[*International Conference on Machine Translation of Languages and Applied Language Analysis*, National Physical Laboratory, Teddington, UK, 5-8 September 1961]

ON THE VALUE OF DEPENDENCY CONNECTION

by

DAVID G. HAYS
(The RAND Corporation)

SUMMARY

VALUES are tentatively defined as numbers assigned to types of syntactic relations such that connections of higher value are established in preference to connections of lower value during sentence-structure determination. Given a text in which sentence structures are known, the values of some syntactic relations can be estimated by the following plan: assign value 1 to relations such that no relation is known to have lower value; assign value 2 to relations such that all relations known to have lower value are also known to have value 1; etc. The same procedure can be used in assigning adjectives to order classes, and for similar purposes.*

* This study was supported by the U.S. Air Force Office of Scientific Research. This *paper* is their Technical Note TN-150.

PROGRAMMES for sentence-structure determination (SSD), also called syntactic recognition or parsing programmes, differ in their responses to "ambiguity." Some programmes yield all possible structures of an ambiguous sentence, but most — like the RAND SSD programme - yield only one structure per sentence, namely, the most plausible structure according to the rules of some screening procedure. A programme of either type can fail to produce any "correct" structure for a given sentence, and a programme that seeks the most plausible single structure for each sentence is bound to miss one or more correct structures for any ambiguous sentence. Any programme of the latter type, which will be called heuristic in this paper, avoids certain excesses of the former type, since an exhaustive SSD programme can yield dozens of different structures per sentence if its grammar is weak.* The more powerful the grammar, the fewer the structures yielded by an exhaustive programme, and the more likely the heuristic programme to yield a complete, correct structure — assuming certain unproved qualities for natural language.

Now, a heuristic SSD programme requires many heuristic devices to lead it, as directly as possible, to a single plausible structure; an exhaustive programme can utilize the same devices to rank its structures from most plausible to least. One device is assignment of ***value*** numbers to ***constructions*** (in an immediate-constituent theory) or to ***dependencies*** (in a dependency theory). Faced with a plurality of possible dependency connections, the heuristic programme establishes the one with highest value. Faced with a plurality of complete structures for a single sentence, the exhaustive programme orders them from highest average value to lowest. The concept of value has appeared before in the machine-translation literature, under several names (such as ***urgency*)** [1] [2]. The present

* The author is indebted to Yehoshua Bar-Hillel for discussion of this point.

paper offers an explication of the concept and a method for assignment of values on the basis of empirical data. Some other linguistic applications of the same method are noted in Sec. 3.

1. EXPLICATION

Values are to be assigned in such a way that establishing high-value dependency connections in preference to low-value improves the average accuracy of an SSD programme. In this section, a plan is given for the use of value numbers during SSD. This plan is not the only conceivable plan, and it is not necessarily useful for all types of dependency connections; it is proposed as a scheme for finding the governors of prepositions.*

The RAND SSD programme establishes dependency** connections one by one; a stage in SSD terminates when a new connection is established. At any stage, certain pairs of occurrences are available for consideration; these are the pairs for which ***precedence****** holds. Among the precedence pairs, some (or none; in which case the programme is blocked) show agreement. If, at any stage, occurrence X precedes occurrence Y, and occurrences X and Y agree, a dependency connection can be established between them. At most stages, these two conditions are satisfied by two or more pairs of occurrences; in general, it is impossible to establish all separately possible connections simultaneously, since connections can interfere with one another in three ways. (i) Two connections can involve the same dependent, but an occurrence can depend on at most one other occurrence. (ii) One connection can cut the precedence relation in the other pair. For example, if 2dl and 3d4 in a sentence (see *Fig.1),* then 1p3, 1p4, and 2p4 (but ***not*** 2p3, since XpY only if X or Y or both are independent). If 4d2 be established, then 1p3 and 1p4 become false. (iii) One connection can modify the grammatic type of an occurrence so that it no longer agrees with another. For example, in the sequence $N_{nom}N_{nom}/_{gen}A_{nom}$, the central noun can depend on the preceding

* The author is indebted to Dolores V. Mohr for drawing his attention forcibly to this problem.

