[*Mechanical Translation and Computational Linguistics*, vol.11, nos.1 and 2, March and June 1968]

A Semantic Analyzer for English Sentences

by Robert F. Simmons* and John F. Burger, System Development Corporation,
Santa Monica, California

A *system for semantic analysis of a wide range of English sentence forms is described. The system has been implemented in* LISP 1.5 *on the System Development Corporation (SDC) time-shared computer. Semantic analysis is defined as the selection of a unique word sense for each word in a natural-language sentence string and its bracketing in an underlying deep structure of that string. The conclusion is drawn that a semantic analyzer differs from a syntactic analyzer primarily in requiring, in addition to syntactic word-classes, a large set of semantic word-classes. A second conclusion is that the use of semantic event forms eliminates the need for selection restrictions and projection rules as posited by Katz. A discussion is included of the relations of elements of this system to the elements of the Katz theory.*

I. Introduction

Attempts to understand natural languages sufficiently well to enable the construction of language processors that can automatically translate, answer questions, write essays, etc., have had frequent publication in the computer sciences literature of the last decade. This work has been surveyed by Simmons [1, 2], by Kuno [3], and by Bobrow, Fraser, and Quillian [4]. These surveys agree in showing (1) that syntactic analysis by computer is fairly well understood, though usually inadequately realized, and (2) that semantic analysis is in its infancy as a formal discipline, although some programs manage to disentangle a limited set of semantic complexities in English statements. An inescapable conclusion deriving from these surveys is that no reasonably general language processor can be developed until we can deal effectively with the notion of "meaning" and the manner in which it is communicated among humans via language strings. Several recent lines of research by Quillian [5], Abelson and Carrol [6], Colby and Enea [7], Simmons, Burger, and Long [8], and Simmons and Silberman [9], have introduced models of cognitive (knowledge) structure that may prove sufficient to model verbal understanding for important segments of natural language. Theoretical papers by Woods [10] and Schwarcz [11], and experimental work by Kellogg [12, 13] and Bohnert and Becker [14] have tended to confirm the validity of the semantic and logical approaches based on relational structures that can be interpreted as models of cognition. In each of these several approaches, semantic and logical processings of language have been treated as explicit phases, and each has shown a significant potential for answering questions phrased in nontrivial subsets of natural English. The indication from these recent lines of research is that a natural-language processor generally includes the following five features:

1. A system for syntactic analysis to make explicit the structural relations among elements of a string of natural language.
2. A system for semantic analysis to transform from (usually) multisensed natural-language symbols into unambiguous signs and relations among the computer objects that they signify.
3. A basic logical structure of objects and relations that represents meanings as humans perceive them.
4. An inference procedure for transforming relational structures representing equivalent meanings one into the other and thereby answering questions.
5. A syntactic-semantic generation system for producing reasonably adequate English statements from the underlying cognitive structure.

The present paper describes a method of semantic analysis that combines features 1 and 2 to transform strings of language into the unambiguous relational structures of a cognitive model. The relational structures are briefly described with reference to linguistic deep structures of language; the algorithms for the semantic analyzer are presented and examples of its operation as a LISP 1.5 program are shown.

II. Requirements for a Semantic Analyzer

If a natural language is to be understood in any nontrivial sense by a computer (i.e., if a computer is to accept English statements and questions, perform syntactic and semantic analyses, answer questions, paraphrase statements and/or generate statements and questions in English), there must exist some representation of knowledge of the relations that generally hold among events in the world as it is perceived by humans. This representation may be conceived of as a cognitive model of some portion of the world. Among world events, there exist symbolic events such as words and word strings.

---

* Now at the Department of Computer Sciences, University of Texas, Austin, Texas.

The cognitive model, if it is to serve as a basis for understanding natural language, must have the capability of representing these verbal events, the syntactic relations that hold among them, and their mapping onto the cognitive events they stand for. This mapping from symbolic events of a language onto cognitive events[1] defines a semantic system.

Our model of cognitive structure derives from a theory of structure proposed by Allport [15] in the psychological context of theories of perception. The primitive elements of our model are *objects, events,* and *relations.* An event is defined either as an object or as an event-relation-event (E-R-E) triple. An object is the ultimate primitive represented by a labeled point or node (in a graph representing the structure). A relation can be an object or an event, defined in extension as the set of pairs of events that it connects; intentionally, a relation can be defined by a set of properties such as transitivity, reflexivity, etc., where each property is associated with a rule of deductive inference.

Any perception, fact or happening, no matter how complex, can be represented as a single event that can be expanded into a nested structure of E-R-E triples.[2] The entire structure of a person's knowledge at the cognitive or conceptual level can thus be expressed as a single event; or at the base of the nervous system, the excitation of two connected neurons may also be conceived as an event that at deeper levels may be described as sets of molecular events in relation to other molecular events.

Meaning in this system (as in Quillian's) is defined as the complete set of relations that link an event to other events. Two events are *exactly equivalent* in meaning *only if* they have exactly the same set of relational connections to exactly the same set of events. From this definition it is obvious that no two nodes of the cognitive structure are likely to have precisely the same meaning. An event is *equivalent* in meaning to another event if there exists a transformation rule with one event as its left half and the other as its right half. The degree of similarity of two events can be measured in terms of the number of relations to other events that they share in common. Two English statements are equivalent in meaning either if their cognitive representation in event structure is identical, or if one can be transformed to the other by a set of meaning-preserving transformations (i.e., inference rules) in the system.

We believe that our cognitive model composed of events and relations should include, among other nonverbal materials, deep relational structures and lexical entries at least sufficient to meet the requirements of Chomsky's [16] transformational theory of linguistics. Ideally, in regard to natural language, the structure should also include very deep structures of meaning associated with words. (These have been explored by Bendix [17], Gruber [18], Olney, Revard, and Ziff [19], Givon [20], and others.) In fact, in regard to both transformational base structures and deep lexical structures, representations of text meanings in implementations of the model fall short of what is desired. These shortcomings will be discussed later.

