to a single class, since in the other case the following procedure would merely be repeated for every class of Z_j. For the present, suppose that each Z_j is either a morph or a construction whose members are morphs (it could also be a construction whose members are constructions; that case is treated below). Thus, either $Z_j = M_j$, a morph, or $Z_j = M_{j1}M_{j2}$, a construction of two morphs. If Z_j is a morph, Y_i and Z_j are connected by a dependency link. If $Z_j = M_{j1}M_{j2}$, then Y_i is linked by dependency to either M_{j1} or M_{j2}. Form two sets; the first is composed of all the M_j such that $Z_j = M_j$ for some j, together with all the M_{j1} such that $Z_j = M_{jl}M_{j2}$ for some j, and the second set contains the M_j's and M_{j2}'s. Calculate a coefficient of variation for each set. If the coefficient of the first set is smaller, the first member of the construction $M_{j1}M_{j2}$ is linked to Y_i by dependency, and otherwise the second member. Now suppose that Z_j can take one of three forms: M_j, $M_{j1}M_{j2}$, or M_{j1} $(M_{j2}M_{j3})$. That is, one partner in the construction being analysed is either a morph, or a construction whose members are morphs, or a construction whose members are a morph and a construction whose members are morphs. The dependency connection is between Y_i and M_j if Z_j has the first form, between Y_i and either M_{jl} or M_{j2} *if* Z_j has the second form, and between Y_i and one of M_{j1} M_{j2} , M_{j3} if Z_j has the third form. There are three sets to be assembled:

	If $Z_j = M_j$	If $Z_j = M_{j1}M_{j2}$	If $Z_j = M_j(M_{j2}M_{j3})$
The first set contains	M_j	M_{j1}	M_{j1}
The second set contains	M_j	M_{j2}	M_{j2}
The third set contains	M_j	M_{j2}	M_{j3}

The omission of all other possible sets is justified by the assumption that in a construction of a given type the two positions are distinct. Again, variation coefficients are calculated and the dependency links are established in the same manner as before. Note that it is not always possible to determine the dependency links within a structure such as $M_{j1}(M_{j2}M_{j3})$ before comparing it with its partners. It may, in fact, be necessary to consider still more complex cases, but the general rules are implicit in the example just given. If projectivity is postulated as a universal feature of natural languages, it simplifies the search for dependency links, but its use would require a long discussion that would be out of place here.

The procedures just given lead to the establishment of dependency links, but they do not indicate the direction of dependency; they do not differentiate governors and dependents. When a long span of text (e.g. a sentence), is connected by dependency links, it only remains to choose one occurrence in the span as origin and all links are automatically directed toward that occurrence. For this purpose, it is important to introduce the restriction of projectivity. In a projective language, every origin occurrence lies on the unique path between the first and last occurrences in a connected span. That is to say, the first occurrence in a sentence is connected to one or more following occurrences, among which one or more are connected to following occurrences, and so on, until some sequence of connections leads to the last occurrence in the sentence. This sequence of connections forms a path through certain occurrences, and one of them must be independent; projectivity would be violated if any occurrence not on the path were chosen as the independent, or origin, occurrence in the sentence. Moreover, every occurrence not on the path depends, directly or indirectly, on some occurrence on the path; hence all connections outside the path are directed, their governors and dependents differentiated, as soon as the universal of projectivity is adopted.

