[*Mechanical Translation and Computational Linguistics*, vol.10, nos.3/4, September and December 1967]

Steps toward a Model of Linguistic Performance: A Preliminary Sketch*

by Robert M. Schwarcz, RAND Corporation, Santa Monica, California‡

This paper discusses the task of formulating a model of linguistic performance and proposes an approach toward this goal that is oriented toward an embodiment of the model as a digital-computer program. The methodology of current linguistic theory is criticized for several of its features that render it inapplicable to a realistic model of performance, and remedies for these deficiencies are proposed. The syntactic- and conceptual-data structures, inference rules, generation and understanding mechanisms, and learning mechanisms proposed for the model are all described. The learning process is formulated as a series of five stages, and the roles of non-linguistic feedback and inductive generalization relative to these stages are described. Finally, the implications of a successful performance model for linguistic theory, linguistic applications of computers, and psychological theory are discussed.

I. On the Goal of a Performance Model

A. WHAT MUST A PERFORMANCE MODEL ACCOUNT FOR?

The range of human use of language is simply enormous, encompassing virtually every situation where two or more people interact, and many more as well. Speaking, listening, reading, writing—how many of our waking hours are spent in performing one or another of these tasks! Even thinking, for the most part, involves the use of mediating linguistic responses. How vast and disparate a range of phenomena, then, must be accounted for by any theory of linguistic performance to be even anywhere near complete. Referring, questioning, requesting, ordering, persuading, relating facts, expressing emotions, greeting, reciting, and soliloquizing are just a few of the various kinds of speech acts that people perform. Letters, novels, plays, poems, textbooks, formal speeches, and technical reports are just a few of the kinds of things that people write. And for each of these types of speech act or writing act, an appropriate identification and a consequent response are required of the listener or reader. To attempt to construct a theory that will account for all this in a unified, rigorous, and comprehensive way is a task before which any contemporary linguist, psychologist, philosopher, or other man of letters must surely pale.

* This research is supported by the U.S. Air Force under Project RAND-Contract No. F44620-67-C-0045-monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq. USAF. Views or conclusions contained in this paper should not be interpreted as representing the official opinion or policy of the U.S. Air Force.

‡ Now at System Development Corporation, Santa Monica, Calif.

Even if one were to account for all these synchronic linguistic behavioral phenomena, he would still be far from having a complete model of linguistic performance. For the essence of natural language is something that is *learned—and* the learning of language is a process that never ceases, as the individual is continually exposed to new words and forms of expression. A theory of linguistic performance cannot be complete without presenting an account of the mechanisms involved in the process of language learning. And these mechanisms must account not only for the learning of the phonological, morphological, and syntactic structure of language but also for the learning of referential associations and their composition in correspondence with the syntactic constructions of the language and for learning the relevance of many different situational contexts to the accurate understanding and non-deviant production of utterances. To learn the immensely complex regularities of a natural language in an environment of exposure to grammatical, semigrammatical, and downright ungrammatical utterances, where explicit instruction is uncommon and insignificant, is a task whose explication requires positing of mechanisms of great power and presumably great complexity as well. The inaccessibility of any introspective evidence of what transpires in the language-learning process and the impracticability of even recording the great masses of behavioral data that accompany the process of language learning in the child produce a situation in which the ingenuity of the theorist is taxed to the utmost in his efforts to account for this complex process. To attempt to devise a model of language learning is clearly not a task for the fainthearted.

Obviously, then, the goal of a comprehensive and rigorous theory of linguistic performance is one that is not about to be achieved for a long, long time. What, then, is a reasonable set of goals to aim for at the present time in endeavoring to pursue as an eventual goal the development of such a theory? The following, as well as the rest of this paper, is an attempt to answer this question.

B. A REASONABLE SET OF TENTATIVE GOALS

In spite of the difficulties outlined in the previous section, the aspiring performance theorist today has, besides his own intelligence, creative talent, and perseverance, a powerful asset working for him in the form of the high-speed electronic digital computer. The computer enables the theorist to state his theories with complete rigor and precision to any desired degree of complexity and then to test those theories by simply observing the behavior of the computer when the programs embodying them are run on it. More appropriate to the question of performance models, the computer is of invaluable aid to the theorist because it can, when programmed with an abstract, formalized model, actually *perform.*

To be sure, the ways in which today's computers can perform are, on the surface at least, quite different from the ways in which a human being performs. A computer does not have eyes, ears, hands, feet, and vocal cords; instead, it must get by with typewriter consoles, card readers, printers, magnetic tapes, drums, disks, and display scopes. Internally, it must get by with a representation of its "knowledge" in terms of discrete digits, and it must process these digits by means of finite, discrete operations. Continuous processes cannot take place inside the memory of a digital computer; instead, they must be approximated by discrete processes involving discrete objects. But it is just these limitations that lead us to a reasonable set of tentative goals for a performance model—namely, a model limited to the sorts of input-output interactions and internal processes that are available on today's computers.

What kinds of capabilities is it reasonable, then, to try to incorporate into a performance model, given these limitations? For one thing, of course, we must dispense with continuous sensory inputs—for the continuous speech signal, we must substitute finite strings composed of the discrete, limited set of characters available on the key punch or typewriter console. Instead of phones or other auditory inputs, we must get by with letters of the alphabet; therefore, it appears that we can effectively bypass the phonological component in a performance model. Similarly, we can bypass the perceptual component on the non-linguistic side of things and input directly to the higher-level conceptual structures as represented in our model. And, finally, we can limit the range of linguistic tasks that our model need perform. Our model need only understand factual information and questions presented to it linguistically, answer questions by means of complete (and possibly incomplete) utterances, and seek factual information by asking questions; it need learn only the morphological and syntactic rules, rules of conceptual inference, and semantic correspondences which will enable it to perform the aforementioned tasks for any natural-language subset and corresponding conceptual-data base that the user wishes to train it to handle. We want our model to be able to carry out a lively interaction with its user, but this interaction is to be carried out solely through the typewriter console and the display scope. And we will allow our model a rampant curiosity for factual information and a desire to conceive of its necessarily limited world in the simplest possible terms, but we will otherwise exempt it from the range of human desires, feelings, and emotions. By taking these shortcuts and imposing these limitations, then, we hope to arrive at a model that will be able to perform linguistically in a limited but interesting way and shed a clearer light on the kinds of processes involved in the acquisition and use of natural language. In the following sections, a preliminary sketch of the form that such a model might take will be exhibited. Let us turn first, however, to the relevance of contemporary linguistic theory to the formulation of a performance model and examine the ways in which its approaches must be modified to render it capable at all of accounting for the acquisition and use of natural language.

II. The Errors of the "Competence Theorists"

A. THE "IDEAL SPEAKER" VERSUS THE "TYPICAL SPEAKER"

Ever since the time of Ferdinand de Saussure, linguistics has concerned itself primarily with the descriptive study of "language" as an abstract, idealized generalization of the linguistic behaviors of certain groups of people. Linguists have traditionally dealt with language under the assumption that there exists one set of rules agreed on and used by all members of the linguistic community, and they have consequently come up with the notion of the "ideal speaker" as one who emits all and only those utterances that are characteristically considered grammatical by native speakers of the language. Even the recently born school of generative linguists (best exemplified by Noam Chomsky), who have instituted considerable progress in linguistic theory by their insistence on formalized descriptions and on getting at the basic underlying structures of languages, have not deviated appreciably from this approach.