** Dependency is a relation over pairs of occurrences; it encompasses virtually all informal syntactic relationships. Dependency is antisymmetric; if X depends on Y (XdY), Y does not depend on X. Cf. [2], pp. 16,17, and [3].

*** occurrence X precedes occurrence Y (XpY) at a given stage of SSD if X is to the left of Y; if X and Y are not connected at that stage, directly or indirectly; if one or both of X,Y are independent at that stage; and if every occurrence between X and Y depends, directly or indirectly, on X or on Y. Cf. [2], p. 14.

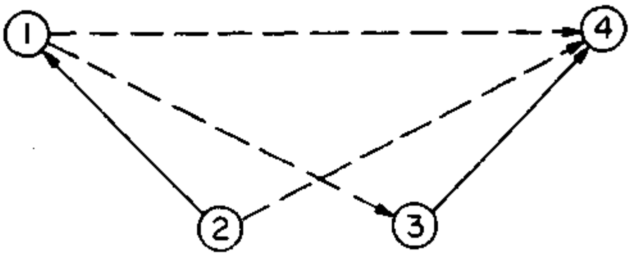
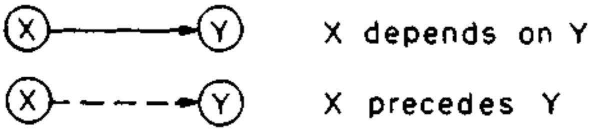


Fig. 1 - Illustrative dependence and precedence relations



noun or govern the following adjective, but it cannot enter both combinations; in one it is genitive, in the other nominative. Hence, in general, it is necessary to choose one connection at a time, make it, and recompute precedences and agreements.

Among the most difficult decisions to be made in many languages* are those concerning prepositions. Prepositions occur with high frequency, they show no morphological agreement with their governors, and they can be separated from their governors by long strings. Sentences are printed in Russian scientific text containing prepositional phrases preceded by sequences of possible governors; if the determination of the preposition's governor were postponed as long as possible, as many as half-a-dozen precedence pairs could be established in some sentences, all showing agreement.

* The following remarks are casual; they serve to motivate, but have no part in, the formal development below. The suitability of the formalism for empirical linguistics is not to be determined by such casual remarks.

An economical, effective plan for selection of the correct governor is badly needed.

In many languages, the information available to any plan includes word order, what preposition is involved, what kind of object it has, and what kinds of possible governors are available.

(i) *Word order.* Most prepositional phrases modify preceding occurrences; some, such as those that open sentences, modify the following ones. In general it is probably easier to locate a following governor; the introductory prepositional phrase, for example, may be a clausal modifier in every instance. In what follows, it is assumed for simplicity that the governor is to be found ahead of the preposition, i.e., that all prepositions with following governors can be handled without recourse to the present procedure. (A major conceptual difficulty is thus avoided; see Sec. 3.) Within a sequence preceding the given phrase, absolute position may be significant; for example, in the occurrence sequence* $P_xN_xN_{gen}$ . . . $N_{gen}P_yN_y$ where any number of genitive nouns can be inserted, N_x ***may*** be the only possible governor of P_yN_y, or N_x and the last N_{gen} may be the only two, unless the last is of a special type and its right to govern a prepositional phrase is transferred to the penultimate genitive, etc. Rules of this order do turn up in natural languages, but they are disregarded here.

Any occurrence X will be called ***accessible*** to another occurrence Y if and only if XpY (X precedes Y) at some stage of a feasible SSD programme. For example, let P be an occurrence of a preposition; establish all possible dependency connections in the sentence without attaching P to a governor. When no further connections can be made, all and only those occurrences X such that XpP at that stage are accessible to P. The plan to be set forth below is intended to choose a governor for each P from among the whole set of accessible occurrences, using values and relative distance as criteria. The accessible occurrences can be ordered as closest to P, next closest, etc., according to their positions in the text sequence (see Fig. 2).

