JOURNAL OF VERBAL LEARNING AND VERBAL BEHAVIOR **4**. 196-210 (1965)

The Nature of the Machine Translation Problem[1]

SYDNEY M. LAMB

Universitv of California, Berkeley, California

Machine translation can be viewed as a type of human translation, since the translating machine will merely follow rules provided by the human linguists now engaged in machine translation research: but it is more difficult than ordinary human translation, and the solution of the problem requires a careful analysis of the translation process and its relation to linguistic structure. The inadequacy of ineffective procedures can be shown by their lack of means of handling various phenomena known to exist in languages. Methods that can be discarded in this way include those of word-for-word substitution and word-for-word substitution plus doctoring, as well as other methods which use words as basic units. More advanced systems, which show promise of success on theoretical grounds, are those which recognize the various independently functioning grammatical units and structural strata of language. For such systems translation consists of a series of interstratal conversions, from morphemic to lexemic to sememic to semantic and from there through the strata of the target language, ending with strings of target-language graphemes.

Professor Y. R. Chao, in a paper presented at the 9th International Congress of Linguists entitled "Translation Without Machine," remarked. "If the human organism is viewed as a machine, then all translation is machine translation, though obviously that will be so only in a trivial sense." Now instead of making this observation I would like to make the opposite point: that all translation can be viewed as human translation since machine translation is nothing but another kind of human translation.

In order to make clear why this is so it is necessary to bring to mind certain important properties of the digital computer. In the first place, machine translation does not require a special machine. Instead, it is possible to use a general purpose computer, such as the IBM 7090. The function of such a machine is essentially one of following instructions. A complete list of instructions provided to the machine for the execution of a sequence of operations is called a *program.* The machine will do exactly and only what the program specifies, and aside from the simple individual operations that it performs, such as adding or moving information from one place to another, it does nothing on its own. Consequently, it must be told by the program exactly how to do everything that is desired of it. Nothing can be left for the machine to take for granted and every possible set of circumstances which might come up during the execution of a program must be provided for.

Thus when we finally get a machine to translate from Russian or Chinese to English, it will not be so much the machine as the program which will be *doing* the translating.

[1] The original version of this paper was presented in December. 1962. with the title "Machine Translation and Human Translation" as one of a series of Lectures on Linguistics given at the University of Toronto under the auspices of the President's Committee on Linguistics. Other versions have subsequently been presented as lectures to various audiences, including the Colloquium of the Institute for Human Learning at the University of California. Berkeley. The research which provided the background for this paper is supported by the National Science Foundation.

The situation is analogous to the construction of a building. The actual work of constructing the building is done by the workmen in their overalls, but what they do is governed entirely by the blueprints which have been supplied by the architect. The workmen correspond to the computer, the blueprints to the program and the architect to the programmer.

The essential problem in machine translation research, then, is to determine what the process should be and what kind of linguistic information is needed for this process, because the machine must be told in complete detail, in advance, every step that it is to take.

And this is why machine translation can be considered just another form of human translation. The humans involved, who in effect will be doing the translation, are people now doing the necessary linguistic analysis and formulating instructions for the machine. That is, people engaged in machine translation research are working now to provide the instructions which are going to be translating articles that have not even been written yet. This is something like the speculator on the grain exchange, who sells wheat that has not been grown yet. But it is actually a little different in that when one is doing *futures translation,* it is an enormously more complicated job than *actual translation* done in the present time, because with actual translation one has the text before one and the job is to provide a translation for that specific text. But machine translation researchers do not know yet what the Russian scientist is going to write and so it is necessary to provide for translation of anything that he might write. And of course this makes the problem much more difficult.

It is necessary, then, to do a very complete analysis of the translation problem, even subjecting to close scrutiny some things which might seem very trivial to the ordinary human translator, things that he takes for granted and does without even thinking about it. Let us therefore examine in detail what the translation process must consist of in order to be successful. This examination, as it happens, will tell us a good deal about the structure of language, so it will be interesting from another point of view also. In analyzing the translation procedure let us begin with the simplest possible procedure and move on step by step to more and more powerful ones.

In determining the inadequacies of any less powerful system than one which is adequate, there are two ways in which one can operate. One way would be to program a computer to use one of the elementary procedures and then give it some text to translate: and when the result turns out to be linguistic garbage it will be apparent that something is wrong with the procedure. Now this can be a very slow process because it takes a good deal of lime to program a computer. The other way is to examine the underlying theory of the structure of language upon which the procedure is based. If one does that and can demonstrate that the theory is inadequate to account for things which we observe in language, then one can predict in advance that this procedure must necessarily fail, and one avoids going through the process of programming it in order to see the inevitable failure. Now strangely enough, several of the machine translation projects have actually operated according to the first policy. They put together a primitive rough-and-ready trial-translation procedure, which could have been examined in advance to demonstrate its inadequacy, and they nevertheless program it for the machine and try to translate with it. Then when they examine the results and see that they are garbage, they must go back to the drawing boards and write a more elaborate program.

Let us save time, however, by following the second approach, looking at various procedures in the light of what we know about the structure of language. Knowledge of the structure of language will indicate what a translation procedure must have in it in order to be adequate.

The first system is diagrammed in Fig. 1.