Major requirements of a semantic system for transforming from text strings into the cognitive structure representation are as follows:

1. To transform from strings of (usually) ambiguous or multisensed words into triples of unambiguous nodes with each node representing a correct dictionary sense in context for each word of the string.
2. To make explicit, by bracketing, an underlying relational structure for each acceptable interpretation of the string.
3. To relate each element of the string to anaphoric and discourse-related elements of other elements of the same and related discourses.

Requirements 1 and 2 imply that the end result of a semantic analysis of a string should be one or more structures of cognitive nodes, each structure representing an interpretation that a native speaker would agree is a meaning of the string. Ideally, an interpretation of a sentence should provide at least as many triplet structures as there are base structures in its transformational analysis. It will be seen in the system to be described that this ideal is only partially achieved. Requirement 3 insists that a semantic analysis system must extend beyond sentence boundaries and relate an interpretation to the remainder of the discourse. The need for this requirement is obvious even in simple cases of substituting antecedents for pronouns; for more complicated cases of anaphora and discourse equivalence, Olney [21] has shown it is essential. The present system however, is still limited to single-sentence analysis.

No requirement is made on the system to separate out phases of syntactic and semantic analysis, nor is there any claim made for the primacy of one over the other as is the case in Katz [22] and Kiefer [23]. The system described below utilizes syntactic and semantic word-classes but does not distinguish semantic and syntactic operations. It operates directly on elements of the English-sentence string to transform it into an underlying relational structure.

Although there are numerous additional requirements[3] for an effective semantic theory beyond the three listed above, our present purpose is to describe an algorithm and a system for analysis rather than the underlying

---

[1] The numbered word senses in an ordinary dictionary can be considered as events in a not very elegant but fairly large cognitive model.

[2] From a logician's point of view, the E-R-E structure can be seen as a nested set of binary relations of the form R (E,E) and the referenced statement is a claim that any event can be described in a formal language.

[3] Two of the more important of these are generative requirements beyond the scope of this paper: to generate meaningful natural language sentences from the cognitive structure, and to control coherence in generating a series of such sentences.

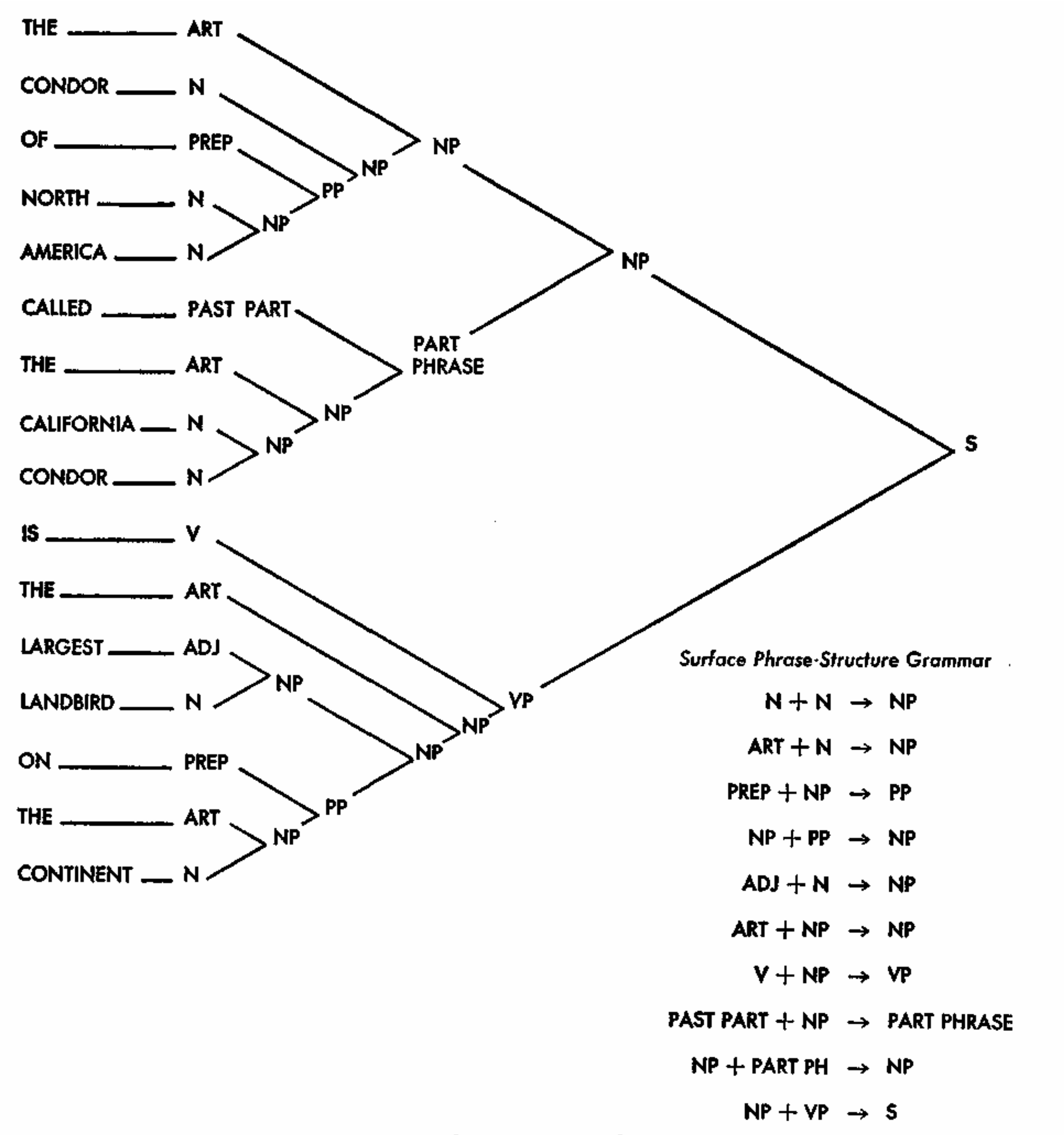

FIG. 1.—Phrase-structure analysis

theory. The basic requirements of the system are sufficient to show the nature of the theory; the means of achieving the first two of these requirements will be described after a more detailed presentation of the cognitive structure model in relation to natural language.