(ii) ***Type of preposition.*** It may be, in some languages, that there exists pairs or larger sets of equivalent prepositions. As a first approximation, it seems best to treat each separately. The method of Sec. 2 below permits grouping prepositions if the data make them equivalent.

(iii) ***Type of object.*** In Russian, some prepositions govern objects of unique cases; others, such as B take objects of various cases. As a first approximation, the type of object associated with a given occurrence

* Hereinafter, capital letters stand for words or word classes, unless it is noted that occurrences of given words or words of given classes are being mentioned. N = noun, P = preposition, X,Y,Z = any word. Subscripts are used for cross classification (as by case: gen. - genitive, x.y = any case) or as dummy indices.

of a preposition can be characterized by grammatic case, but further characterization is probably needed (cf. Harper's study of prepositional equivalents [4]), and can result from the procedure to be outlined.

Initially, let the ***type*** of a prepositional phrase be defined by the preposition that is contains and the grammatic case of the object. If prepositions can be grouped, if objects must be subclassified, or if dependents of the object influence the syntactic functions of the phrase, this definition must be revised.

(iv) ***Type of governor.*** It is difficult, and it may be impossible in terms of traditional parts of speech, to eliminate broad classes of words as not possible governors of a given type of prepositional phrase. However, in in a fixed corpus, it is possible to list all the governors that actually occur. Further characterization of governors is the purpose of value assignment.

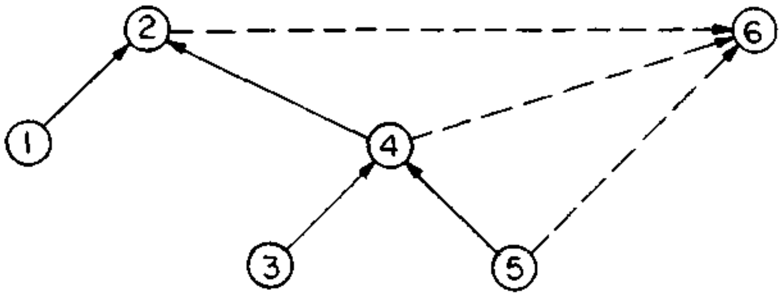


Fig. 2 Accessibility during SSD
Occurrences 2, 4, 5 are accessible to 6

A word will be called a ***potential governor*** of a given type of prepositional phrase in a fixed corpus if it occurs anywhere in the corpus as a governor of that type of prepositional phrase.

The following plan has not been programmed or verified; it is offered as a hypothesis, subject to empirical test. To locate the governor of a prepositional occurrence, P, during SSD:

(1) Connect P with its object. Mark P to indicate the type of phrase that it heads.
(2) Eliminate P if its governor follows it.
(3) Find all occurrences, X, accessible to P.

(4) Eliminate any X that is not an occurrence of a potential governor.

(5) Obtain v(X,P), that is, the value of X as governor of P, for each remaining X.

(During SSD, the values are obtained from a table; v(X,P) is a function of the types of X and P. Discovering the values to be stored in the table is the object of the procedure described below, Sec. 2.)

(6) Take the closest X such that v(X,P) is not greater for any more distant X.

It is not necessary to find all accessible occurrences before connecting P to a governor, provided that any occurrence that is later found to be accessible is tested by steps (4) through (6) of the plan.

This plan can be taken as one definition of the concept of value as applied to syntactic relations. Values are numbers assigned in any fashion such that this plan yields correct results. Two questions remain: can such numbers be assigned to yield correct results throughout a fixed corpus of substantial size? If so, do the assignments tend to stability as the size of the corpus increases indefinitely? In the following section, a method for obtaining answers to these two questions is described.