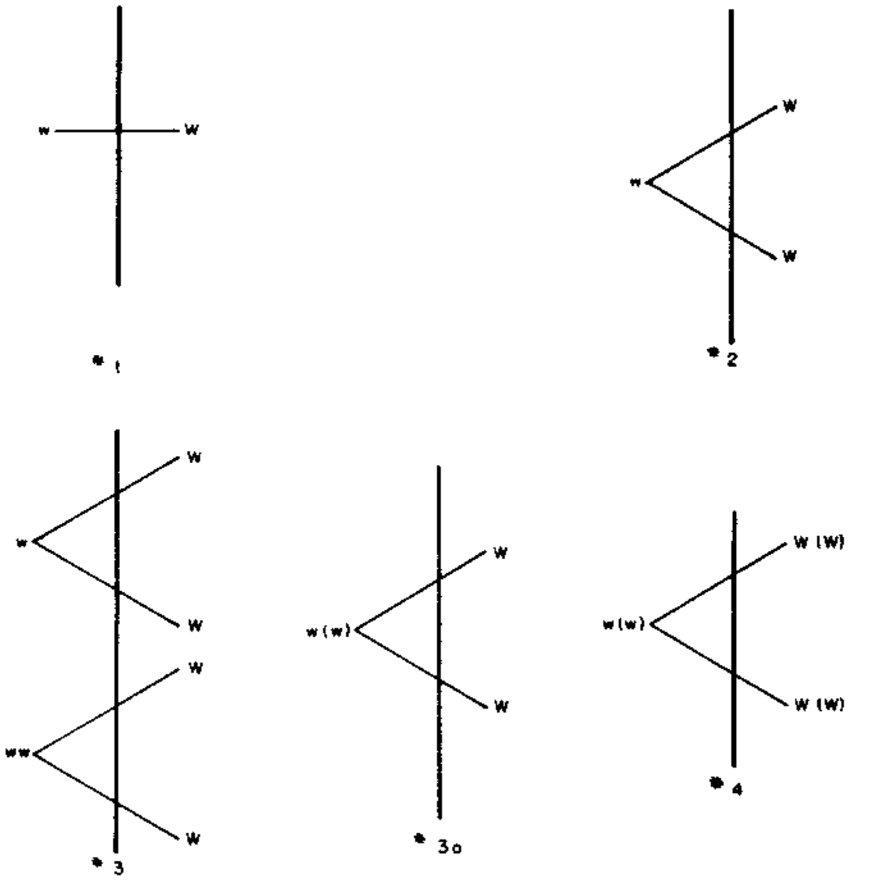

FIG. 1. Word-for-word substitution.

What is depicted here can be called *word-for-word substitution.* Some people call this "word-for-word translation," but it should not be dignified by the term "translation." The vertical line in the diagram separates the *source language* from the *target language.* For example, we could have on the left Russian, on the right English. The "w" stands for *word,* and it is necessary to show only one word in the diagram because in word-for-word substitution, every word is treated the same way. So this is a picture of what is happening to one word. In these diagrams small letters are used for items of the source language and capital letters for items of the target language. The first diagram simply depicts that we take for every word of the source language a substitute in the target language.

This is clearly much too primitive a procedure to be able to work; so let us immediately move on to #2 (Fig. 1), which can also be referred to as a type of word-for-word substitution. But in this case, instead of being one-to-one word-for-word substitution, it is *one-to-many word-for-word substitution,* since we recognize that a given word of the source language might have one translation in one context but another in other contexts. (Throughout these diagrams, wherever two items are shown, it is intended to represent any number, i.e.. not just two. but two or three or more, or, in special cases, only one, depending on the individual item.) In other words, we recognize that a word of the source language has multiple equivalents in the target language, two or moe, only one of which is appropriate in any given circumstance. Now of course it is one of the most difficult problems in machine translation research to determine which is the best equivalent to use in any particular case, but this will not be considered further here. The important point is that there would be rules of some kind supplied to the machine, so that it would make a choice in any specific case, rightly or wrongly, by examining the context. For a system of the type of #2. such rules would have to be very complicated indeed to be effective, but some of the more advanced systems provide a simpler way of handling this problem.

This procedure has numerous inadequacies, including failure to recognize idioms, failure to deal with problems of arrangement, etc., which can be remedied by moving on to more sophisticated systems. In #3 the inadequacy taken care of is the failure of #2 to deal with idioms. In addition to what is in #2 we have "ww" on the left, by which is meant a combination of two or more words which has to be treated as a single unit in translation. So not only will there be situations in which a word is substituted for by one of a set of others, there will also be these instances in which a combination of words taken as a unit—in other words an idiom—must be given a translation.

Number 3a is called #3a rather than #4 because it is not a basically different system from #3. Instead of giving the two possibilities with both "w" and "ww" separately, it shows simply "w(w)" to stand for one or more words. In other words,

w(w) == w or ww.

This is, then, a slightly more elaborate form of word-for-word substitution than #2, in that the things being substituted for can be either single words or combinations of words.

The fourth system is just a slight elaboration on #3. The only difference is that it recognizes that the substitution in the target language can also be a combination of words instead of just a single word. This is a slightly more general scheme in which for the source language the unit to be substituted for may be either a word or a combination of words, and the substitution in the target language may likewise be either a word or a combination of words; and it would be hoped that the machine could be programmed to make a choice among alternative substitutions for each unit.

Here we have come, in terms of the results that would be achieved, a long way from #1, but the system is still very primitive. Nevertheless this actually does represent, approximately, a type of system that has been put into operation by one machine translation group—with disastrous results, of course (although they claimed that these results were very useful indeed). A very curious phenomenon occurs when one looks at the output of this system. In most people's previous experience, during their entire lives, they have never looked at a page with English words on it except under the circumstance that these words go together and form meaningful sentences. One is conditioned by this habit over a period of many years so that when one looks at this output, one sees English words and supposes that they form English sentences. But some people look a little more closely and try to put the words together into sentences, and they find that the message is not quite there after all. In this particular system, there was a limited attempt to make a choice between different English equivalents, but only very limited and of course not always successful. Most of the Russian words were provided with only a single English equivalent which was used in all cases.