III. Representing Text Meanings as Relational Triples

The semantic system to be described in Section IV can be best understood in the framework of the cognitive model that represents some of the meanings communicated by language. The model uses recursively defined, deeply nested E-R-E structures to represent any event or happening available to human perception. The semantic system relates the symbols in a given string of natural language to this underlying structure of meaning.

Let us take for an example the following English sentence:

A. The condor of North America, called the California Condor, is the largest land bird on the continent.

It is not immediately obvious that this resolves into a set of nested E-R-E triples. Figure 1 shows a surface

N1 + N2 → NP ⇒ N2 MOD N1

ART + N → NP ⇒ N MOD ART

PREP + NP → PP ⇒ NIL

NP + PP → NP ⇒ N1 PREP N2 (N1 head of NP, N2 from PP)

ADJ + N → NP ⇒ N MOD ADJ

ART + NP → NP ⇒ IN MOD ART

V + NP → VP ⇒ NIL

PAST PART + NP → PART PHRASE ⇒ NIL

NP + PART PHRASE → NP ⇒ N1 PAST PART N2 (N2 from PART PHRASE)

NP + VP → NP ⇒ N1 V N2 (N2 from VP)

→ REWRITE

⇒ TRANSFORM

FIG. 2.—Phrase structure plus transformations

syntactic structure for example A with a simple phrase-structure grammar to account for the analysis.

Let us assume that the English lexicon can be divided into two classes—event words and relation words—such that nouns (N), adjectives (Adj), adverbs (Adv), and articles (Art) are event words, and prepositions (Prep), verbs (V), conjunctions (C), etc., are relation words. Let us further assume that there is an invisible relation term in any case where an article or adjective modifies a noun, or an adverb modifies a verb or adjective. Then a set of transformations can be associated with a phrase-structure grammar as in figure 2 to result in the following nested triple analysis of example A:

B. ((((CONDOR OF (AMERICA MOD NORTH)) CALLED ((CONDOR MOD CALIF) MOD THE)) MOD THE) IS (((LANDBIRD MOD LARGEST) ON (CONTINENT MOD THE)) MOD THE)).

Terms such as "MOD," "OF," "ON," "CALLED," and "IS" act as syntactic relational terms in analysis B. Thus the syntactic, relational triple structure is simply obtainable from a phrase-structure grammar in which each phrase-structure rule has associated with it a transformation.

The structure of analysis B is claimed to be of greater depth than the surface structure of figure 1. The base structures underlying adjectival and prepositional modifications are directly represented by such triples as (CONDOR OF (AMERICA MOD NORTH)) AND (LANDBIRD ON CONTINENT). However, the underlying structures for triples containing terms like "CALLED" and "LARGEST" is left unspecified in the above example, so the resulting analysis is by no means a complete deep structure. In addition, we follow a convention of using word-sense indicators as content elements of the structure, rather than following the linguistically desirable mode of using sets of syntactic and semantic markers. (However a word-sense indicator will be seen to correspond to exactly one unique set of syntactic and semantic markers.)

Analysis B is in the form of a semantically unanalyzed syntactic structure. The semantic analysis of B is required to select all and only the structural interpretations available to a native speaker and to identify the (dictionary) sense in which each element of B is used in each interpretation. If the semantic analysis were to operate on a syntactically analyzed form (as in this example), it would also be required to reject any syntactic interpretation that was not semantically interpretable. The result of this, semantic operation would produce analysis C as follows, where subscripts indicate unique sense selections for words:

C. ((((CONDOR$_1$ LOC (AMERICA$_1$ PART NORTH$_1$)) NAME ((CONDOR$_1$ TYPE CALIFORNIA$_1$) Q DEF)) Q DEF) EQUIV (((LANDBIRD$_1$ SIZE LARGEST$_1$ LOC (CONTINENT$_1$ Q DEF)) Q DEF)).

The relational terms have the following meanings:

Q = quantifier; LOC = located at; PART = has part; NAME = is named; TYPE = variety; EQUIV = equivalent; SIZE = size. Since all of these relations are relational meanings (i.e., unique definitional senses of relational words) frequently used in English, they are further characterized in the system by being associated with properties or functions that are useful in deductive inference rules.

Analysis C is now of a form suitable for its inclusion in the cognitive structure. In that structure it gains meaning, since it is enriched by whatever additional knowledge the structure contains that is related to the elements of the sentence. For example, the structure sufficient to analyze the sentence would also show that a condor is a large vulture, is a bird, is an animal; that California is a state of the United States, is a place, etc. The articles and other quantifiers are used to identify or distinguish a triple in regard to other triples in the structure, and the relational terms, as mentioned above, make available a further set of rules for transforming the structure into equivalent paraphrases.

The advantages of this unambiguous, relational triplet structure are most easily appreciated in the context of such tasks as question answering, paraphrasing, and essay generation, which are beyond the scope of this paper. These applications of the structure have been dealt with in Simmons et al. [18], Simmons and Silberman [9], and from a related but different point of view by Bohnert and Becker [14], Green and Raphael [24], Colby [7], and Quillian [5]. Their use in the semantic analysis procedure is described in the following section.