The notion of the "ideal speaker," however, is at best a fiction, and a rather absurd one at that. There is simply no single well-defined language that everyone in a linguistic community speaks; this has been pointed out in many places, in particular by Ziff.[1]

The ideal-speaker model is not even valid as an approximation, since it implies a knowledge of the language that is fixed and immutable, whereas, as was pointed out in Section I, natural language is of essence something learned, something of which its speakers' knowledge changes continually over time. Clearly, one must reject the ideal-speaker concept if he is at all interested in developing a model of linguistic performance, particularly one that involves learning (as any realistic model of language use must).

But we must find, now, some concept with which to replace the notion of the ideal speaker, since the actual speaker is still much too complex a quantity to deal with in theoretical terms. To arrive at this concept, we must first abandon the notion that there is a single pool of linguistic knowledge from which every speaker in a linguistic community draws, and recognize that each speaker has his own idea of what his community's spoken language is. We thus reduce the problem of the study of language to the problem of the study of the idiolect (a reduction that, as Ziff pointed out, is valid for procedural as well as theoretical reasons): We shall attempt to justify the observed agreement among different speakers of the same linguistic community as to what their language is in terms of the effects of social interaction in the language-acquisition process and not simply set that agreement forth as an *a priori* assumption. Therefore, let us take as our basic characterization of linguistic phenomena something that we shall refer to as the "typical speaker." The typical-speaker model shall consist of a set of basic mechanisms for understanding, using, and learning language, plus a memory structure for the storage of both linguistic and non-linguistic facts. It will understand and produce utterances, not of an entire language (or even of that subset that the majority of its speakers accept as grammatical), but of a "representative idiolect" that changes continually over time and that, after a certain initial training period, will be extensive enough for the model to communicate successfully on a variety of topics of discourse with other members of its "linguistic community." The typical-speaker concept is clearly applicable to the considerations of a performance model; throughout the rest of this paper, therefore, it will be assumed as representative of the level of explanation that is being sought.

B. THE UTILITY OF PURELY GENERATIVE REPRESENTATIONS IN A PERFORMANCE MODEL

As was pointed out above, the recent trend toward the use of generative models in linguistics, while a significant advance over the earlier taxonomic approach, is still a reflection of the traditional emphasis on the ideal-speaker concept. These models, as characterized by Chomsky,[2] are an attempt at representing the "underlying linguistic competence" of a speaker of the language, or, more precisely, the set of all fully grammatical sentences that a speaker is capable of producing and the structural relations underlying these sentences. Although disclaimers are often made concerning this, it has been tacitly assumed by the proponents of these models that the structure of any viable model of performance will include, as a basis, the data structures of the competence models as they are presently being proposed. It is evident, however, that in their presently proposed form these models, in particular transformational grammar, do not easily lend themselves to use in performance devices that are to be recognizers and learners of language as well as language generators.

A transformational grammar is characterized as a set of phrase-structure rules that generates trees, or "deep structures," which represent the content of the sentence to be generated, and a set of transformations ordered linearly and cyclically, some obligatory and others optional, which maps the deep structures into "surface structures" whose terminal nodes, read left to right, represent the output form of the sentence. Transformational grammar can be faulted as a basis for a performance model from even a purely generative point of view since sentences are treated as strictly independent units (which is obviously not true in ordinary discourse), and no basis is provided for the choice of one derivation in the base component over another in terms of either the immediate linguistic (extra-sentential) context or the over-all situational context (external or internal to the speaker). There are doubts, too, about the efficiency of generative models based on transformational grammar, both in terms of efficiency of expression and in terms of computational efficiency of the resulting performance model. For example, Lieman[3] has come up with a very interesting example of a formal language derived from a classic problem in combinatorial mathematics but with parallels in English and other natural languages, for which arithmetic formalisms (including programming languages such as ALGOL) provide a representation that is far more elegant and efficient in terms of a generative model than any linguistic formalism known, including transformational grammar.

But transformational grammar fares much worse in matters of recognition or parsing. One cannot simply turn the grammar loose and let it generate sentences until it comes up with one that matches the sentence to be parsed, for the simple reason that to do this would take (in some cases, literally) forever. To parse a language generated by a transformational grammar most simply, one would have to have a phrase-structure grammar that assigns structures to all and only the legal surface strings, and then apply inverses of the transformations to the resulting surface structures to recover the original deep structures. There are several problems with this approach, though. Transformational grammars do not explicitly include recognition grammars for surface structure, and there is no known procedure for deriving such a recognition grammar from a transformational grammar. In fact, as Postal[4] has shown in the case of Mohawk, there are natural-language subsets describable by a transformational grammar for which there exists no context-free *or* context-sensitive recognition grammar. Bellert's[5] relational phrase-structure grammar may provide a solution to the problem of describing surface structures, as Bellert[6] herself has shown for the Mohawk example, but no computationally efficient process has been devised to date for parsing with relational phrase-structure grammars (and it is likely that none ever will be, simply because of the combinatorics involved).

What has actually been done in the transformational parsing systems that have been programmed to date is to devise a context-free grammar that assigns surface structures to the sentences generated by the transformational grammar as well as to some others, and then to separate the wheat from the chaff by performing *ad hoc* structural tests and later trying to synthesize the surface structure by means of the transformational grammar. Needless to say, in practice this procedure has turned out to be painfully slow. The second problem in parsing with transformational grammars is that the reverse transformations and their ordering are not explicitly specified, and in some cases it may be impossible to specify some of them. However, it appears that transformational formalisms that employ Katz and Postal's[7] restriction of forbidding irrecoverable deletions will have the property that their transformations will have unique inverses, so that the problem exists only in the matter of finding these inverses. In any case, in terms of their use as a basis for the recognition component in a performance model, transformational grammars require far more comprehensive specification than they presently have, and even with a more comprehensive specification they still lead to notorious inefficiencies in actual performance.

In terms of the learning aspect of a performance model, too, transformational grammar presents problems. Adding new lexical items to already established categories is, of course, no problem. But adding a rule to, or deleting a rule from, the base component requires modifications in the surface-structure component that are essentially unpredictable, and perhaps modifications in the forward and reverse transformational components as well. (Of course, we are here assuming the requirement that the model be at all times intrinsically able to recognize every sentence it can produce, and vice versa.) Similarly, addition of rules to or deletion of rules from the surface-structure component may require modifications in the base or transformational components. In either case, there is no obvious way of co-ordinating all the different types of changes that have to be made simultaneously. (However, there may exist procedures for this that are not obvious but still reliable and computationally efficient.) This problem is certainly not inherent in phrase-structure systems, since there the same grammar is used for both production and recognition of sentences; neither is it obviously inherent in the system of grammar proposed in the next section of this paper.

Other problems may arise in the formulation of a learning model based on transformational grammar, such as the determination of the ordering of the rules. The nature and extent of these problems will only become clear, however, when attempts are actually made to construct such a model.