But most machine translation workers long ago went on to more elaborate systems. One type of elaboration is #5 (Fig. 2). This can be called *word-for-word substitution with deletions and insertions.* The diagram shows at the top. for example, that after the initial conversion into the target language there are two courses of action: either that item—word, or combination of words—can remain or (notice the other branch going down to nothing) it undergoes a process of deletion. And, correspondingly, down at the bottom of this diagram there is a word or combination of words at the right which comes from nothing: this represents an insertion. For example, in translation from Russian to English according to this scheme it would be necessary to insert prepositions at the beginning of noun phrases in many instances, where prepositions are not present in Russian but are necessary in English. Deletions would be necessary for these situations if we are translating from English to Russian, and would generally be necessary regardless of the target language for instances of *empty words*. For example, the word *do* as in *I do not know him* will occasion a deletion because this *do* really does not carry any meaning in English. It is required because of a grammatical rule that English has. (It is not grammatical to say *I not know him* although it is perfectly meaningful.) So in translating from English it would be necessary to delete whatever had first been put in as the equivalent of "do." At least this is one way to treat such problems, although there are more sophisticated ways, described below.

This scheme of course is still inadequate, and the chief inadequacy of all the systems considered so far is their total failure to cope with differences in the ordering of words from one language to another. No consideration has been taken at all of word order and consequently any kind of translation will look very awkward. In #6, however, we have *word-for-word substitution with deletions, insertions, and rearrangements.* In the diagram the part added to what was present in #5 is an attempt to depict the rearranging of each item relative to other items in the sentence. This procedure first produces substitutions in the same order as in the source language, and then rearranges certain items as necessary in order to provide a more grammatical arrangement in the target language. Now #6 is a scheme that has been used by more than one of the American machine translation projects. One group has in fact achieved amazingly good results considering how inadequate the system actually is.

But #6 is still taking only a limited cognizance of the importance of features of arrangement in language. In particular, it implies a denial that features of arrangement can be used to express meaning, since the rearrangement is applied only *after* the target language equivalents have been chosen, for providing a grammatically acceptable order in the target language.

Before that deficiency is considered, however, the procedure as we have it so far can be simplified. The seventh system is like #6 but somewhat simpler. Here, instead of first choosing target language equivalents and then deleting some of them, we simply say that some of the source language items will be translated by zero, that is, by nothing at all. So some will have words or combinations of words as their equivalents: others will simply not be translated by anything—that is what that line going down to nothing represents—and instead of insertions there are certain places where there is zero in the source language and where a target language word or combination of words will appear.

Number 7a is the same scheme as #7 but with a simplified diagram, in which there is another pair of parentheses around the word or combination of words in each case. This shows, then, that we can have either a word or combination of words or nothing represented in the target language by a word or a combination of words or nothing. That is, we can have a word represented by nothing, nothing represented by a word, and so forth —all the various possibilities—and then rearrangement.

Now we are ready for a more sophisticated way of dealing with the arrangement problem, which will recognize that the way in which words are arranged in a language can be a carrier of meaning. For example, *the dog bit the man* means something entirely different from *the man bit the dog.* The words are exactly the same: the only difference is in

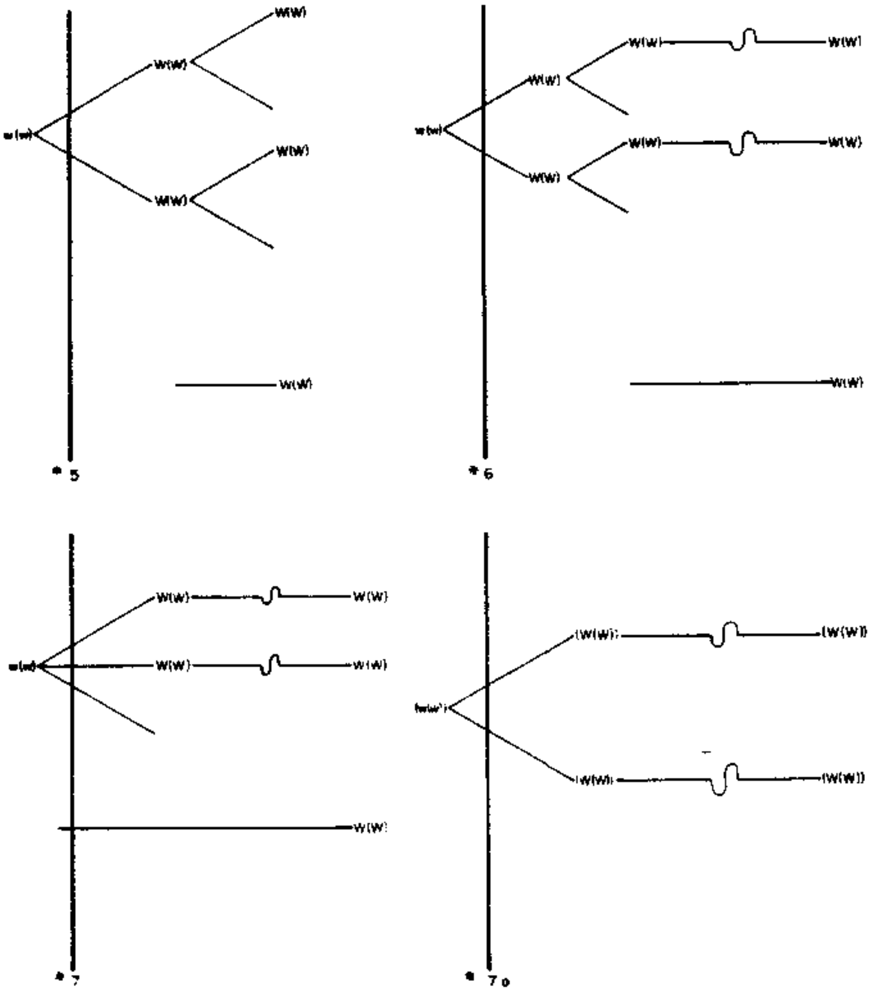

FIG. 2. Word-for-word substitution plus doctoring.

the order. It is necessary, then, for a good translation system to do some analysis of what the syntactic relationships are, and some kind of transfer of them to the target language. The diagram for #8 (Fig. 3) has a simple tree in the source language, which is converted into a target language tree; these simple trees in the diagram are intended to represent constituent-structure trees, since constituent structure is a commonly used means of indicating syntactic relationships. and is easily implemented on the computer. In the tree are given distribution-class symbols at each node, each one indicating the combining characteristics or *distribution* of the form (word or combination of words) to which it corresponds. System #8. then, has word-for-word substitution together with determination of syntactic relationships present in the source-language sentence and substitution of an equivalent syntactic structure in the target language (instead of rearrangement as in systems 6 and 7).