IV. The Semantic Analysis Procedure

The procedure for semantic analysis requires two major stages. First a surface relational structure is obtained by using triples whose form is transformationally related to that of phrase-structure rules, but whose content may include either syntactic or semantic elements. More complex transformations are then applied to the resulting surface relational structure to derive any deep structure desired—in our case, the relational structures of the current cognitive model. Although our procedure derived from a desire for computational economy with some restrictions to psychologically meaningful processes, it is satisfying to discover that the approach is largely consistent with modern linguistic theory as promulgated by Chomsky, Katz, and others. We will note similarities and contrasts, particularly with regard to Katz, as we present the elements of the procedure.

The procedure requires (1) a lexicographic structure containing syntactic and semantic word-class and feature information, (2) a set of Semantic Event Form (SEF) triples, and (3) a semantic analysis algorithm.

Lexical structure.—The lexicon, as mentioned earlier, is an integral part of the cognitive structure model. For each English word that it records it contains a set of sense nodes, each of which is characterized by both a label and an ordered set of syntactic and semantic word-classes or markers. Each syntactic word-class is further optionally characterized by a set of syntactic features showing inflectional aspects of the word's usage. Syntactic classes include the usual selection of noun, verb, adjective, article, conjunction, etc. The normal form for a noun sense of a word is marked by the syntactic feature, Sing(ular); for a verb sense it is marked Pl(ural), Pr(esent). A root-form procedure is used in scanning input words to convert them to normalized form and to modify the relevant syntactic features in accordance with the inflectional endings of the word as it occurred in text.

The semantic word-classes form an indefinitely large, finite set that can never exceed (nor even approach) the number of unique sense meanings in the lexicon. A semantic word-class is derived for any given word *W1* by fitting it into the frame "*W1* is a kind of *W2*." Any members of the set that fit in the frame position of *W2* are defined as semantic classes of W1. Thus semantic word-classes for "man" include "person," "mammal," "animal," "object." A distinguishable set of syntactic and/or semantic word-classes (analogous to Katz's markers) is required to separate multiple senses of meaning for words. For example, minimal sets for some of the senses of "strike" are as follows:

STRIKE = 1 N, SING, DISCOVERY, FIND
2 N, SING, BOYCOTT, REFUSAL
3 N, SING, MISSED-BALL, PLOY
4 V, PL, PR, BOYCOTT, REFUSE
5 V, PL, PR, DISCOVER, FIND
6 V, PL, PR, HIT, TOUCH
etc.

Thus "strike" may be used with the same semantic markers in its senses of "boycott" and "discovery" as long as the syntactic markers N and V (or equivalent semantic markers such as "object" and "action," respectively) separate two possible usages. And, of course, the set of noun usages is similarly distinguished by semantic-class markers. It is a requirement of the system that any distinguishable sense meanings be characterized by a distinguishably different set of markers.

As a consequence of the test frame, a word-class can be defined as a more abstract entity than the words that belong to it, namely, if A is a kind of *B, B* is more abstract than A. The set of word-classes associated with each word is ordered on abstraction level in that, at a minimum, the syntactic class is more abstract than any semantic class. In addition, the semantic classes are ordered from left to right by level of abstraction. Some consequences of this ordering are that each semantic class is a subclass of a syntactic class and that each may also be a subclass of other semantic classes. These consequences are used to considerable advantage in the analysis procedure as described later in this section.

In detailed representation of the lexical structure, it is important to note that semantic classes are not in fact words as shown in the previous examples, but designators of particular senses of the words we have used in the examples to stand for markers. The tabular representation of a dictionary structure in figure 3 will clarify this point.

So far, the use of class relations of words has been sufficient for the task of distinguishing word senses. Occasionally the content has to be rather badly stretched, as in characterizing a branch as a "tree-part" or one sense of bachelor as a "non-spouse." Our underlying assumption is that semantic characterization of a word is a matter of relating it to classes of meanings in which it partakes. Papers by Kiefer [23] and Upton and Samson [25] show the extent to which this kind of classification can be used in accounting for such semantic relations as synonymy, antonymy, etc.

Semantic event forms.—The next important element of the system is a set of semantic event forms which we will refer to as SEFs. The SEF is a triple of the form (E-R-E). The three elements of the triple must be either

	SENSE NBR	SYNTACTIC CLASS	FEATURES	SEMANTIC CLASSES
BOYCOTT	52	N	SING	———
	53	V	PL, PR	———
DISCOVERY	67	N	SING	———
DISCOVER	68	V	PL, PR	———
FIND	98	N	SING	———
	99	V	PL, PR	———
REFUSE	123	V	PL, PR	———
STRIKE	165	N	SING	52, 67, 98
	166	V	PL, PR	53, 123

FIG. 3.—A fragment of lexical structure

syntactic- or semantic-class markers. A subset of the SEFs is thus a set of Syntactic Event Forms, identical in every way to other SEFs but limited in content to syntactic-class markers. The following are examples of SEFs:

Syntactic: (N V N), (N MOD ADJ), (V MOD ADV), etc.

Semantic: (person hit object), (animal eat animal), etc.

The form of an SEF is essentially that of a binary phrase-structure rule that has been transformed to (or toward) the pattern of a transformational base structure sentence. The ordering of the elements thus approaches the corresponding ordering of the elements in a base structure reflected by the triple.

In terms of the cognitive model, an SEF is a simple E-R-E triple whose elements are limited to objects and elementary relations (i.e., no nested events are legitimate elements of a SEF). The set of SEFs serves for the system as its primary store of semantically acceptable relations. For each word in the system, the set of SEFs to which it belongs makes explicit its possibilities to participate in semantically acceptable combinations. A word "belongs" to a SEF if any element of the SEF is a class marker for that word.