C. THE PLACE OF SEMANTICS AND ITS MISREPRESENTATION BY COMPETENCE THEORISTS

Modern linguistics seems to have suffered from an overreaction to the position of traditional grammarians that the definition of grammatical categories was to be based on "meaning." To get around what was then the obvious flaw in this position—namely, that meaning was something that was incapable of being formalized and could only be decided on by appealing to the linguistic intuitions of a human speaker—modern linguists chose either to abolish meaning from their linguistic descriptions entirely or to relegate it to a subsidiary position, as Katz and Postal,[7] and Chomsky[2] following them, have done. The contention of these linguists, which is again a concomitant of their insistence on the ideal-speaker concept, is that the set of utterances acceptable in a language should be describable on a purely syntactic basis, with the role of semantics being a purely interpretive and therefore a necessarily secondary one. Notwithstanding the fact that this relegation of semantics to a secondary role is quite counterintuitive (as Quillian[8] has so cogently pointed out, it is nonsense to claim that a speaker produces the syntactic structure of a sentence before its meaning or brings his semantic knowledge into play in the understanding process only after he has produced all possible syntactic structures of the sentence—it is *what the speaker wants to say* that is important), it remains the case that these purely syntactic theories are simply not able to explain the linguistic facts. In an effort to get around this difficulty, Chomsky[2] and others have introduced more and more essentially semantic relations (such as context-sensitive "selectional rules," which block if certain essentially semantic relations between their constituents are not satisfied) into the base component of their transformational grammars. But for every extension that is made, a new counterexample is found, until it appears that, ultimately, to make certain utterances unambiguous (such as Katz and Fodor's[9] example, "Our store sells alligator shoes"), one will need reference to some sort of conceptual model of the world.

Indeed, the base component of transformational grammar, particularly in the recent formulation of Chomsky,[2] is beginning to look more and more like a primitive kind of conceptual network, particularly when Katz and Postal's[7] semantic markers, distinguishers, and projection rules are appended to it. That even this will not do is clear (see, e.g., Bar-Hillel's[10] discussion), but the trend in this direction strongly suggests a reconceptualization of transformational theory along the following lines: Instead of "deep structures" generated by a phrase-structure grammar with a large number of complex rules, we might start with "conceptual structures," which are subgraphs extracted from a "conceptual network" that expresses the speaker's present knowledge of the world. The transformational rules then operate on this conceptual structure to produce a surface structure that is interpreted lexically and phonetically to produce the output form of the sentence. To go the other way, the phonological, lexical, and surface-grammar components would operate on the input to provide its surface structure, which is transformed by reverse transformations into a set of conceptual structures that may then interact with the conceptual network as a whole either to modify it or to extract new conceptual structures to be expressed as sentences, or both. Notice that by performing this reconceptualization we reintroduce semantics into its rightful primary place in utterance generation. But notice, too, that this reconceptualization does not destroy any of the formal properties of the system; in fact, because of the interaction with the conceptual network, full semantic and pragmatic disambiguation is now possible.

In pointing out the errors of the "competence theorists" here, we have not argued with their claim that performance must be viewed as an indication of underlying competence. Nor can any contention with this claim be sensibly made, for any device that is to perform linguistically at all must have some internal representation of the data structures it is to process and the processes it is to perform. But to have a competence model applicable to a theory of performance, we must take into account (1) the difference in competence between different speakers and the changes in a single speaker's competence over time, (2) the practical ease and efficiency with which the model's formalism may be employed in generation, parsing, and modification of itself, and (3) the primacy of semantics, including the speaker's knowledge of the world as represented in a conceptual network, in the generation and understanding of utterances. The errors of the competence theorists have been in not taking these three considerations seriously into account in the formulation of their models. Let us now outline a theory of linguistic descriptions in which care is taken to satisfy these and other considerations, so it may serve as the basis of a model of performance.

III. The Data Structures Underlying a Performance Model

A. THE NECESSITY FOR SIMILARITY OF SYNTACTIC AND CONCEPTUAL REPRESENTATIONS

To those who regard man as the product of a long evolutionary process, it is evident that we must seek the origins of man's language-processing capabilities in the more primitive cognitive abilities that his forebears possessed. In particular, it appears that man's language-processing and language-learning abilities are just specific manifestations of his general capabilities for recognizing patterns and forming concepts. "Words are in the world" is a dictum that is true not only literally but psychologically. It is indeed reasonable to assert, then, that information about language is represented and processed in our brains in the same way as perceptual and conceptual information about the world to which language refers.

Several advantages accrue in the decision to take linguistic and conceptual representations and processes to be the same in a performance model, which alone would justify one in making this decision. The foremost considerations, of course, are those of economy and explanatory power. Economy is gained from the ability to use the same procedures in the processing of both syntactic and conceptual information. Explanatory power derives from the ability to extend the representations and processing mechanisms required to account for language processing to the explanation of other human mental functioning as well. Another advantage of this decision is the elimination of the need to distinguish between the syntactic and the semantic information associated with a lexical item, thereby permitting semantic information to be utilized in syntactic processing, and vice versa, and completely eliminating the boundary that linguists have been trying so unsuccessfully to place between syntax and semantics. (Later we will indicate how this boundary may be reestablished in terms of the different phases of processing.) Finally, this decision provides a powerful heuristic in facilitating the design and construction of a performance model, for it limits the forms of data structures, inference rules, control processes, and learning mechanisms to those that can be used for both syntactic and conceptual information, and it thereby considerably reduces the range of possible representations from which one has to choose in the design of his model.

B. THE SEMANTIC CHARACTERIZATION OF FORMAL LANGUAGES AND ITS EXTENSION TO NATURAL-LANGUAGE DESCRIPTION

In the last section, we argued that the conception of semantics developed by the competence theorists is inadequate. The aim of this subsection will be to give an alternative characterization of the semantics of natural languages that may serve as the basis for a performance model.

We start by describing a characterization of the semantics of formalized languages that stems from the field of logic and was developed most fully there by Tarski.[11] This characterization has been recently extended to natural language by Thompson,[12] and it is central to the design of the DEACON natural-language question-answering system.[13] The formulation is as follows: Each referent symbol in the language is assigned a value out of some possible range of values by the "semantic interpretation," ϕ. The range of values from which the value of $\phi(x)$ for the referent symbol x may be selected is called the "semantic category" or "structural type" of $\phi(x)$; with this semantic category is associated a "part of speech" of which x is a member. For each phrase $X = x_1, \ldots, x_n$ in the language defined by a phrase rule $P \rightarrow P_1; \ldots, P_n$, where $x_1, \ldots, x_n$ have parts of speech $P_1; \ldots, P_n$, respectively, its semantic interpretation is given by a function τ, associated with the phrase rule expanding P, operating on the semantic interpretations of $x_1 \ldots, x_n$; that is,

$$\phi(X) = \tau\,[\phi(x_1), ..., \phi(X_n)]$$

In the propositional calculus, for example, if P and Q are propositional variables, then their semantic interpretations are taken from the range $\{T, F\}$, and the τ's, or "semantic transformations," which assign semantic interpretations to the phrases $(\sim P)$, $(P \wedge Q)$, $(P \vee Q)$, $(P \supset Q)$, and $(P \equiv Q)$, are simply the familiar truth tables for the connectives $\sim$, $\wedge$, $\vee$, $\supset$, and $\equiv$, respectively. Any of the $x_1 \ldots, x_n$ may themselves be phrases as well, with the result that their derived denotations are employed as arguments of τ. An important type of phrase from the logician's point of view is the sentential formula, which always has a semantic interpretation of either T (truth) or F (falsity), and a fundamental concern of logic and mathematics in general is to identify those sentential formulae that always evaluate to truth regardless of the semantic interpretations assigned their constituent variables. Tarski[11] discusses this matter very cogently and in great detail with an illuminating example from the calculus of classes to emphasize his points.