We must immediately move on to #9, which recognizes, as #8 does not. that a given feature of arrangement in the source language, like a word, may not always be translated by the same feature of arrangement in the target language but may require different target arrangements in different situations.

Number 10 differs from #9 in efficiency rather than in adequacy. Here we recognize that it is not just the case that a given source language item has multiple equivalents in the target language; it is also the case that an item of the target language may be the representation of more than one item of the source language. If one takes #9 seriously in terms of a theory of language, then this theory would imply that every target language has a larger vocabulary than every source language, a proposition which is manifestly false, since the set of potential source languages coincides with that of potential target languages. Consider, for example, a Russian-to-English translation system based on #9. Since many of the Russian lexical items would have multiple equivalents, there would be more English lexical items than Russian in the dictionary, and the implication would be that English has a larger vocabulary than Russian. Now consider an English-to-Russian system,

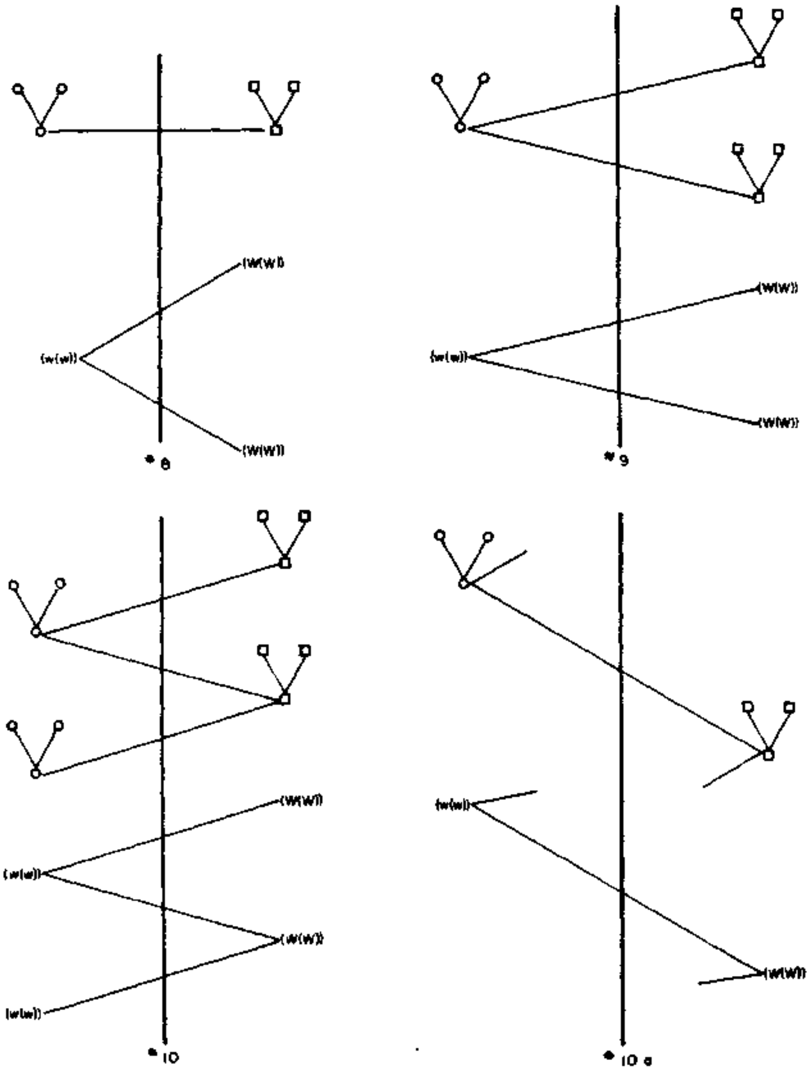

FIG. 3. Systems with syntactic decoding and transfer, based on words.

and by the same argument it would appear that Russian has a larger vocabulary than English. This little paradox is solved by recognizing that in. e.g., the Russian-to-English dictionary, the incidence of multiple English equivalents in dictionary entries will be matched by a roughly equal amount of occurrence of English vocabulary items in multiple dictionary entries. This situation is taken care of in #10, and the figure also indicates that the corresponding situation can be expected in the case of features of arrangement. The way to implement #10 on a machine is to separate the bilingual dictionary into two parts. That is. there should be a Russian dictionary and an English dictionary instead of a Russian-English dictionary. In the Russian dictionary would be supplied reference numbers to the English equivalents to enable them to be located in the English dictionary. Thus each English equivalent in the separated dictionary occurs only once, so that the total dictionary volume is considerably smaller than for the unseparated one of #9. This smaller size, plus the fact that the machine's active storage area is not cluttered up with English equivalents during its examination of the Russian material, enables the translation to proceed much more rapidly than would be possible for a system of type #9.

Number 10a is a simplification of the diagram for approach #10. Here, instead of showing overtly all of the multiple correspondences between source and target languages, it shows only one. Thus one may read the diagram as indicating that in any specific case just one of the alternatives is chosen and that the one which is chosen might also be chosen as the equivalent of one or more different items of the source language at other points in the text. In other words, all of the same multiple possibilities depicted for #10 still exist as possibilities but each specific instance involves only one of them.