The function of SEFs is threefold. First, they act as phrase-structure rules in determining acceptable syntactic combinations of words in a sentence string. Second, they introduce a minor transformational component to provide deep structures for modificational relationships of nouns and verbs and to restore deletions in relative clauses, phrases containing conjunctions, infinitives, participles, etc. Third, they select a sense-in-context for words by restricting semantic class-marker combinations. How these functions are accomplished can be seen in the description of the semantic analysis algorithm, the third requirement for the procedure.

Semantic analysis algorithm.—The form of the semantic analysis algorithm is that of a generative parsing system that operates on the set of SEFs relevant to the interpretation of a particular sentence. The set of SEFs has been shown to be comparable with a modified phrase-structure grammar, and the semantic analyzer generates from the relevant subset of this grammar all and only the sentence structures consistent with the ordering of the elements in the sentence to be analyzed. Since the set of SEFs contains semantic elements that distinguish word-senses, the result of the analysis is a bracketed structure of triples whose elements are unique word-senses for each word of the analyzed sentence.

If we consider the sentence, "Pitchers struck batters," where "pitcher" has the meanings of person and container, "batter" has the senses of person and liquid, and "strike" the senses of find, boycott, and hit, the sentence offers 2 X 3 X 2 = 12 possible interpretations. With no further context, the semantic analyzer will give these twelve and no analytic semantic system would be expected to find fewer.

By augmenting the context as follows, the number of interpretations is reduced: "The angry pitcher struck the careless batter." If only syntactic rules containing class elements such as noun, verb, adjective, and article were used, there would still remain twelve interpretations of the sentence. But by using semantic classes and rules that restrict their combination, the number of interpretations is in fact reduced to one. We will use this example to show how the algorithm operates.

Figures 4 and 5 show minimal lexical and SEF structures required for analyzing the example sentence. The first operation is to look up the elements of the sentence in the lexicon using the root-form logic to replace inflected forms with the normal form plus an indication of the inflection. Thus, the word "struck" was reduced to "strike" and the inflectional features "Sing(ular)" and "Past" were added to the lexical entry for this usage. The syntactic and semantic classes of each word in the lexicon are then associated with the sentence string whose words have been numbered in order of sequence. The resulting sentence with associated word-classes is shown in figure 6.

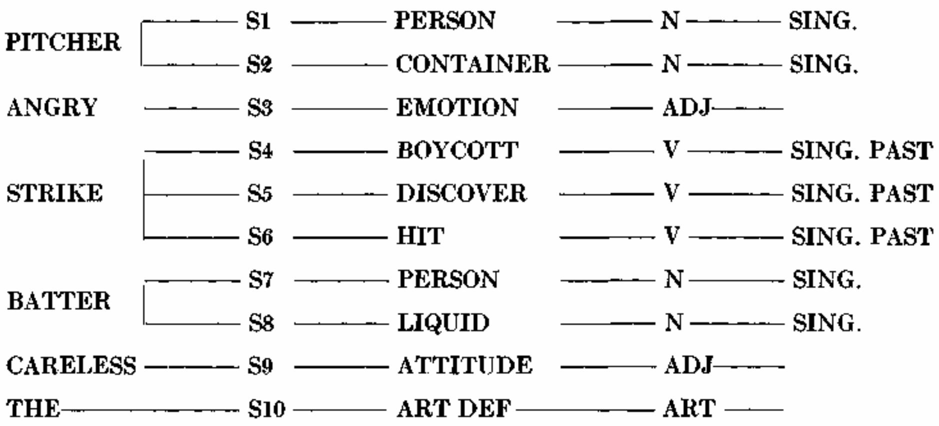

FIG. 4.—Minimum lexical structure

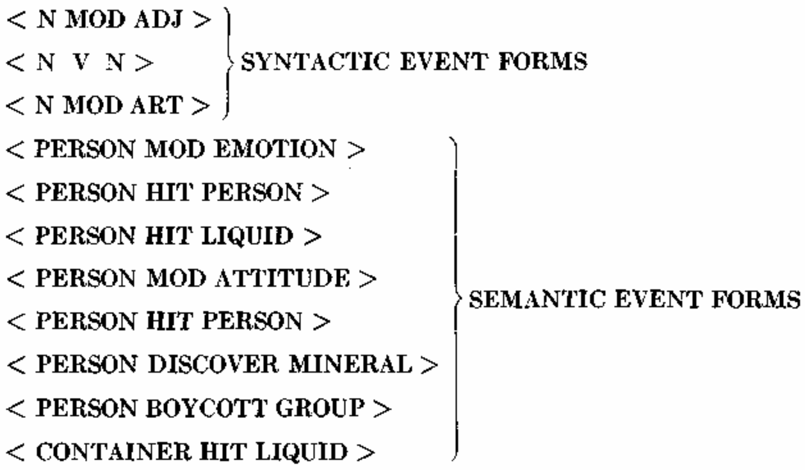

FIG. 5.—Minimal SEF structure

The word-classes are now used as follows to select a minimally relevant set of SEFs:

1. Select from the SEF file any SEF in which there occurs a word-class used in the sentence.
2. Reject every SEF selected by 1 that does not occur at least twice in the resulting subset.
3. Assign word-order numbers from the sentence to the remaining SEFs to form complex triples:

i.e., ((PERSON MOD EMOTION) (3 0 2)
(PITCHER * ANGRY)) .

4. Reject any of the complex triples resulting from 3 that violate ordering rules such as the following:

(N MOD ADJ) ; N > ADJ
$(N_1\ MOD\ N_2)$; $N_1 - N_2 = 1$
$(N_1\ V_1\ N_2)$; $N_1 < V_1$ AND NOT $V_1 < V_2 < N_2$
(V PREP N) ; PREP < N
$(N_1\ PREP\ N_2)$; $N_1 < PREP < N_2$
etc.