2. AN EMPIRICAL PROCEDURE FOR THE ASSIGNMENT OF VALUES

Given a corpus in which the structure of every sentence has been determined, the procedure outlined here assigns a set of values to the potential governors of any given type of prepositional phrase, such that the plan set forth in Sec. 1 will yield correct results throughout the corpus, or else it reveals that no consistent set of assignments is possible.

The values of two words, X and Y, with respect to a preposition heading a given type of phrase, say P, only influence the structure of sentences in which both occur accessible to an occurrence of P. Suppose that X occurs to the left of Y; then if X governs P, v(X,P)> v(Y,P), but if Y governs P, v(X,P) ≤ v(Y,P). An inference of this type can be made from each sentence in which two potential governors occur; if more occur in a sentence, all accessible, inferences can be made for each pair consisting of the correct governor and one other potential governor.

Comparing inferences made from two sentences can reveal inconsistencies. Suppose that v(X,P) > v(Y,P) is inferred from one sentence, but v(Y,P) ≥ v(X,P) from another; no assignment of values can satisfy these conditions. Again, suppose that three sentences separately lead to the inferences that v(X,P) > v(Y,P), v(Y,P) > v(Z,P), and v(Z,P) > v(X,P); the last inference is inconsistent with the implication of the first two; namely that v(X,P) > v(Z,P). The object of an empirical assignment procedure is not to gloss over such inconsistencies, but to reveal them; they invalidate the hypothesis of simply ordered values, not the research procedure.

In an infinite corpus, every potential governor of P could have a unique value, and the values could be simply ordered. The rule of Sec. 1 will, however, yield correct results in a finite corpus if the same value is assigned to two or more words, say X and Y, provided that the values of X and Y are not directly comparable in any sentence, and that if y(W,P) > v(X,P) > v(Z,P), then v(W,P) > v(Y,P) > v(Z,P) for all W,Z. If the unique values attainable in an infinite corpus are considered the true set, the procedure to be outlined only guarantees assignment of estimates less than or equal to the true values. Any word that is not a potential governor of P has, effectively, the value zero; if such a word is found, in a later corpus, to govern P, its value must be raised. This process can continue indefinitely, but it would tend in the limit to assign true values.

When the structure of a sentence is known, the set of occurrences accessible to an occurrence of P can be located without recomputation of precedence pairs. Assume that the governor of an occurrence of P is to its left, since sentences in which P's governor follows it are irrelevant in this treatment. Then: (i) The governor of P is accessible to P. (ii) The occurrences from which P's governor derives are accessible to P, up to and including the last that lies to the left of P. These occurrences are ***beyond*** the governor of P, i.e., they lie to its left. (iii) Among the occurrences that depend on the governor of P, only one can be accessible to P; it must lie between P and its governor, and if two or more satisfy this criterion, the rightmost is the only one that is accessible. Applying this rule to the dependents of the accessible dependent of P's governor, etc., we can develop a ***rightmost derivation*** chain headed by P's governor and ending with the closest occurrence to P that does not derive from P. These occurrences are ***between*** P and its governor (see Fig. 3). Every occurrence accessible to P belongs to one of the three categories, (i), (ii), (iii); every value comparison involves the unique member of (i) and a member of (ii) - accessible/beyond- or (iii) - accessible/between.

The assignment procedure consists of the following steps, carried out separately for each type of prepositional phrase, P:

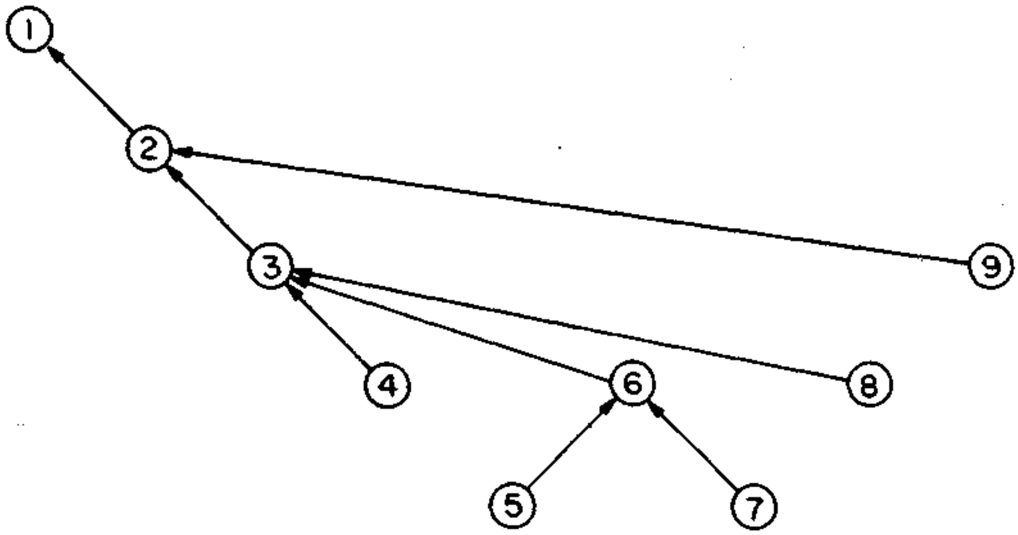


Fig. 3 - Accessibility after SSD

Occurrence 8 is a preposition. Occurrences 1,2,3,6, and 7 are accessible to it. If 8d1 were established, 9d2 would be impossible, but that fact does not influence accessibility. The rightmost derivation chain headed by P's governor (see text) consists of 6 and 7.

(1) Define the set A = $\{X_i\}$, where X_i is a potential governor of P. List the members of A.

Set A is the set * of words to be evaluated.

(2) Define the sets $B_i = \{X_j\}$, where** $X_j \in A$ and X_j occurs accessible/ between X_i and P. List the members of B_i for all i.

If $X_j \in B_i$, then it must be inferred that $v(X_i,P) > v(X_j,P)$.

(3) Define the sets $C_i = \{X_k\}$, where $X_k \in A$ and X_k occurs accessible/ beyond X_i and P. List the members of C_i for all i.

If $X_k \in C_i$, then it must be inferred that $v(X_k,P) \leq (X_i,P)$.

Steps (1) through (3) tabulate the data to be analyzed.

* Curly brackets enclose the members of a set; A = $\{X_i\}$ is read "A is the set whose members are X_i"

** Here $\in$ means "is a member of".

(4) Define $D_{1,0} = \{X_i\}$, where $X_i \epsilon A$ and*** $B_i = \theta$. List the members of $D_{1,0}$. Set $r = 1$.

Set $D_{1,0}$ is a zeroth approximation to the set of potential governors with lowest possible value, say $v = 1$. Unless a word belongs to $D_{1,0}$ it cannot have unit value since if any X_j belongs to B_i (so that $B_i \neq \theta$, i.e., B_i is not null), then $v(X_i, P) > v(X_j, P)$. Setting $r = 1$ is preparatory bookkeeping for the next step.

(5) Define $D_{1,r} = \{X_i\}$, where $X_i \epsilon D_{1,r-1}$ and**** $C_i \leq D_{1,r-1}$, $r \geq 1$. List the members of $D_{1,r}$

No X_i can have a value less than the values of the $X_j \epsilon C_i$. Hence step (5) eliminates some members of $D_{1,r-1}$.

(6) If $D_{1,r} = D_{1,r-1}$, define $D_{1,r} = D_1$ and go on to step (7). If $D_{1,r} = \theta$, stop. Otherwise increase r by unity and return to step (5).

If $D_{1,r} = \theta$, i.e., the set is null, no word can be assigned unit value. In general, if there exists a value V such that V can be assigned to no word, then no value $V' > V$ can be assigned to any word. Hence it is necessary to stop under the specified conditions. If $D_{1,r} = D_{1,r-1}$, then every member of $D_{1,r}$ has unit value, since for each $X_i \epsilon D_{1,r}$, no X_j has smaller value (by step (4)) and every X_j with smaller or equal value has unit value (by steps (5) and (6)).