Now we may move on to #11. Up to now each system has had *words* as the elements to be translated, and that is really rather crude because words as such are not directly the bearers of meaning. If we consider an inflected word, say a genitive case form of Russian or Latin, it really has at least two meaningful elements—the stem and the case ending—and to be efficient a translation system must deal with these two elements independently of each other. Suppose that the system must accommodate ten thousand nouns each of which can occur with ten different case suffixes. Then a system with words as its basic units would require for these noun forms 100,000 dictionary entries (containing much duplicated information) while if words are separated into their independently functioning parts only 10,000 entries are needed for the noun stems plus a few for the case suffixes. System #11 (Fig. 4) therefore recognizes that the meaningful elements to be translated can be not only words and combinations of words but also *parts* of words. A procedure of this type must therefore begin by segmenting words into lexical elements (indicated by "1" in the diagram). For such a system, the input text is in the form of graphemes (i.e. elements of the writing system, such as letters) and the first stage of the procedure segments this string of graphemes into those substrings which have dictionary entries. A method for this segmentation has been described by Lamb and Jacobsen (1961) and has been implemented on the 7090. The diagram shows "gg" representing any such substring, from a single grapheme to a combination of two or three or more, comprising part or all of a word or more than a word. It is such lexical elements which will be given substitutions in the target language in #11, and it should be taken as understood that lexical items may also be recognized, in either language, which consist of no graphemes at all, e.g., the nominative singular element for many Russian nouns. But the device of inserting prepositions in translating from Russian to English, mentioned above for #5, is not such a situation here because with segmentation of words we can recognize that such English prepositions are

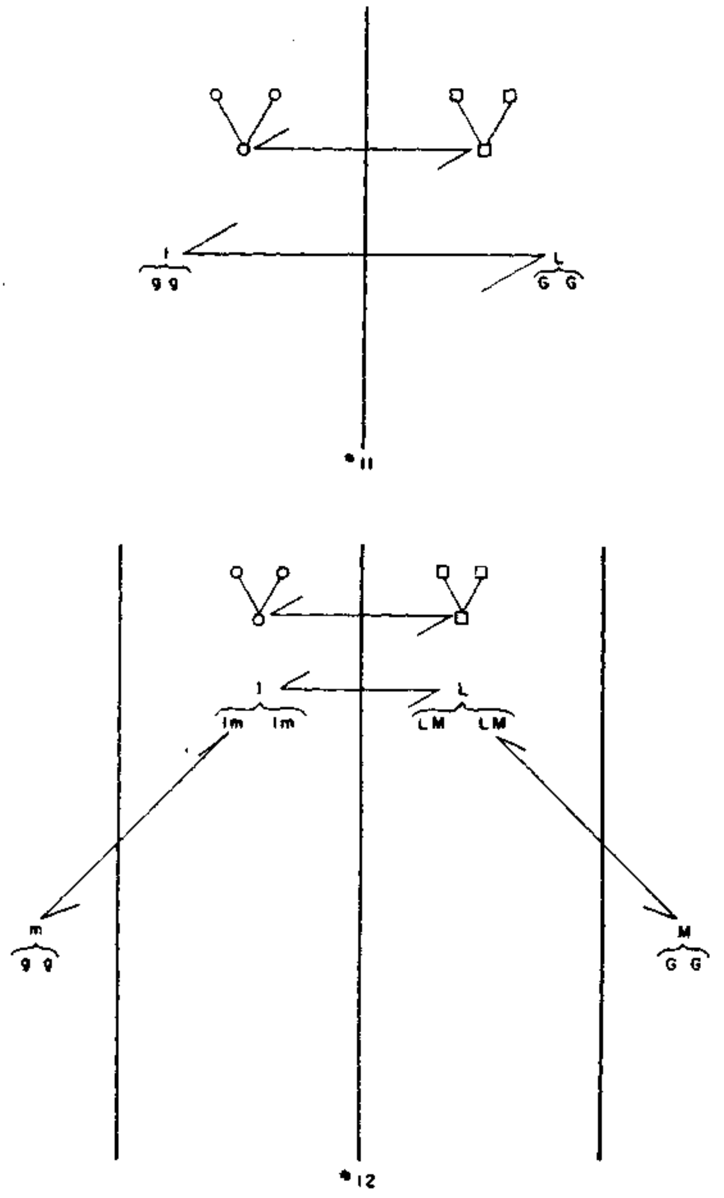

FIG. 1. Systems with segmentation of words before syntactic decoding.

really equivalents of Russian case suffixes rather than elements which correspond to nothing in Russian.

Some machine translation projects have advocated segmentation of words only for purposes of making the dictionary look-up process efficient, after which words are reconstituted and dealt with as units for determination of syntactic relationships and conversion to the target language. Such groups, unable to break away from the notion that words as such are bearers of meaning and basic units of syntax, are really using what is basically a variety of approach #10. rather than #11. Several investigators, however, have seen the desirability of moving on to #11, while a very small minority has recognized the need for the further advance of #12.

All of the systems up to now have dealt with translation as crossing directly over (even if after syntactic decoding) from the words or parts of words or combinations of words of the source language into equivalent items of the target language, ignoring alternating equivalent elements within a single language. In English there is a plural element which can appear in various different shapes. It can be realized as just an *s* as in *tigers,* as *es* as in *boxes,* or as *en* as in *oxen* or *children.* It is these individual shapes which the machine is first confronted with: and before it can translate efficiently it must convert from these into the underlying structural elements of which they are merely the graphemic realizations. So for an item like *en* it should, in effect, look at the stem that this occurs with, and on seeing that it is *ox* it should decide that this *en* represents the plural element, while if the stem is *eat* it is quite a different element and if the stem is *bright* it is still another.