A rule such as

$(N_1\ PREP\ N_2)$; $N_1 < PREP < N_2$

means that the word-order number from the sentence associated with the first noun must be less than that associated with the preposition, and that the number associated with the preposition must also be less than that associated with the second noun. The fact that every semantic class implies a corresponding syntactic class allows the set of rules to be expressed in terms of syntactic classes with a consequent increase in generality.

5. Further reduce the surviving set of complex triples by the following operations:

a. If two triples have the same order numbers associated with them, discard the triple whose SEF is made up of the more abstract elements. Thus, since syntactic elements are more abstract than semantic classes in the following pair of complex triples:

((N MOD ADJ) (3 0 2) (PITCHER * ANGRY))
((PERSON MOD EMOTION) (3 0 2) (PITCHER * ANGRY)) ,

the first of the pair is eliminated. The reason for this rule is that the lower the level of abstraction

A	1	2	3	4	5	6	7
B	THE	ANGRY	PITCHER	STRUCK	THE	CARELESS	BATTER
C	ART	EMOTION	PERSON	HIT	ART	ATTITUDE	PERSON
				V			
		ADJ	N	BOYCOTT		ADJ	N
			CONTAINER	V			LIQUID
			N	DISCOVER			N

1. SELECT EVERY SEF IN WHICH ANY ELEMENT OF C OCCURS
2. REJECT EVERY SEF, SELECTED BY 1, THAT DOESN'T OCCUR AT LEAST TWICE
3. ASSIGN WORD ORDER NUMBERS TO THE REMAINING SEFS
   I.E. ((PITCHER * ANGRY) (PERSON MOD EMOTION) (3 0 2))
   ((PITCHER * ANGRY) (N MOD ADJ) (3 0 2))
4. REJECT ANY OF THE TRIPLES IN 3 ACCORDING TO ORDERING RULES SUCH AS THE FOLLOWING:

N MOD ADJ, N > ADJ
N_1 MOD N_2, $N_1 - N_2 = 1$
N_1 V N_2, $N_1 < V < N_2$ EXCEPT:
V PREP N, PREP < V
N_1 PREP N_2, $N < PREP < N_2$
ETC.

WHERE: A > B or A < B MEANS: THE SENTENCE-ORDER NUMBER OF A IS GREATER OR LESS THAN THAT OF B.

FIG. 6.—Example sentence with associated word-classes

the more information carried by a SEF. This rule selects word senses by using a semantic event-form wherever one exists, in preference to a syntactic or more abstract semantic form. b. Eliminate modificational triples, that is, (X MOD Y) where the difference of X and Y is greater than one and there is not a set of MOD triples intervening. This is a more complex ordering rule than is expressible in the form used by step 4. The resulting set of complex triples may be viewed as the relevant subset of semantic grammar sufficient to analyze the sentence. The analysis is performed as a generation procedure which generates all and only the structures permitted by the grammar consistent with the ordering of the words in the sentence. For the example sentence, the following set survived the filtering operations 1-5:

(N MOD ART) (3 0 1)
(N MOD ART) (7 0 5)
N ADJ
(PERSON MOD EMOTION) (3 0 2)
N ADJ
(PERSON MOD ATTITUDE) (7 0 6)
N V N
(PERSON HIT PERSON) (3 4 7).

6. Begin the generation algorithm by selecting a triple whose middle element is a verb, or a class that implies verb. From the grammar resulting from steps 1-5, the selection is:

(PERSON HIT PERSON) (3 4 7).

The primary generation rule is as follows: *Each element of a triple may be rewritten as a triple in which it occurs as a first element.* Thus, starting with (PERSON HIT PERSON) (3 4 7), the following chain of expansions generates the structure of the sentence:

(PERSON HIT PERSON) (3 4 7)
\+ (N MOD ART) (3 0 1)
→ ((PERSON MOD ART) HIT PERSON) ((3 0 1) 4 7)
\+ (PERSON MOD EMOTION) (3 0 2)
→ (((PERSON MOD EMOTION) MOD ART) HIT PERSON)
(((3 0 2) 0 1) 4 7)
\+ (N MOD ART) (7 0 6)
→ ((PERSON . . .) HIT (PERSON MOD ART))
(((3 0 2) 0 1) 4 (7 0 6))
\+ (PERSON MOD ATTITUDE) (7 0 5)
→ ((PERSON . . .) HIT ((PERSON MOD ATTITUDE) MOD ART)
(((3 0 2) 0 1) 4 ((7 0 6) 0 5)) .

A successful generation path is one in which each numbered element is represented once and only once. In such sentences as, "Time flies like an arrow," several successful paths are found. In the generation example above, it can be noticed that "person" in (PERSON MOD EMOTION) and in (PERSON MOD ATTITUDE) is found to occur as a left member in the triple (N MOD ART). This is another important consequence of the fact that a semantic class in context implies a syntactic word-class. The fact that "person" and "N" in the two triples refer to the same word number is the cue that if one is implied by the other, the two triples may be combined. The generation algorithm is a typical top-down generator for a set of phrase-structure rewrite rules. It has the additional ordering restriction for precedence of modifying elements as follows:

7. Adjective modification precedes prepositional modification precedes modification by relative clause precedes article modification precedes predication by a verb. (This precedence rule is not believed to be exhaustive.)

5 6 6
. . . OLD MEN . . . (PERSON

6 7 8 9
. . . THE VERY OLD MEN ....

The operation of the analysis algorithm is rapid in that most possible generation paths abort early, leaving very few to be completely tested. The completed analysis of a path translates the word-order numbers of the complex triples back into English words from the sentence and associates with each of these its identifying sense marker as:

((((PITCHER • PERSON) MOD (ANGRY • EMOTION)) MOD (THE • ART)) (STRUCK • HIT)
(((UMPIRE • PERSON) MOD (CARELESS • ATTITUDE)) MOD (THE • ART))).