Let us assume that every morph has been assigned to some class, and consider all pairs of occurrences such that the first belongs to some class, say X, the second to some class Y (the two classes may be the same or different), and the two occurrences are connected. If the direction of dependency is sometimes from X to Y, sometimes from Y to X, the description of the language is more complex than if the dependency always goes in one direction. A partial test of the projectivity postulate is to determine, for each such pair, whether the directions induced by projectivity are consistent or variable. Variability for many pairs would make the postulate doubtful for the language. If, on the other hand, consistency is found, the determination of an origin occurrence for each sentence can also be based on a consistency argument. Define a dependency type by the classes of two connected morphs and their order, and assign to each dependency type found in the text a direction according to the findings just described. For some types, i.e. those that occur only on the initial-final paths of text sentences, direction will be undetermined. In text, mark each connection on an initial-final path with the direction pertaining to its type. In each connected span, there are three possibilities, (i) Every connection is marked, and a unique origin occurrence is determined. Every connection is directed toward it. (ii) Not all connections are marked, but the marks are consistent; those near the beginning of the span point toward the end, those near the end point toward the beginning. If any connections are unmarked before the last right-pointing connection or after the first left-pointing connection, they can be marked and their types assigned the indicated direction. The origin occurrence lies in an unmarked zone and remains to be chosen (see below), (iii) The marked connections are inconsistent; they do not point toward a single occurrence or cluster of occurrences. In such spans, one or more occurrences must be located according to a minimization of inconsistency. If the result is unique, the origin is determined; otherwise, the origin of the span remains to be chosen by the procedure below.

The procedure above can be iterated, since it sometimes determines the direction of a new dependency type. When it stabilizes, there may remain spans (sentences) with indeterminate origins. In fact, there may be many such spans, and the number of possible origin occurrences in each may be large. Note that no dependency type within the indeterminacy region of any span occurs anywhere else. Hence all such dependency types could be given the same direction without inconsistency. If that plan is not satisfactory, all dependency types with undetermined direction can be partially ordered by sequence of occurrence and those up to any arbitrary point in the partial order made right-pointing, those beyond it left-pointing.

Another criterion can be introduced here, or even earlier if it is regarded as linguistically more important than the kind of consistency used up to this point. Some classes must occur at sentence origins; it makes linguistic sense to minimize the number of different classes there. If some origins are determined by projectivity and consistency, they determine certain origin classes, and members of those classes can be sought in each sentence with indeterminate origin. If there is one, in a sentence, its origin is determined. If there are two, the choice can wait on consistency (every time an origin is chosen, new dependency types are given directions). The sentences containing no possible origin of a known origin class are collected and choices made simultaneously for all of them in such a way as to minimize the number of new origin classes; this calculation is feasible if the number of choices to be made is not too large.

Thus if Lamb’s procedure, or Garvin’s, can give a phrase-structure to each span of text, it is possible to extend the analysis to a dependency structure. It remains to be seen whether the procedures of Lamb and Garvin will be satisfactory; almost without doubt, they will need elaboration. Lamb’s has been applied, in tentative fashion, to a small amount of English text with gratifying results; such a trial, as Lamb remarked in reporting it, is far from a demonstration of workability. The dependency procedure added here has not been tried at all.

According to the viewpoint developed earlier, the determination of morphemic structure, relations among morphs, is not the end of linguistic research. First the sound or letter sequences were segmented into morphs, then the morph sequences analysed for syntactic relations so that a dependency diagram could be given for each sentence. (In cases of ambiguity, there are alternative diagrams, of course.) The identification of morphs with similar spelling and identical or closely related distributions as alternative representations of the same morpheme remains to be done, but that is a side problem that can at best reduce the difficulty of the following main step, which is analogous to the segmentation of the original letter sequence: the dependency diagrams have to be segmented into semes (Lamb’s sense of the term [26], approximately). These semes are more nearly the units wanted in translation than the morphs or morphemes that comprise them; they have syntactic relations of their own; and a sentence must satisfy simultaneously conditions best stated in terms of (a) letter or sound sequences, (b) dependency diagrams over morphs or morphemes, and (c) diagrams over semes or sememes. The research procedures that can now be envisaged are very like those already discussed in this section, and this formulation of the problem of ‘deep grammar’ as Hockett [27] calls it or semantic compatibility in the traditional terms is so recent that little can be said beyond a plea for attention to it.

CONCLUDING REMARKS

The procedures described in Section 3 are not ‘discovery’ procedures; they are merely aids to the linguist who uses them along with all his knowledge of linguistic theory, semantics, and the rest. He is aided, if he is fortunate, by insight or intuition, or perhaps by fortunate guesses. His result may be a grammar in the formal sense, or merely a collection of observations. The procedures of Section 4 are ‘discovery’ procedures in the linguistic sense, but they are not infallible. They can be applied, to the extent that they have been specified, without any use of semantics, intuition, or judgment. Their application, however, will not always lead to a complete, consistent grammar capable of assigning at least one description to every sentence in the text on which it is based and to some other sentences not in that text.