These underlying lexical elements, such as *pl., past participle, child* (which is realized in the shapes *child* and *childr),* etc.. may be called *lexemes.* It is the lexemes rather than the combinations of graphemes that should be analyzed syntactically and translated into lexemes of the target language, which in turn should be converted into their proper graphemic realizations at the end of the translation process. Thus in the case of translating into English the conversion will first be to *pl.* for a given lexeme of the source language, after which the proper graphemic shape (which has nothing to do with the source language) may be determined by reference to the morphological code of the (target language) stem with which it occurs.

A system like that just described is of type #11½, part of the way from #11 to #12, but only part of the way, even though it represents a considerable advance over #11. To get all the way to #12 it is necessary to recognize an intermediate grammatical unit between the grapheme and lexeme, namely the *morpheme.* The alternations described above (for *pl.,* etc.) are of the type which in many cases are concerned with only *parts* of lexemes. For example, the past participle element of English, which is realized graphemically in the forms *ed, d, en, n,* etc., is lexemic in some of its occurrences while in others it is only a part of a lexeme. For example *red-headed woodpecker* is a lexeme, since its meaning is not inferrable from its constituents (as opposed to, say, *yellow-headed canary,* which is polylexemic), but it is made up of clearly recognizable grammatical elements *red, head, ed* (past participle), *wood, peck, er,* each of which in turn is realized as a string of graphemes. Here, as in *covered wagon,* the past participle element is only a part of a lexeme (note that *uncovered covered wagon* is not absurd). Such a grammatical element may be called a *lexomorpheme* ("lm" and "LM" in Fig. 4) and its realizations may be called *morphemes* ("m" and "M"). Thus *ed, d, en, n,* etc. are morphemes which (although *ed, d* can also represent *past tense, en pl,* etc.) are realizations of the *past participle* lexomorpheme. which, besides occurring by itself as a lexeme, as in *covered table,* occurs as a constituent of various other lexemes such as *covered wagon, red-headed woodpecker.* and the *passive* and *perfect* lexemes.

But now notice one further property which lexemes can have, illustrated by the *passive* and *perfect* lexemes which occur with English verbs. They are discontinuous. The *passive* lexeme consists of the lexomorphemes *be* and *past participle* (each of which is realized in various ways as determined by the environment), but these constituent lexomorphemes are not adjacent, e.g. *be eaten, was kept, is covered.* Similarly the *perfect* lexeme consists, discontinuously, of the lexomorphemes *have* and *past participle,* a? in *have eaten, has covered,* and (together with passive) *has been kept.* These examples and many others which could be added make it clear that it is at best highly inefficient to try to segment the grapheme string of the input directly into realizations of lexemes.

Instead, the move to scheme #12 should be made. Here the initial segmentation is into morphemes, and these are then converted to the underlying lexomorphemes of which they are realizations. These first two steps together comprise what may be called *morphological decoding. N*ext there is a second stage of segmentation, to segment the resulting string of lexomorphemes (strictly speaking, alternative strings, since there will generally be morphological ambiguity, resulting in multiple morphological decodings) into lexemes. And this stage of segmentation must be capable of recognizing discontinuous combinations of lexomorphemes (including not just *passive* and certain tense lexemes but also *look up,* as in *look it up, blow up,* etc.)- Then, it is these lexemes which are to be converted to the target language; and, as the diagram shows, there must also be syntactic decoding, with lexemes (rather than lexomorphemes or morphemes or words) as the basic syntactic units. (The diagram shows only one morpheme, as the realization of just one of the lexomorphemes which comprise the lexeme shown, but it is to be understood that every lexomorpheme has a morphemic realization.) Now this system still is not elaborate enough because in dealing with the so-called "multiple-meaning" problem it says only that for any lexeme there are various different alternatives in the target language. The underlying assumption is that the entire multiple-meaning problem is assignable to differences between the two languages. But this clearly is not sophisticated enough because we know that there is a multiple-meaning problem within any individual language. So the multiple-meaning problem as treated in a more realistic translation scheme would be recognized as due in some cases to the source language, in others to the target language. And the way to treat the situation efficiently is to introduce still another stratum, at which we have elements called *sememes. Sememe* is a term designating roughly a unit of meaning (cf. Lamb, 1964), a term introduced by Noreen (1923). The lexeme *big* represents one sememe in *big rock* and quite another one in *big sister,* and the former but not the latter may alternatively be represented by the lexeme *large.* So the machine should convert first from lexemes to the sememes and from them to sememes of the target language. But if one uses this approach, what happens to the features of arrangement?

Notice that in system # 10 the syntactic decoding was done with words as the basic units, while in #11 the basic units of the syntax would be quasi-lexemes and in #12 the basic units are lexemes. Both of these advances, from #10 to #11 and from #11 to #12 have the effect of simplifying the syntax. In other words, the syntactic decoding procedure is simpler in a system which first decodes from the graphemic material to lexemes. Now words and graphemes occur in strings, i.e., linear combinations, so that syntactic trees as applied to words or to strings of graphemes are applied as adjuncts which account for the linear arrangement. But a system which has additional strata to reflect the structure underlying the grapheme strings does not need to have such trees as mere appendages. Instead it is more economical simply to say that the lexemes themselves occur in trees, and that the morphological rules specify not only the morphemic realizations for lexemic material but also the linear order of the morphemes.

Thus it would appear that (1) syntactic decoding should be applied to lexemes to determine their arrangements in trees, and (2) the lexemes should be decoded to the underlying sememic units of which they are realizations. Now it turns out that these two processes go hand-in-hand, since (1) syntactic relations are even simpler and clearer as determined for sememic units than for lexemes, and (2) it is the rules used in syntactic decoding which make it possible to decode from lexemes to their corresponding sememic units. The problem in decoding from lexemes to their underlying sememic units is one of choosing among several possibilities, since it is quite common for a lexeme to be a realization of several different sememes; and the method of making the choice is by determining that (hopefully) only one of the possibilities fits the syntactic rules. For example the lexeme *big* might be a realization of ^{S}/big$_1$/ (as in *big rock)* or of ^{S}/big$_2$/ (as in *big sister),* but in the expression *my sister is too big* the syntactic rules, if properly formulated, will select ^{S}/big$_1$/ as the sememic realizate since they do not allow ^{S}/big$_2$/ to occur in this type of syntactic construction.