A careful examination of the bracketing of the above structure shows that it is the surface syntactic structure of the example sentence in which the word elements have been identified by a marker such that their appropriate dictionary sense can be selected from figure 4. For other usages, the sense of each word can conveniently be identified by the sense number or by its associated set of syntactic and semantic markers instead of by the dotted pairs shown above.

V. Transformations and Embeddings

The result of the semantic analysis algorithm operating on a relevant set of SEFs is a syntactic structure with word-sense identifiers as elements. Although this structure is somewhat deeper than the ordinary phrase-structure analysis as previously discussed, it can best be characterized as a Surface Relational Structure (SRS). Deep structures of any desired form can be obtained by use of an appropriate set of transformations applied to the surface elements. Some of the simpler of these transformations can be seen to be included in ordering of elements within SEFs; some are obtained by the use of rules signified by elements of SEFs, and others are only available by the use of explicit transformation rules applied to the SRS. We will briefly illustrate several complexities associated with embeddings and show our method for untangling these.

Adjectival and adverbial modification.—The general SEF format for this type of modification is (NOUN MOD ADJECTIVE) or (VERB MOD ADVERB) or (ADJECTIVE MOD ADVERB). In each case the event form is taken to approximate a base structure sentence of the form "noun is adjective," etc.[4] The ordering in English sentences is generally of the following form: adjective followed by a noun, adverb followed by an adjective, and verb modified either by a following or preceding adverb. By associating with each SEF the ordinal numbers of the elements of the sentence that it represents, and by then rewriting the elements in the SEF order, the transformation is accomplished. Thus in the following simple case:

0 5 6 0 5
MOD AGE) (MEN MOD OLD)

the precedence rules offer a control on the ordering of the transformations. Thus:

(PERSON MOD AGE)	(9 0 8)
(PERSON MOD ARTICLE)	(9 0 6)
(AGE MOD INTENSIFIER)	(8 0 7)

results in:

((9 0 (8 0 7)) 0 6)
. . . ((MEN MOD (OLD MOD VERY)) MOD THE)....

Relative clauses with relative pronouns.—The system can find the embedded sentences signaled by relative pronouns such as who, which, what, that, etc. The relative pronoun carries a syntactic feature marked "R/P." SEFs of the following form use this marker: (N R/P TH), (PERSON R/P WHO). The marker R/P is a signal to use the generation system recursively according to the rule: (X R/P Y) $\Rightarrow$ RULE R/P: Generate a sentence with X as subject or object and use this sentence as a modifier of X.

Using this mechanism the system can manage examples such as:

---

[4] Although in the current system we allow doubtful base structures such as "verb is adverb," we can modify the system so that it will produce "event is adverb." Thus although presently we have the structure (John (ate MOD slowly) fish), in the future we can express it ([John ate fish] MOD slowly) and the square brackets show that the event "John ate fish" was accomplished slowly.

1.
3 4 5 6
. . . MEN WHO EAT FISH . . .
(PERSON R/P WHO) (3 0 4)
\+ (PERSON V N) (3 5 6)
→ ((3 SUBJ [3 5 6]).. .)
[(MEN SUBJ [MEN EAT FISH]) . . .]

2.
3 4 5 6
. . . MEN THAT FISH EAT . . .
(N R/P TH) (3 0 4)
\+ (N V N) (5 6 3)
→ ((3 OBJ (5 6 3)).. .)
or [(MEN OBJ [FISH EAT MEN]) . . .] .

Infinitives and participles.—An infinitive or a participle that can be identified by the root-form procedure has a syntactic feature S/O marking it as INF, PAST PART, or PRESENT PART. The marker S/O is used analogously to the marker R/P to call a recursion rule: (X X/O Y) ⇒ RULE S/O: Generate a sentence with X as its verb and use this sentence as a modifier of its X, R or Y element, whichever occurs in an SEF with its R.

Using this rule, the system accounts for the following four types of structures as illustrated:

1.
1 2 3 4 5
TO FLY PLANES IS FUN
(FLY S/O INF)) (2 0 1)
(PLANES FLY *) (3 2 0)
(* FLY PLANES) (0 2 3)
[(FLY RELOF [* FLY PLANES]) IS FUN]

2.
1 2 3 4 5 6
FLY /ING PLANES CAN BE FUN
< FLY S/O /ING > (1 0 2)
< PLANES FLY * > (3 1 0)
< * FLY PLANES > (0 1 3)
[(FLY RELOF [* FLY PLANES]) (BE AUX CAN) FUN]
[(PLANES SUBJ [PLANES FLY *]) (BE AUX CAN) FUN]

3.
BROKEN → BREAK + EN
1 2 3 4 5
BREAK +EN DRUMS ARE TINNY
< BREAK S/O EN > (1 0 2)
< * BREAK DRUMS > (0 1 3)
< DRUMS BREAK * > (3 0 1)
[(DRUMS OBJ [* (BREAK T PP) DRUMS]) ARE TINNY]

4.
1 2 3 4 5 6 7
DRUMS BROK/EN IN PIECES ARE TINNY
< BREAK S/O EN > (2 0 3)
< BREAK DRUMS^{-1} * > (0 1 3)
[(DRUMS OBJ [* ((BREAK TENSE PAST-PART) IN PIECES) DRUMS]) ARE TINNY]

It will be noticed in example 4 that we transform the sentence from active to passive.

Other embeddings.—A. few classes of English verbs that have the semantic class of Cognitive Act or Causative have the property of allowing the infinitive to drop its "to" signal. The presence of one of these classes signals that a following embedded sentence is legitimate.
This is managed in accordance with the example:

1 2 3 4 5
MARY SAW JOHN EAT FISH
< PERSON COGACT S > (1 2 0)
< N V N > (3 4 5)
[MARY SAW [JOHN EAT FISH]] .