(7) Define $v(X_i, P) = 1$ if and only if $X_i \epsilon D_1$. Post $v = 1$ for all those X_i. Set $n = 2$.

Step (7) summarizes what is known and sets up a new iteration on n.

(8) Define $D_{n,0} = \{X_i\}$, where***** $X_i \epsilon A - \bigcup_{s=1}^{n-1} D_s$ and $v(X_j, P) < n$ for all $X_j \epsilon B_i$, $n > 1$. Set $r = 1$.

The set $D_{n,0}$ is a zeroth approximation to the set of potential governors with value n. If X_i is a member of $D_{n,0}$, its value is unknown, but all words that occur accessible/between X_i and P have values less than n.

(9) Define $D_{n,r} = \{X_i\}$, where $X_i \epsilon D_{n,r-1}$ and $C_i \leq D_{n,r-1} \cup \bigcup_{x=1}^{n-1} D_s$, $r \geq 1$.

*** Here θ is the null set, with no members.

**** Here $\leq$ means "is included in;" $A \leq B$ means that every member of A is a member of B.

***** Here U means "or" and U means "union"; $X \epsilon A \cup B$ is thus read "X is a member of A or B (or both)," and $\bigcup_{s=1}^{n} D_s = D_1 \cup D_2 \cup \dots \cup D_n$.

The members of C_1 have values less than or equal to the value of X_1 . The union of the D_s includes all words with *smaller* value, and $D_{n,r-1}$ includes, among others, all words with equal value.

(10) If $D_{n,r} = D_{n,r-1}$, define $D_{n,r} = D_n$ and go on to step (11). If $D_{n,r} = \theta$, stop. Otherwise increase r by unity and return to step (9).

This step is a straightforward generalization of step (6).

(11) Define $v(X_1,P) = n$ if and only if $X_1 \in D_n$. Post $v = n$ for all those X_1.

This step is another posting step.

(12) If $A = \bigcup_{s=1}^{n} D_n$, stop. Otherwise increase n by unity and return to step (8).

This step stops the procedure if values less than or equal to n have been assigned to all potential governors of P. Otherwise, the iteration continues with a zeroth approximation of the set of words with value n + 1.

If the procedure is stopped because values cannot consistently be assigned to all potential governors of P, it can be converted into an approximate method, but the plan of Sec. 1 will yield some errors if the approximate method must be used.

In steps (2) and (3) of the assignment procedure, the frequency of occurrence must be shown for each member of each B_i and C_i. That is to say, the number of times that X_j occurs accessible/between or accessible/beyond X_i must be noted. In step (4), if there is no i such that $B_i = \theta$, the approximate method finds those X_i such that the sum of occurrence frequencies over B_i is minimal. In step (8), the same must be done. Approximations can also be used in steps (5) and (9).

An alternative procedure, which complicates the results but avoids introducing error if it is successful, is to subclassify prepositional phrases. Suppose that $X_j \in B_i$ (word X_j occurs accessible/between X_i and P) and $X_i \in B_j$; then it must be inferred that $v(X_i,P) > v(X_j,P)$ and also that $v(X_j,P) > v(X_i,P)$. These two inferences are inconsistent; but they must be made from different sentences, and if the preposition has different objects in those two sentences, P can be resolved into two different phrase types, P' and P". The procedure is then carried out separately for P' and P", but the same inconsistencies can arise again. Indeed, if $X_j \in B_i$, and $X_i \in B_j$ on the basis of two sentences in which the same prepo-sltlon-object pairs occur (and if dependents of the object do not differ, etc.), then subclassification of P is useless.