Thus the decoding process should be as follows: (1) segmentation of the grapheme string into morphemes; (2) decoding from morphemes to their underlying lexomorphemes : (3) segmentation into lexemes; (4) syntactic decoding and decoding to sememic units. This last step will result not in constituent-structure trees appended to strings of linguistic units but rather in trees consisting of sememic units.

Such a system is #13 (Fig. 5), but the diagram for #13 has one additional feature,

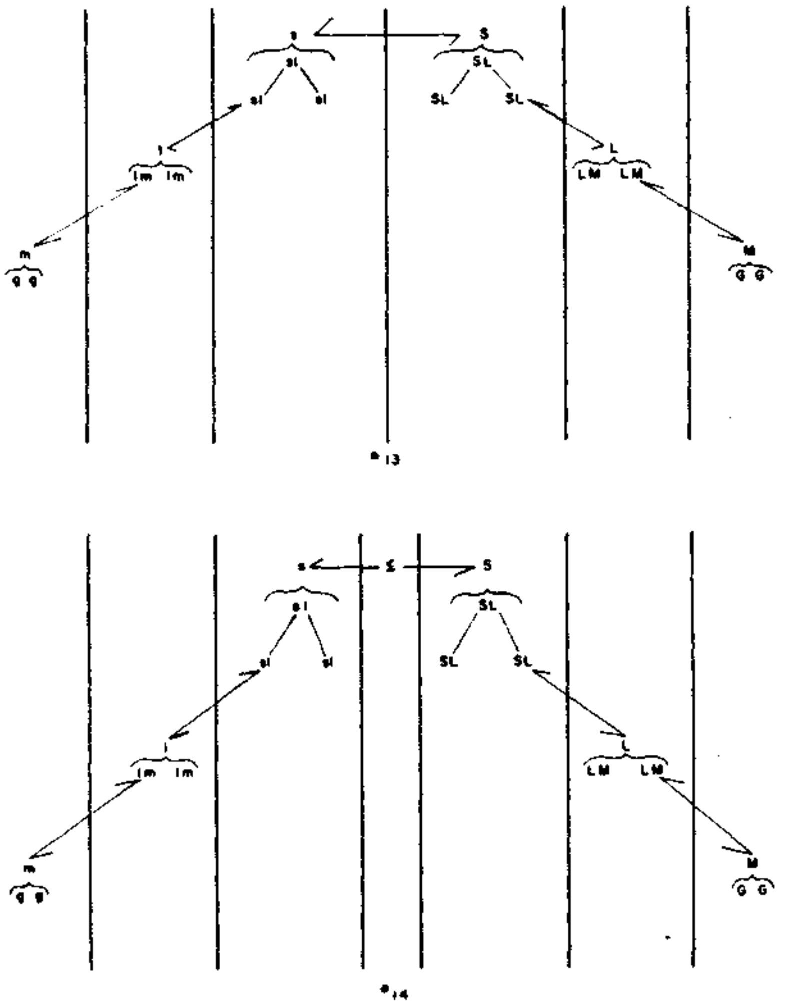

FIG. 5. Systems recognizing the sememic stratum.

namely a distinction between two sizes of units at the sememic stratum, like that at the lexemic and morphemic strata. At the lexemic stratum the larger unit, i.e. the lexeme, is that which relates to the sememic stratum, while the smaller unit, the lexomorpheme, relates to the morphemic stratum. (Of course, many lexemes are composed of single lexomorphemes.) Similarly we should not assume *a priori* that the sememic units which relate to lexemes, i.e., the semolexemes ("sl" and "SL" in the diagram) are the most suitable size for purposes of transfer between languages. In many cases they will be, but in others it may be efficient to recognize sememes composed of more than one semolexeme. For example, in translating from English to Russian it might be efficient to treat *light blue* as a single sememe since the corresponding Russian sememic unit is realized by a single lexeme. Also English tense semolexemes can occur in certain combinations (but not in all combinations) and such combinations should perhaps be treated as units for translation purposes. More generally, proverbs and similar sayings, including expressions whose significance depends on a direct connection to specific cultural features, e.g., *he can't get to* *first base,* can be considered sememes composed of more than one semolexeme.

The linguistic analysis needed to implement a system like #13 for a pair of languages like Russian and English, even in a very imperfect form, will require several more years of work. But such a system, even when provided with relatively complete syntactic and sememic information, will still be unable to resolve certain ambiguities of a type that can be solved with a system like #14 (Fig. 5). Here there is. in addition to what is present in #13, a *semantic stratum* in the middle, so there is one additional stage of decoding. This stage, semantic decoding, is similar to the syntactic decoding in that it uses information about distributional properties to eliminate potential decodings which violate the construction rules. But for semantic decoding it is co-occurrence information determined by semantic (rather than syntactic) properties which is used. As an example, consider the following excerpt from a Russian sentence occurring in a biochemical text analyzed by the Mechano-linguistics Project at the University of California, Berkeley:

. .. ведущее положение занимает сахар . . .
. . . leading position occupies sugar . . .

To the human it is obvious that the correct translation is ". . . sugar occupies a leading position. . ". That is. *sugar* is the agent and *position* the goal of *occupy.* But that fact is not made clear by any grammatical device, since the usual grammatical marking of these relations—use of the nominative lexeme for agent and the *accusative* for goal—is neutralized here in that for both полужние and сахар it happens that the nominative and accusative forms have identical morphemic realizations. (In English the agent and goal relations are generally expressed by ordering features in this type of expression—by putting *sugar* before the verb and *position* after it—but not in Russian.) Thus the stage of morphological decoding will provide two possibilities for both сахар and полужение and the syntactic decoding will end up with two possibilities for the clause, one with *sugar* as agent and *position* as goal, the other *vice versa.* But the rules for co-occurrence possibilities based on semantic properties, if detailed enough, will specify that, in effect, positions are things that can be occupied but that do not occupy things (especially things like sugar).