On the contrary, such discovery procedures can be written without difficulty. For example, given a text, cut it at random into ‘morph occurrences’, insert ‘sentence boundaries’, and assign every morph to one or both of two classes: class *X* does not occur just before a sentence boundary, class *Y* always does. Adopt two dependency rules: an occurrence of a class *X* morph governs a following occurrence of a class *X* morph or of a class *Y* morph. An occurrence of a class *Y* morph therefore governs nothing. This grammar covers the text and can generate an endless number of additional sentences. It will account for new texts chosen at random, except for the necessity of adding some new 'morphs' to the dictionary. It is unambiguous, in that it assigns exactly one structure to every sentence. Unfortunately, this grammar will accept a great many intuitively undesirable sentences, help but little in machine translation or information retrieval, and recognize too few morphs in new text. This morphemic grammar, moreover, will show no relation to any higher or lower-stratum grammar. The two classes, *X* and *Y*, are not morphologically differentiated, even approximately, and the discovery of semes would be fortuitous. Thus its internal simplicity is matched by the enormous complexity of its external relations.

Bar-Hillel, during the Advanced Study Institute at which these lectures were given, stated several theorems that have not yet been published. Their general tone, when applied to problems of empirical linguistics, is to denigrate ‘discovery’ procedures. Given an infinite set of sentences, it is impossible to determine their grammar, even if it is known in advance that they have a context-free phrase-structure grammar; the theorems quoted are even stronger and broader, but their essential feature is the impossibility of absolute inference from a finite analysis to the infinite set of sentences. Given a finite text, as we have seen, finite grammars are easy to obtain. The issue is extrapolation.

One could suspect, even before the enunciation of these theorems, that there would be difficulties. Supposing the existence of an infinite set of sentences for theoretical purposes and deciding whether a given sequence belongs to the infinite class of 'English sentences' for empirical purposes are two distinctly different problems. The only ways to decide, empirically, about a given sequence are to find it in text and to ask an informant. Text usually gives no answer; the number of possible sequences over a given alphabet or vocabulary is much greater than the number of sentences even in an immense text, and the linguist wants to extrapolate, not to describe the given finite text. Asking an informant gives an uncertain answer, one that varies from informant to informant and even from time to time with a single informant; the answer depends on the kind of question asked as well as on the sentence given, and there is not unanimity about the question. The answer to these difficulties has always been to impose more and more criteria on the grammar derived from a finite text, to check it against new text, to check it, overall, not in minute detail, against intuition, and to include criteria of *interstratal* consistency: Syntax must accord with morphology and semantics or sememics.

The new theorems confirm this approach by denying the possibility of any other. The empirically difficult concept of an original infinite set of sentences for which a grammar must be found is now seen to be theoretically worthless, since the correspondence of grammar and 'language' (infinite set of sentences) would be unverifiable. Intuition and insight could yield a perfect grammar, but its perfection would be untestable. Systematic procedures may never yield a perfect grammar, but their connection with finite text samples, via criteria of analysis, can be explicated, as the connection of an intuitively derived grammar cannot be. The basic concepts of linguistics, replacing the empirically and theoretically difficult concept of an *a priori* infinite set of sentences, will therefore be the finite collection of textually validated sentences and the set of sentences generated by a grammar. (There are theoretical difficulties about the latter set, but they do not influence this discussion.) The connection between these two sets is made in two steps: criteria for derivation of a grammar from a finite text, and procedures for the generation of a set of sentences under the control of a grammar. The grammar is rigidly connected with the finite sample. Its connections with the rest of the 'natural language' for which it is proposed as a summary description necessarily remain vague, but the linguist can test its adequacy for the recognition of sentences in new text by mechanical procedures, and he can test, by recourse to informants, the acceptability of its analyses of given sentences and the acceptability of sentences that it generates. The grammar has become the instrument of extrapolation, as Chomsky once hinted [12], and the criteria of its derivation determine the extrapolation made.