The alert, linguistically oriented reader by now will surely have noticed that the τ's in the above formulation correspond exactly to the projection rules of Katz and Fodor[9] and Katz and Postal.[7] He will also have observed that the above formulation of semantics is too closely tied to a phrase-structure syntax to be by itself of value in the description of natural language, and he might be tempted to argue that, by applying the authors' projection rules only to base structures and deriving surface syntax transformationally from these base structures, the Katz-Fodor-Postal formulation furnishes the necessary "distance" between surface syntax and meaning while preserving their relationship through the transformational rules. This is indeed a very tempting view, especially if one regards the projection rules as appended to the inverse transformational rules to provide a mapping from surface syntax into a representation of meaning, which gives the reconceptualization of transformational grammar that was proposed in the last section. Nevertheless, it, as well as the Tarski-Thompson theory that underlies it, suffers from the defect that it does not treat syntactic and semantic representations in symmetrical terms, as was seen in the first part of this section to be necessary in a realistic model of performance. But the Tarski-Thompson theory lends itself to a simple modification that keeps this symmetry intact and, moreover, enhances the theory's adequacy for describing natural language.

Recall that the "semantic transformation" τ, in the Tarski-Thompson formulation, was associated with a syntactic phrase rule $P \rightarrow p_1 \ldots, p_n$. Now if we replace this phrase rule by a *syntactic transformation* $\rho(x_1, \ldots, x_n)$, *we* have achieved the symmetry that was desired. The transformation ρ may now take the form of a simple phrase rule, a context-sensitive phrase rule, a phrase rule with relational conditions on its application (as in Bellert's[5] relational phrase-structure grammar), a transformational rule of grammar, etc. Each ρ may be associated with one or more τ*'s* and each τ with one or more ρ*'s,* thereby allowing ambiguities to be introduced that may be resolved at a later stage, either through context or through heuristic methods such as the use of some sort of evaluation function. We now have a formulation of grammar and semantics that is highly general, yet which meets our symmetry constraints. The remainder of this section is concerned with the elaboration of the form of these syntactic and semantic transformations and of the form of the syntactic and conceptual data structures on which they operate.

C. A REPRESENTATION FOR CONCEPTUAL DATA STRUCTURES AND ITS EXTENSION TO SYNTACTIC STRUCTURES

Recently, a number of different researchers (including Quillian,[8] Simmons,[14] Longyear,[15] and several European groups), in working with computer applications to natural-language problems, developed representations of natural-language information that take the form of generalized graphs consisting of objects, sets, binary relations, and various combinations of these. At the moment it appears that in these generalized graphs lies a form of representation powerful enough to be adequate for conceptual information and (as will be demonstrated below) syntactic information as well. Let us proceed, then, to describe the form of such a representation.

The entities that may be taken as primitive are *individuals, classes, relations, operators,* and *natural* *numbers.* Several of these may be taken as *paradigmatic,* that is, an essential (universal) part of the system; among these may be the null class, the "indefinite" element, the truth values ("true," "false," and "undefined"), several classes for relations (e.g., reflexivity, transitivity, symmetry), the discourse classes of declarative statement and question, the relations of identity, class membership, class inclusion, logical negation, logical "and," logical "or," greater than, adjacent to, dominated by, preceded by, followed by, co-occurring with, occurring at, beginning at, ending at, continuing at, and likelihood, and the operators of set union, set difference, set intersection, set augmentation by an element, cardinality (number of), abstraction, description, conditional, and function definition. Other entities will be *syntagmatic,* or acquired by the model through experience. All primitive relations and operators, whether paradigmatic or syntagmatic, will be binary (except for logical negation, which is unary, and the conditional operator, which is ternary). (These two can also be reduced to binary relations and operators, since $\sim x$ can be expressed as $x \varepsilon \{F\}$ and Cond (x, y, z) can be expressed as $(x \vec{\wedge} y) \vec{\vee} z$ [where $\vec{\wedge}$ is "ordered-ond", and $\vec{\vee}$ is "ordered-or"].) The primitive structural unit, then, will be of the form *R(a,b),* where *R* is a relation or operator and *a* and *b* can be anything at all. This unit shall be referred to as a *triad.* Since *R, a,* and *b* can themselves be triads, recursive structures of arbitrary depth can be formed. And the function-definition operator permits new relations and operators to be defined in terms of the composition and iteration of already defined ones. Clearly, then, this representation is powerful enough to describe anything that is describable at all in a computational sense, and it should provide an efficient, "natural" form of description as well.

For describing the syntactic structure of utterances, this representation has, obviously, a great deal more power than the usual phrase-structure formalisms. The graph that represents the syntactic structure of an utterance can express not only relations of grouping and concatenation but also all the possible relations of grammatical agreement and dependency. Such a detailed graph, as a representation of the *surface* (in the transformational sense) syntactic structure of an utterance, certainly ought to provide enough information for a complete *syntactic* description of an utterance, and to provide it in a form that lends itself with facility to all phases of the performance process. Approaches toward this more complete description of surface syntax have evolved recently in theoretical linguistics in the relational phrase-structure grammar of Bellert[5] and in computational applications in the DEACON,[13] Protosynthex-II,[16] and DISEMINER[17] question-answering systems, all of which use types of dependency analysis to supplement phrase-structure analysis in their syntactic descriptions. The form of syntactic representation proposed here merely represents a way of carrying these approaches to their ultimate conclusion.

An important part of any linguistic system is its lexical component. In the representation proposed here, lexical items can be handled quite easily and straightforwardly. Each entity that is a lexical item is simply made part of a triad that relates it to its phonemic or graphemic representation or representations. The passage from morphological to syntactic-semantic analysis is then achieved by means of a universal rule, which "looks up" the item with a given phonemic or graphemic representation and substitutes it for its representation. Since each item's full syntactic and semantic concept (i.e., all information about it) is linked to it by means of the triads, the syntactic-semantic component can then operate. Conversely, in the generation process a "terminal" syntactic item, if it is not already a lexical item, is "discriminated" by another universal rule to find that lexical item which is most similar to it in terms of its full concept, and the representation of the resulting lexical item is then output.

D. A REPRESENTATION FOR SYNTACTIC RULES AND ITS EXTENSION TO SEMANTIC RULES

Syntactic rules, whether phrase-structure or transformational, are always composed of at least two parts—a *recognition part* and a *replacement part.* In addition, syntactic rules may have a *go-to,* which indicates the next rule or rules that may be applied. Following this general characterization of a syntactic rule, let us examine the form that a syntactic rule would have for combining and manipulating the type of data structures described in Section III.C.

The recognition part of a rule consists of a list of variables, followed by a list of triads representing relations that must be satisfied by the variables, either with each other or with external parameters. For the recognition part of a rule to "succeed," distinct objects must be found in the graph being searched to correspond to the variables in the rule, such that all the relational conditions specified in the rule are satisfied for those objects. The recognition part may also succeed on the finding of a partial match, that is, one for which not all the relational conditions are satisfied. This is a feature that may be included in the processor to enable the model to interpret ill-formed utterances or utterances containing new lexical items and to express incompletely represented "concepts." In a phrase-structure grammar, for example, the relations specified in the recognition part would be those of identity, adjacency, and linear ordering (preceded by, followed by).