When observations on a new corpus are to be collated with the analysis of an old, it is necessary to merge the two sets of data and repeat the entire procedure - realizing that the number of inconsistencies can be increased, but not decreased, in the combined data. In principle, the number of distinct values assigned can increase without limit as the size of the corpus is increased; substantively, however, the total number of distinct values should remain small, since speakers of the language are presumably unable to handle many nuances. For the same reason, even if it is necessary to subdivide prepositional phrases according to object type, the number of subclasses should be small. If the number of subclasses or the number of distinct values assigned increases rapidly, the linguist would do well to look for another theory.

The whole assignment procedure described in this section can be programmed for automatic operation on a computer, but most linguists would be unsatisfied with a list of value assignments as the sole output, and with good reason. It would be naive to expect as simple a plan as this to capture the whole of prepositional usage. Syntactic rules of quite different types are probably obeyed by the speakers of every language; only empirical test will show whether rules of the type assumed are obeyed in any language. At least in early applications, therefore, lists of exceptional occurrences will be wanted as part of the output. The procedures described in this section can be programmed easily and run at little expense on relatively large corpora. If only by winnowing exceptional occurrences out of masses of ordinary ones, the procedure should be useful to the linguist.

3. DISCUSSION

Automatic aids to linguistic analysis and lexicographic research are essential because the volumes of data that must be processed are too large for systematic, thorough study by manual techniques. Even relatively unsophisticated lexicography has consumed whole lifetimes of talented effort. In this paper, one computational aid has been presented. Beginning with a definition of ***value*** for certain classes of dependency types, a procedure for assigning estimated values to words in a text has been developed. The procedure requires postedited text, in which the structure of every sentence is known, as input; other procedures will eventually be developed that operate on unedited text [5], but editing is only a small part of analysis, and the analyst benefits if other parts of the task can be made automatic in the meantime.

One conceptual difficulty that remains to be investigated is that of the interaction between direction and distance. If the governor of every prepositional occurrence lies ahead of it in the sentence, accessible/between and accessible/beyond can be distinguished by a simple criterion (as in the present development). If the governor can lie in either direction, a more complicated criterion is required, and what that criterion should be is not obvious.

Essentially the same procedure can be applied in the establishment of order classes of suffixes, adjectives, etc.*

Hill, for example, asserts the existence of six adjective order classes in English [6]; the six adjectives in "All the ten fine old stone houses" belong respectively to classes VI-I. When adjectives of different classes are used to modify a single noun, the adjective belonging to the lower numbered class must stand nearest to the noun. Hence an occurrence of A_iA_jN implies that $c(A_j < c(A_i)$. Data of this type are simpler than those analyzed in Sec. 2, since all of the inequalities are strict. The same ordering problem arises with suffixes that must be added to roots in a particular order. The procedure in Sec. 2 establishes as many suffix "positions" or adjective "order classes" as the data require, and assigns suffixes to positions or adjectives to classes, provided that the ordering is transitive, invariant over noun categories or root types, and unique in the sense that no suffix or adjective belongs to more than one class. Perhaps other applications will occur to other students of language.

REFERENCES

1. RHODES, Ida, ***A New Approach to the Mechanical Translation of Russian,*** Report No. 6295, National Bureau of Standards, Washington, D.C., February 6, 1959.
2. HAYS, D.G., and T.W. Ziehe, ***Studies in Machine Translation - 10 Russian Sentence-Structure Determination,*** The RAND Corporation, RM-2538. April 1, 1960.
3. TESNIERE, Lucien, ***Esquisse d'une Syntaxe Structurale,*** Paris: Klincksieck, 1953.
4. HARPER, K.E., ***Machine Translation of Russian Prepositions,*** The RAND Corporation, P-1941, May 19, 1960.
5. LAMB, S.M., "[Research on Mechanical Translation at the] University of California," ***Current Research and Development in Scientific Documentation,*** No. 7, November, 1960, pp. 71-73.
6. HILL, A.A., ***Introduction to Linguistic Structures,*** Harcourt, Brace, New York, 1958.

* The author is indebted to Sydney M. Lamb and to Duane G. Metzger for the suggestion of these problems.