REFERENCES

[1] K. E. HARPER, D. G. HAYS and B. J. SCOTT : Studies in Machine Translation—8: Manual for Postediting Russian Text, in *Natural Language and the Computer* (Ed. Paul L. Garvin), pp. 183-213. McGraw-Hill, New York, (1963).

[2] D. G. HAYS: *Research Procedures in Machine Translation.* RM-2916, The Rand Corporation, Santa Monica, California (December 1961).

[3] H. POUTSMA: *A Grammar of Late Modern English.* Noordhoff, Groningen (1928) (two parts in five volumes).

[4] C. A. FERGUSON: *Language* 1962, 38, No. 3, 284.

[5] Z. S. HARRIS: *Methods in Structural Linguistics.* Chicago University Press (1951).

[6] E. A. NIDA: *Morphology.* Michigan University Press (1949).

[7] K. TOGEBY: *Structure Immanente de la Langue Française* (Vol. VI of Travaux du Cercle Linguistique de Copenhague), Nordisk Sprog- og Kulturforlag, Copenhagen (1951).

[8] A. MARTINET: *Economie des Changements Phonétiques.* Francke, Berne (1955).

[9] G. A. MILLER: *IRE Transactions on Information Theory,* Vol. IT-8, No. 2, pp. 81-83 (February 1962).

[10] G. K. ZIPF: *Human Behavior and the Principle of Least Effort.* Addison-Wesley. Cambridge, Massachusetts (1949).

[11] H. P. EDMUNDSON and D. G. HAYS: *Mechanical Translation* 1958, 5, No. 1, 8.

[12] N. CHOMSKY: *Syntactic Structures.* Mouton, The Hague (1957).

[13] H. GAIFMAN: *Dependency Systems and Phrase Structure Systems.* P-2315, The Rand Corporation, Santa Monica, California (May 1961).

[14] K. E. HARPER and D. G. HAYS: *Information Processing,* pp. 188-194. Unesco, Paris (1960).

[15] D. G. HAYS and T. W. ZIEHE : *Studies in Machine Translation*—10*: Russian Sentence-structure Determination.* RM-2538, The Rand Corporation, Santa Monica, California (April 1960).

[16] Y. LECERF: *La Traduction Automatique* 1960,1, No. 4, 11; 1960, 1, No. 5, 17.

[17] L. TESNIERE: *Eléments de Syntaxe Structurale,* Klincksieck, Paris (1959).

[18] Z. S. HARRIS: *Language* 1955, 31, No. 2, 190.

[19] P. GUIRAUD: *Bibliographie Critique de la Statistique Linguistique.* Spectrum, Utrecht (1954).

[20] D. H. HYMES: *The Use of Computers in Anthropology* (to be published).

[21] H. BORKO: *The Construction of an Empirically Based Mathematically Derived Classification System.* SP-585, System Development Corporation, Santa Monica, California (October 1961).

[22] Z. S. HARRIS: *Language* 1952, 28, No. 1, 1.

[23] E. V. PADUCHEVA : *Exact Methods in Linguistic Research,* by O. S. Akhmanova *et al.,* translated by D. G. Hays and D. V. Mohr. California University Press, Berkeley (1963).

[24] P. L. GARVIN: *Proceedings of the* 1961 *International Conference on Machine Translation of Languages and Applied Language Analysis,* pp. 655-671. H.M. Stationery Office, London (1963).

[25] S. M. LAMB: *Proceedings of the* 1961 *International Conference on Machine Translation of Languages and Applied Language Analysis,* pp. 673-686. H.M. Stationery Office, London (1963).

[26] S. M. LAMB: The strata of linguistic structure. Presented at a meeting of the Linguistic Society of America, Hartford, Connecticut (December 1960).

27] C. F. HOCKETT: *A Course in Modern Linguistics.* Macmillan, New York (1958).