If the recognition part of a rule succeeds on application to a graph, the elements recognized are transformed according to the replacement part of the rule. The replacement part may specify that any of the elements recognized and any of the triads connecting them be deleted from the graph and that new triads connecting the remaining elements or connecting them to new elements be introduced into the graph. Control is then passed to each of the rules specified in the go-to list in turn until one is found whose recognition part succeeds, whereupon the process of rule application begins anew. The processor may also incorporate a backtrack procedure, whereby all rules on the go-to list that have not been tried are saved on a push-down list, and these are tried whenever the current processing path is completed (or, alternatively to a backtrack procedure, all rules on the go-to list are applied in parallel, with an indication of their relative ordering being saved somewhere).

Since conceptual information is represented in the same form as syntactic information, the same form of rule can be used to process conceptual-data structures. The rules can be employed here as either "inference rules" to modify the structure of the conceptual network or as "semantic rules" to analyze some subportion of the conceptual network and map it into a syntactic structure representing an utterance. In general, these rules will analyze and synthesize structures that are *both* syntactic *and* conceptual, so that the syntactic-semantic distinction is obliterated here also. The next two sections, which discuss the use and formation of these rules, will hopefully help to sharpen the somewhat sketchy picture of them that has been given here.

IV. Modeling the Use of Language

A. THE PROCESS OF UNDERSTANDING

The understanding process begins with the input of a string of phonemic or alphanumeric symbols to the model. This string is first converted by the universal lexical-lookup rule into one or more graphs of terminal elements connected by the relations of linear ordering and adjacency. Then a set of rules is applied to parse each of these graphs and to form concurrently a conceptual structure that represents its meaning. For this operation, each new phrase grouping is made concurrently (in the same rule) with the development of new conceptual structure; hence the ρ's and τ's of Section III.B. are combined here in a single application of a rule. There may also be rules (corresponding to the contemporary linguists' transformations), interspersed with the phrase-recognition rules, that do not form any new syntactic structure but merely rearrange an existing one and do not change the corresponding conceptual structure. The purpose of such rules, if they are in fact needed at all, will be to convert semantically equivalent syntactic forms into a "standard" structural form before further rules are applied. When no further rules can be applied to the syntactic structure, the conceptual structures formed are taken as the representation of the meaning of the input.

Depending on the number of conceptual structures formed from the input, it may be either unambiguous, ambiguous, or anomalous (in the sense of Katz and Fodor[9]). In the case of an anomalous input, the processor backtracks, successively relaxing relational conditions on the rules employed until it can come up with an interpretation; if no interpretation is possible at all, the processor will output a "surprise" response. If the input is ambiguous, the processor compares each conceptual structure formed with the conceptual structures contained in its "short-term memory" (a list of conceptual structures corresponding to the most recent inputs to and outputs of the processor) and chooses that structure that has the highest degree of correspondence with the structures in short-term memory. (This is to simulate the effect of preceding discourse in establishing conceptual "set.") Unambiguous inputs, of course, present no problem.

When one conceptual structure has been selected, it is allowed to interact with the entire conceptual network. First, all remaining "indefinite" elements in the structure (corresponding to anaphoric expressions and instances of ellipsis) are "filled in," if possible, from the contents of short-term memory. The remaining action depends on whether the structure is "labeled" a declarative, a question, or neither. If it is neither, the structure is simply entered into short-term memory, and the processor outputs an "acceptance" response. If it is a question, the processor seeks to fill in the questioned item or items, and if it succeeds, it produces the conceptual structure representing what it finds plus the path traversed to reach it from the non-questioned items in the question and generates an output from that structure (as will be described next). If the processor cannot fill in the questioned item, it may either echo the question or return a "don't know" response. If the input is interpreted as a declarative, several things may happen. First of all, the process attempts to "assimilate" the structure into the conceptual network by putting into the network all items and triads that are in the structure but not in the network. For this to be accomplished, at least one element of each new triad must already be in the network, and no relational triad entered may "disagree" with an existing triad (according to the rules of inference that apply to the two triads). If such a disagreement is encountered and the processor cannot resolve it by any means (such as "refining" its conceptual network by adding new relational conditions to appropriate portions of the network), the processor returns a "surprise" response. Second, the entering of this new information may cause a "curiosity" condition to become activated within the network, which will cause a conceptual structure representing a question to be formed so that a question can be generated and output from it. Several different types of curiosity-motivating conditions will be discussed in the next section. If no disagreements or curiosity-motivating conditions are encountered, an "acceptance" response is output. If a single input is interpreted to contain both declaratives and questions, the declaratives are processed first. Whatever the case, the conceptual structures representing the input and the processor's response to it are entered into short-term memory, so that they may be used if necessary in processing the next input.

B. THE PROCESS OF UTTERANCE PRODUCTION

The process of utterance production starts with a conceptual structure that is produced as either a declarative in answer to a question or as an interrogative in response to a curiosity-motivating condition. The process of transforming this structure into an utterance is essentially the inverse of the understanding process, and it is carried out in much the same manner. In one-to-one correspondence with the rules for syntactic parsing and conceptual-structure generation is a set of rules for semantic parsing and syntactic-structure generation. These rules are applied to parse the conceptual structure and to generate a complete syntactic structure. The subgraph of terminal nodes of this graph is then processed by the universal lexical-substitution rule to convert it into a string of phonemic or alphanumeric symbols, which is output by the processor.

The "surprise" response (which may be something like "Huh?") and the "acceptance" response (which may be something like "Mm-hmm") are, for the moment, conceived of as being "canned"' responses that are output by the processor without any intervening semantic-syntactic processing. They are "emotive," as opposed to purely informational, responses, and as a result they fall outside the scope of the tentative performance model envisioned in Section I. Ultimately, since emotive expression is very much a part of natural language, it will have to be integrated into the model as a whole rather than handled on an *ad hoc* basis; but this will have to wait until a mastery of the task at hand, plus a greater understanding of human motivational processes, has been achieved. Such an extension should certainly lead to a vast increase in the performance capabilities, as well as in the over-all complexity, of the model.

The processes of understanding and utterance production as described here are summarized in the flow charts of Figure 1.

V. Modeling the Acquisition of Language

A. THE STAGES OF LANGUAGE LEARNING

The reader who has familiarized himself with the current state of the art in transformational theory may balk at the extreme generality of the specifications presented here so far and therefore may be somewhat unwilling to accept the form of linguistic representation presented here as a superior alternative to transformational theory for the purpose of serving as an underlying basis of a model of performance. The use of a rather unrestricted formalism for the representation of linguistic knowledge does not in itself imply that the way linguistic information is actually represented will involve the use of all the degrees of freedom available within the representation. The subset of the set of representations available in the formalism actually used will depend, rather, on the particular learning mechanisms available to the model for generating these representations, as well as on the particular set of experiences to which the model is subjected. Use of a very general form of representation will allow the theorist, in the formulation of these learning mechanisms, to concentrate on determining just what kinds of information can be extracted from a given experience rather than worrying about what sorts of changes and additions are possible and/or convenient to make within the bounds of his particular constrained form of representation.