It might be thought that such knowledge is beyond the scope of computers, beyond their storage capacities, but that is so only if one thinks in terms of a brute-force method of storing facts in the machine, i.e. a non-structural method. One clearly cannot approach such a problem as one of simply storing great collections of facts. Instead such information is to be organized in terms of general semantic properties which exist as components of semantic units. For example, the fact that the Russian sememe ^{S}/пучок/, when it occurs with ^{S}/нейтрон -pl./, is to be translated in English as "beam" (of neutrons) rather than "bunch" is not to be stored as an isolated fact about beams and neutrons; instead, every sub-atomic particle has as a semantic component, to be included in its semantic code, an element specifying in effect that it is a sub-atomic particle; and the proper translation, "beam," is signalled by that property rather than by the individual sub-atomic particles. That is, the information leading to that translation has to be stored only once, not separately for each particle. Similarly any human being has *human* as one of its semantic components, so that various properties of that component are to be specified only once instead of repeatedly for all human semantic units. Moreover, properties that humans share with other mammals are taken care of by a property of the human component specifying, in effect, that humans are mammals, so that the properties of mammals need not be repeated even within the *human* semantic code.

But even with such a compact structural organization of semantic properties, the amount of information that might be desired for coping with a wide variety of difficult translation problems is of tremendous proportions and is in fact apparently unlimited. It will be up to future machine translation workers (present workers have their hands full with grammatical problems) to provide as much of it as possible within the limits of future computer storage capacities, available time, and available methods of analysis. The amount of analysis needed is so great that it will require considerable help from the machines themselves, using programs which will enable them to assist humans in breaking down information contained in encyclopedias and other reference works into semantic properties capable of being incorporated in large semantic networks. Even though the job of semantic analysis can never be completed, a little semantic information is better than none. Future translation machines will have not all of it but only as much as they can be provided with, and that will be a great deal.

A machine translation system of type #14 may be viewed as constructed by putting together the structures of two languages, joining them at the semantic stratum. Thus translation is seen as a series of stratum-to-stratum conversions through a sequence of linguistic strata. Beginning with a graphemic representation the process converts it to a lexemic representation and from there successively through the other strata, ending at the graphemic stratum of the target language. Since the rapid-access storage capacity of the machine is limited, each of the stages is to be performed on the entire text being translated before the next stage is begun. Wherever an ambiguity is encountered in any of the decoding stages, multiple possibilities are passed on to the next stage, where they may hopefully be resolved. For a 7090 such a system can be designed to accept up to 40.000 running words of text (i.e.. the full contents of an issue of a typical scientific journal) as the amount to be processed at each stage.

In one respect the diagram for #14, which portrays the system as symmetrical, is somewhat simplified. It indicates that the morphemes of the target language (which are provided by its morphological rules) are composed directly of graphemes, but there is actually an intervening stage desirable for purposes of economy. This stage makes it possible to deal efficiently with morphographemic alternations in the target language. For example, the English grapheme strings *come* and *com,* the latter being a part of *coming,* are partially different on the graphemic stratum since the latter lacks the ^{G}/e/ which is present at the end of the former. The alternation of this grapheme with zero is a widely occurring phenomenon throughout English vocabulary, which may be accounted for in a morphographemic rule instead of separately for each morpheme affected by it. so that there is a single morphographemic entity of which ^{G}/e/ and ^{G}/Ø/ (i.e., zero) in situations like this are alternate graphemic representations.

Note that the text starts out as ordinary linguistic material, i.e., as a string of graphemes, and that it ends up in a comparable condition, as a string of graphemes of the target language, but that no linguistic material of either type is present while the procedure goes, as it were, deep into the structure of the source language and then emerges from the deep structure of the target language. During these intervening stages the linguistic items are represented by addresses, i.e.. computer reference numbers.

In the complete program there are various intermediate stages which occur between the stages as described. The function of each intermediate stage is to bring from magnetic tape into the rapid-access memory the linguistic information needed for the following stage. Since the linguistic information for upward conversion to the sememic stratum for the lexicon as a whole will doubtless be too voluminous to fit into the rapid-access memory, the intermediate stage preceding it will have to select from the tape just that information which pertains to those lexemes actually occurring in the text. Statistical studies indicate that normally only two to three thousand *different* lexemes will occur in a given text of forty thousand running words. (This intermediate stage will contain a cut-oft" device to cut the text into smaller portions if too many different lexemes happen to be present in some text.)

There is still much work to be done, and most of it is linguistic analysis, which must be more exhaustive and detailed than any undertaken by linguists in the past, because the goal of this analysis must be to provide linguistic rules which will, in the aggregate, give complete specification for every operation required in the translation of any of a wide range of texts which have not even been seen yet. And so it is the (human) programmers and the (human) linguists providing the detailed linguistic information for each of the stages who are, in a sense, now doing the translation of Russian and Chinese scientific articles which have not even been written yet and which only at some future date will be fed into the machine and to the numerous computer instructions and grammatical rules waiting there in readiness for them.

REFERENCES

LAMB. S. M., AND JACOBSEN, W. H., A high-speed large-capacity dictionary system. *Mech. Translation,* 1961, 6, 76-107.

LAMB. S. M., The sememic approach to structural semantics. In A. K. Romney and R. G. D'Andrade (Eds.) Transcultural studies in cognition. *Amer. Anthropologist Suppl,* 1964, 66, 57-78.

NOREEN, A. *Einführung in die wissenschaftliche Betrachtung der Sprache,* translated from Swedish by H. W. Pollack. Halle, 1923.

(Received April 16, 1964)