The question of what particular learning mechanisms a performance model should embody is still very far from being answered. However, under the hypothesis that the structures formed by these learning mechanisms will look something like those the transformational theorists have been proposing as models of linguistic structure, it is possible to provide an account of the stages of learning through which the formation of these structures can be explained. There are five stages in all, described below in the logical order in which they must occur. (A more comprehensive discussion of these stages and their psychological relevance is presented in my paper, "Linguistic Relativity and the Language Learning Process."[18])

The first stage of language learning is the recognition that certain sequences of sounds, or classes of sequences of sounds (or, in our case, of phonemic or alphanumeric symbols) constitute lexical items. In the case of the recognition that classes of such sequences constitute a single lexical item, the model must learn to discriminate between instances of different classes. At this stage, the model's discrimination must be on the basis of *inherent* features of the stimuli; therefore, some variety of "clustering" procedure would be applicable here.

The second stage is the associating of these lexical items with *referents;* that is, a relation of denotation is established between the lexical item and an individual, class, relation, or operator characterized by a set of semantic features. This process requires the use of some sort of extra-linguistic feedback, which in the human learner is achieved through his other sense modalities but in our model must be achieved through discussed in greater detail in Sec. V.B.)

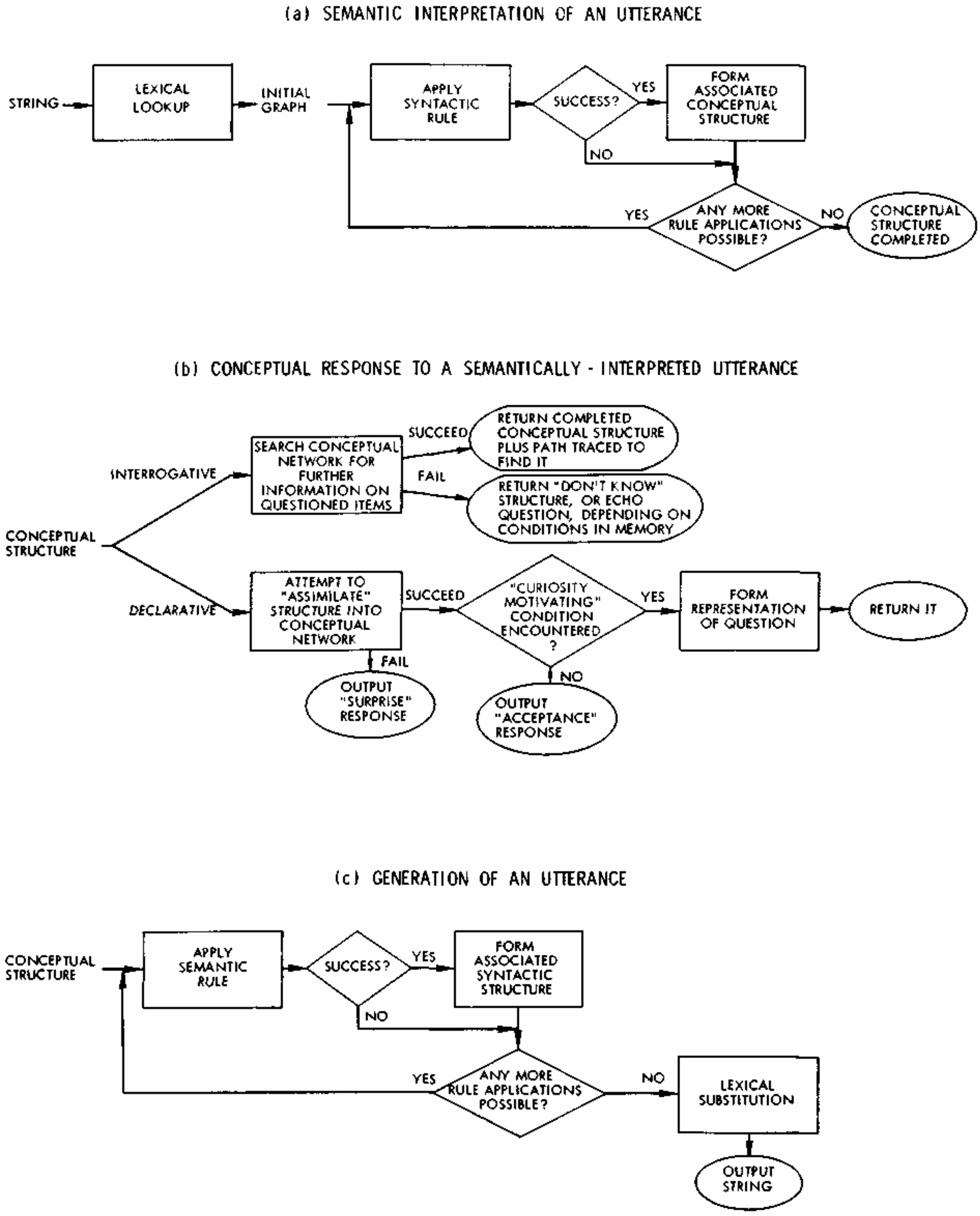

FIG. 1.—The understanding and generation processes

The third stage is learning the co-occurrences and linear ordering relations among lexical items and the the presentation of explicit feedback. (This will be particular triads or combinations of triads that such co-occurrences denote. Experiencing such combinations also adds to the concepts denoted by the lexical items the fact that these concepts are so related. Furthermore, in case one or more of the co-occurring items is ambiguous (has more than one denotation), a particular denotation of that item or those items may be indicated by the combination. Again at this stage, the use of non-linguistic feedback is required.

The fourth stage is the generalization of similar co-occurrences into classes and the resulting formulation of the rules that relate these co-occurrences to their semantic counterparts as *functions* rather than direct associations. This generalization makes possible the application of the resulting rules to novel instances and their incorporation into a recursive hierarchical structure. When the model reaches this stage of learning, it is possible for the first time for it both to understand adequately and to produce completely novel utterances. Unlike the two stages that precede it, the fourth stage is not dependent on non-linguistic feedback; instead, it relies on the inductive generalization capabilities built into the model.

Finally, the fifth stage of language learning is what might be called the "transformation learning" phase. It is the learning of equivalent modes of expression of the same or similar semantic concepts which may be related to each other through simple structural transformations. These transformations in turn lend themselves to generalization and recursive hierarchical ordering, so the entire range of stylistic devices available in a language can be opened to the model for use in the recognition and production of a wide variety of different syntactic forms. The learning of similar transformations on conceptual structures will, in turn, enable the model to form conceptual structures that it had not experienced in linguistic contexts (we may be pushing toward a possible explanation of creative thinking here!) and to express them linguistically. This stage, like the fourth, is realized through the inductive generalization capabilities of the model and does not depend on non-linguistic feedback, although it may be facilitated through explicit linguistic instruction.

In setting forth the five stages of language learning here, no claim or presumption of a strict temporal, maturational sequence should be inferred. Only the *logical* order of the stages is indicated here; on a temporal basis, several of these stages may be taking place simultaneously with respect to different bits of linguistic data. Still, the existence of this logical order has some psychologically relevant implications, which should be testable by experiment. For example, Roger Brown,[19] in an experiment with preschool children, was able to confirm the strict correspondence between syntactic and semantic categories present in the fourth stage of language learning which is broken in the fifth stage. Also, one cannot overlook the possibility that this logical order is indeed reflected in an order of maturational development, since each of the five stages involves different learning mechanisms, which might develop in the order in which they are needed. Let us proceed to look in greater detail at the roles of both explicit feedback and inductive generalization in the learning processes involved in a model of performance.

B. THE ROLE OF EXPLICIT FEEDBACK

As was pointed out in Section V.A, two of the five stages in the language-learning process as it is conceived of here involve the use of non-linguistic feedback. It is clear that this is unavoidable in the second and third stages, because these stages involve the learning, respectively, of the denotations of lexical items and syntactic constructions. The questions remaining are, first of all, in what form such feedback is to be presented to the computer, which represents the actual embodiment of the model, and, second, just how such feedback is to be utilized by the learning mechanisms of the model.

The present-day digital computer has certain basic limitations on the kinds of information it can handle and on the ways available for it to input and output this information. The environment in which a performance model is to operate, namely, in interaction with a human trainer at a remote console, imposes still further limitations. Specifically, at this time the model is limited to interactions it can conveniently carry out with its human user by means of the teletypewriter and the display scope. Clearly, then, one cannot hope to communicate feedback to the computer by means of anything like the sort of primitive sensory feedback that the human learner receives. Even drawing pictures on the face of the scope would not be adequate, since not nearly all the linguistic concepts that one would want the model to learn could be adequately represented by pictures drawn on the face of a scope—and, even if they were, the model would then have the additional task of picture processing, which is indeed a problem in itself (although it might well be handled with much the same mechanisms that are employed in language processing). What is clearly needed is a *language* for communicating feedback to the model. The obvious choice for such a language is the relational graph-structure formalism that is used in the model to represent conceptual-data structures. Such a language could be input to the computer either from the teletypewriter in the form of parenthesized expressions or by constructing graphs with nodes and lines on the display scope, in the manner of Sketchpad.[20] Furthermore, since the inputs and outputs on the semantic end of the generation and understanding processes are these same graph structures, providing feedback explicitly in this form would facilitate to the greatest possible extent its efficient utilization by the learning mechanisms of the model.

What, then, will be the role of this feedback in the actuation of the various learning processes? It will provide initial "concepts" to which lexical items may be associated. It will provide triads and combinations of triads to which particular syntactic forms may be associated by means of new syntactic-semantic rules; this not only will provide referents for the forms concerned but will also facilitate the substructuring of the forms. It will provide a direct means of correcting the model's mistakes and will thus furnish information to the model to be used for changing the relative ordering and/or weighting of rules, for deleting unsuccessful rules, and for adding relational conditions to or deleting relational conditions from existing rules. Finally, feedback will enable the model to discover contexts for the resolution of ambiguity. Some of these tasks will be performable algorithmically from information the feedback provides; others may involve the use of heuristic search and/or evaluation procedures to make the appropriate inferences. All are necessary if the model is ever to "get off the ground" in learning a language.

C. THE BOLE OF INDUCTIVE GENERALIZATION

The major role in the language-learning process, though, is played by the various mechanisms of inductive generalization. Inductive generalization serves a dual purpose: first, it reduces and simplifies the memory structures built up by the model; second, it permits knowledge that the model has gathered from a relatively limited range of experience to be applied to a much wider, perhaps even infinite, range of possible new experiences.

First and foremost, the mechanisms of inductive generalization in a performance model apply to the syntactic-semantic rules and the classes involved in the specification of these rules. Here, the different forms that inductive generalization takes are (1) the formation of classes in order to combine several rules into a single rule, (2) the addition of new items to these classes, (3) the inference of inclusion relations between, and ultimate merging of, classes specified in different rules, and (4) the induction of transformational equivalences among different rules. All these forms of inference take place in the fourth and fifth stages of the language-learning process. The conditions governing their application can be formulated as follows: Classes are formed whenever two or more rules correspond in both their recognition parts and their replacement parts on every item except one; the differing items in each part are then lumped into a class, and the several rules are condensed into a single rule that specifies the new classes to replace the items of which they were composed and in which relations of class membership replace relations of identity. Classes are added to whenever a new element is experienced in a relational context for which this class is specified in some rule. The validity of the application of this rule to the new case is not disconfirmed by either linguistic or nonlinguistic feedback—in case the new element is one introduced by the rule itself, this inference will make the rule recursive. Mergings or inclusions are inferred whenever two or more classes specified in different rules are found to have a sufficient proportion of elements in common, no conflicts or ambiguities would be introduced by the inference, and (in the case of merging) information concerning the behavior of each of the items in each of the rule contexts involved exists or can be obtained through further inferences. The conditions for merging, of course, are more stringent than the conditions for class-inclusion inference. Finally, syntactic transformations are formed whenever two or more syntactic-semantic rules have syntactic recognition parts that are structurally related and identical semantic replacement parts; conceptual transformations may be handled in a somewhat similar manner.

Other forms of inductive inference involved in the learning process include (1) the segmentation and classification of utterances into morphemes and (2) the application of both paradigmatic and syntagmatic rules of conceptual inference to produce changes in the conceptual network above and beyond those actuated by inputs from the user or the mergings and inclusion inferences on conceptual classes mentioned above. Morphemes are classified roughly in the first stage of language learning by means of clustering techniques; this classification is refined in the second stage as sequences of symbols that were previously assigned to the same class are discovered to have referents that differ. Clustering can also be used at this second stage to discover synonymy classes of lexical items, perhaps in a manner similar to that employed by Sparck-Jones.[21] Possibly further study will uncover ways in which clustering can be applied in the later stages as well.

The ability, and sometimes the inability, of the model to make a conceptual inference that leads to a simplification of its conceptual network can create a curiosity-motivating condition within the model's conceptual network, which leads the model to attempt to ask a question of the user in order to obtain information that will enable it to make the inference more reliably. Examples of such conditions are (1) an imminent merging or inclusion inference, where the commonness-of-membership criterion is almost but not quite satisfied, (2) the ability to apply a newly formed (and therefore not well-verified) rule of conceptual inference, and (3) the ability to form a recursive rule of conceptual inference. Another curiosity-motivating condition, which can be thought of as a "table filling-in" motivation, arises from situations of the following type: The model has learned, or inferred, that all members of the set X are in the relation R to some member of the set Y. It then learns or infers that the element a is a member of X. It will then want to know to what member of Y is a in the relation R. For example, if the model learns that all countries have capitals, and then that Mexico is a country, it will ask (or attempt to ask) the user what the capital of Mexico is. An important factor in the design of a realistic performance model is the establishment of enough of the right sort of curiosity-motivating conditions so that the model is able to seek information efficiently, but not so many that it tires the user by asking countless questions. Perhaps some sort of "fatigue factor" will have to be introduced eventually to limit the model's manifestations of curiosity.

We have seen here, first, a sequence of stages whereby a model of linguistic performance could learn language and, second, some of the mechanisms that will be involved in these various stages. On the basis of this model, one can give an answer to the hotly disputed issue as to the role non-linguistic feedback plays in the learning of language—namely, that it is vital in the early stages of language learning but can be effectively eliminated in the later stages because of dependence on various mechanisms of inductive inference. Now that the preliminary sketch of our model has been completed, let us discuss implications it could have for various areas of scientific endeavor if it should prove successful.

VI. Conclusions

A. IMPLICATIONS FOR LINGUISTIC THEORY

As was indicated in Section II, a performance model, as outlined here, implies a reconceptualization of the goals of linguistic theory, namely, the abandonment of the ideal-speaker model and its replacement by the typical-speaker model. Language is to be viewed, not as something existing relative to a society as a whole, but as the net result of a learning process engaged in by each member of the society. And the dictum that performance is to be considered a reflection of competence is to be supplemented by the converse dictum that competence must be viewed as something that can effectively lead to performance.

Another way in which the proposed performance model implies a reshaping of linguistic theory is in the obliteration of traditional distinctions, like "syntactic" versus "semantic," with respect to the *knowledge* of the speaker, and their re-establishment with respect to the different *phases of processing* employed by the speaker. The speaker's knowledge should be thought of, and represented in any model, as a unitary sort of thing, with different portions of it perhaps being employed in different phases of processing.

Finally, an emphasis on performance models in linguistics will ultimately lead to a merging of linguistic and psychological theory. The concern with performance and behavioral measurements in general, the formulation of theories emphasizing processing mechanisms, and the introduction of "motivational conditions" such as curiosity into models of performance all point linguistics in directions that have been followed by psychologists for decades. And psychologists, sensing this shift in direction, will be all too eager to apply mechanisms first formulated in models of linguistic performance to the explanation of other facets of human behavior as well. The resulting interchange between linguists and psychologists cannot fail to be of enormous benefit to both groups of scientists.

B. IMPLICATIONS FOR LINGUISTIC APPLICATIONS OF COMPUTERS

The formulation of a model of linguistic performance as a computer program, constituting as it does an application of computers to natural-language processing, will represent an advance in the state of the art of computational linguistics as well as of linguistic theory. Two major applications of the results of such an advance lie in the areas of fact-retrieval systems and computer-assisted instruction.

A model of linguistic performance, as proposed here, represents in actuality a prototype fact-retrieval system, with the difference that as the model learns facts, it also learns the language in which these facts are expressed. Such a feature is important to a fact-retrieval system that serves a community of users over a reasonable time span, for the linguistic modes of expression will vary from user to user and even for the same user over a period of time; thus it is important for the system to be able to adapt to these changes and variations since they could never all be anticipated in advance by any team of designers. The ability of a fact-retrieval system to *seek* information will result not only in better over-all performance of the system but perhaps also in the setting up of dialogues between a user and an expert who has been most active in answering questions from a subject area with which the user is concerned.

As for computer-assisted instruction, a model of linguistic performance provides the first step toward the development of "intelligent" teaching machines. Teaching machines will no longer have to be explicitly programed; instead, a subject area (including the language associated with it) would be taught to the machines by a human trainer, just as he would teach a student. (The machine, of course, would already possess the "prerequisite" knowledge.) The machine could then analyze the changes in its conceptual network that had been made during its training period and compile a program that would present the material linguistically to the student and build up a "model of the student" from the student's responses, seeking to transform this model of the student so as to bring it into correspondence with the model of the subject area that had been built up during the initial training period. Such a method, if perfected, would enable computer-assisted instruction to provide much more individualized instruction at a far lower cost in terms of human effort than do present techniques.

C. IMPLICATIONS FOR A GENERAL THEORY OF HUMAN MENTAL PROCESSING

As pointed out in the first of this section, a model of linguistic performance points very close to the interests of psychologists in general. Language is a phenomenon very central to human behavior as a whole; the complexity of linguistic behavior, as noted in Section I, is indeed representative of the complexity of human behavior in general. Perception, cognition, learning, motivation, and social interaction—in other words, almost all of the phenomena that experimental psychologists study—are involved in the processing of language. Consequently, the data structures, processing mechanisms, and over-all organization of a linguistic performance model could conceivably be extended to other areas of human mental functioning and perhaps, given computers of sufficient speed, memory size, and sensorimotor capabilities, to human mental functioning as a whole. The latter is a far-off dream, to be sure, but an understanding of the processes involved in the acquisition and use of language will surely go a long way toward bringing to man a deeper understanding of that most profound of all nature's mysteries, the human mind.

Received February 20, 1967

References

1. Ziff, Paul. *Semantic Analysis.* Ithaca, N.Y.: Cornell University Press, 1960.
2. Chomsky, Noam. *Aspects of the Theory of Syntax.* Cambridge, Mass.: M.I.T. Press, 1965.
3. Lieman, S. L. *The Queens Grammar.* (RM-5209-PR.) Santa Monica, Calif.: RAND Corp., January, 1967.
4. Postal, P. M. "Limitations of Phrase-Structure Grammars," in J. A. Fodor and J. J. Katz (eds.). *The Structure of Language: Readings in the Philosophy of Language.* Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1964.
5. Bellert, Irena. "Relational Phrase Structure Grammar and Its Tentative Applications," *Information and Control,* Vol. 8 (October, 1965).
6. -------. "Relational Phrase-Structure Grammar Applied to Mohawk Constructions," *Kybernetika,* Vol. 3 (1966).
7. Katz, J. J., and Postal, P. M. *An Integrated Theory of Linguistic Descriptions.* Cambridge, Mass.: M.I.T. Press, 1964.
8. Quillian, Ross. "Word Concepts—A Theory and Simulation of Some Basic Semantic Capabilities." Unpublished paper, Carnegie Institute of Technology, Pittsburgh, 1965.
9. Katz, J. J., and Fodor, J. A. "The Structure of a Semantic Theory," in J. A. Fodor and J. J. Katz (eds.). *The Structure of Language: Readings in the Philosophy of Language.* Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1964.
10. Bar-Hillel, Yehoshua. "The Outlook for Computational Semantics." *Proceedings of the Conference on Computer-Related Semantic Analysis* (Las Vegas, Nevada, December 3-5, 1965). Detroit: Wayne State University Press, 1965.
11. Tarski, Alfred. "The Concept of Truth in Formalized Languages," *Logic, Semantics and Metamathematics.* Trans. J. H. Woodger. Oxford: Clarendon Press, 1956.
12. Thompson, F. B. "The Semantic Interface in Man-Machine Communications." (RM 63TMP-35.) Santa Barbara, Calif.: General Electric Co., September, 1963.
13. "Phrase-Structure Oriented Targeting Query Language." (RM 65 TMP-64.) Santa Barbara, Calif.: General Electric Co., September, 1965.
14. Simmons, R. F. "Storage and Retrieval of Aspects of Meaning in Directed Graph Structures," *Communications of the ACM,* Vol. 9 (March, 1966).
15. Longyear, C. R. "Memory Structures in DEACON Natural-Language Question-Answering Systems." Paper presented at the 4th Annual AMTCL Meeting, Los Angeles, July, 1966.
16. Simmons, R. F., Burger, J. F., and Long, R. E. "An Approach toward Answering English Questions from Text." (SP-2445.) Santa Monica, Calif.: Systems Development Corp., April, 1966.
17. Klein, Sheldon, Lieman, Stephen, and Lindstrom, Gary. "DISEMINER — A Distributional-Semantics Inference Maker." Paper presented at the 4th Annual AMTCL Meeting, Los Angeles, July, 1966.
18. Schwarcz, Robert. "Linguistic Relativity and the Language Learning Process." (RM-5210-PR.) Santa Monica, Calif.: RAND Corp., December, 1966.
19. Brown, Roger. "Linguistic Determinism and the Part of Speech," *Journal of Abnormal and Social Psychology,* Vol. 55 (1957).
20. Sutherland, I. E. "Sketchpad—A Man-Machine Graphical Communication System." (Technical Report 296.) Lexington, Mass.: M.I.T. Lincoln Laboratory, January, 1963.
21. Sparck-Jones, Karen, "Experiments in Semantic Classification," *Mechanical Translation,* Vol. 8 (1965).