The presence of a conjunction in an SEF signifies that two or more base structures have been conjoined. The form of this SEF is (X CONJ Y). It allows the generator to generate two similar sentences whose only independent elements are the X and Y terms of the SEF. Thus for "John ate dinner and washed the dishes," the structure results:

[[JOHN ATE DINNER] AND [JOHN WASHED (DISHES MOD THE)]].

One common class of sentences in which the cues are too subtle for our present analysis is typified by "Fish men eat eat worms." The lack of an obvious cue, such as a relative pronoun, is compensated for by the presence of two strong verbs and by the requirement that the embedded sentence use a transitive verb with the subject of the main sentence as its object. We have not yet been able to write a rule that calls our generator twice in an appropriate manner.

Another weakness of the present system is that, although each of the recognizable embeddings can be dealt with individually, their combinations can easily achieve a degree of complexity that stumps the present analysis system. For example, a sentence such as the following thus far defies analysis: "The rods serve a different purpose from the cones and react maximally to a different stimulus in that they are very sensitive to light, having a low threshold for intensity of illumination and reacting rapidly to a dim light or to any fluctuation in the intensity of the light falling on the eye." Apart from the fact that some of the embedding structures of this sentence would go unrecognized by the present analyzer, the complex interaction of such embeddings as signified by the conjunctions, the relative pronoun, and the present participles, would exceed its present logic for disentangling and ordering the underlying sentences.

Explicit transformations.—In the sentence "Time flies like arrows," our system offers the following three syntactic structures:

1. (IMPER(TIME LIKE ARROWS) FLIES) (IMPER (V SIM N) N).
2. (TIME (FLIES LIKE ARROWS) *) (N (V SIM N) *).
3. ((FLIES MOD TIME) LIKE ARROWS) ((N MOD N) V N).

Although item 3 would presumably be eliminated on semantic grounds, we will keep it, for the present example, as an acceptable deep structure that came directly from the SRS analysis procedure. Interpretations 1 and 2, however, are surface structures that need to be further processed to obtain their underlying bases. The cue for the existence of these deep structures is found in the conjunctive use of "like" which is equivalent to the "SIM(ilarity)" sense of its meaning. Although it is possible to use the CONJ signal outlined previously, it is also possible and (because of the dissimilar word-classes of the conjoined elements) desirable to use the following two transformational rules:

A [N_1 (V SIM N_2) N_3] $\Rightarrow$ [[N_1 V N_3] SIM [N_1 V N_2]]

B [N_1 (V SIM N_2) N_3] $\Rightarrow$ [[N_1 V N_3] SIM [N_2 V N_3]

to result in the interpretations:

4. [[IMPER TIME FLIES] LIKE [IMPER TIME ARROWS]].

5. [[IMPER TIME FLIES] LIKE [ARROWS TIME FLIES]].

? 6. [[TIME FLIES *] LIKE [ARROWS FLIES *]].

7. [[TIME FLIES *] LIKE [TIME FLIES ARROWS]].

In Rules A and B, the terms N_1, N_2 and N_3 are subscripted for positional order. Interpretation 6 obviously requires a noun-verb agreement transformation and 7 can probably be eliminated on semantic grounds. However, 4 and 5 are legitimate and desirable base structures.

The general requirement for use of transformational rules is the presence of a distinct cue in the SRS.

The present system does not yet incorporate explicit transformations as exemplified in this section. However, we expect to include these as a final stage in the analysis to obtain the deeper levels of structure required in the cognitive model for answering questions.

VI. Discussion and Conclusions

Computer implementation.—With the already noted exception of the explicit transformational component, the semantic analysis system that has been described is realized in a LISP 1.5 system on the SDC Q-32 interactive Time-Shared System. The program is integrated with a question-answering system that has been briefly described (Simmons and Silberman [9]). Together the two programs account for a large portion of LISP free storage, leaving approximately 12,000 cells of free space for linguistic and factual information. It is immediately apparent that with the Q-32 LISP 1.5's inability to effectively use auxiliary storage devices, the programs are useful primarily for experimentation with the semantic analysis system rather than for any experimentation with large amounts of text.

To overcome these limitations, we are currently programming a system in JOVIAL that uses disk storage and will allow a dictionary of 10,000 words to support text samples of the order of 50,000 words. This larger system will presumably be completed early in 1968. The approach we have found generally acceptable is to use LISP as a convenient system to express and test our initial ideas and to follow the LISP system, once the design has been stabilized, with a large-scale fast-operating program in a language that is more efficient for computation, storage, and retrieval (although less well matched than LISP to human thought patterns).

The actual LISP system has been used to parse most of the examples mentioned in Sections IV and V. The computation time required is typically a few seconds; the interactive delay in accomplishing the analysis on time-sharing rarely exceeds a minute. Authorized users of the SDC Time-Shared System can experiment with the system on-line at their teletypes by requesting from us a set of user instructions.

Some linguistic considerations.—Current structural linguistic theories of syntax and semantics are primarily derived from a generative point of view. Our semantic system is a recognition approach, and consequently comparisons are somewhat more difficult than if it were a generative system. Our aim is to derive from a given English-sentence string a set of deep base structures to represent each of its possible meanings. Elements of the base structures are required to be unequivocal word-sense indicators and bracketings of the structural description to show embedded base structure sentences.

So far, these requirements are consistent with transformational theory. However, no complete set of base structure forms has as yet been specified by transformational linguists, nor have they as yet settled on an appropriate depth for the elements of the structure.[5] In our system, we occasionally deviate from some forms of base structure that have been specified (i.e., we use such doubtful forms as VERB-MOD-ADVERB and VERB-PREP-NOUN), and we are not yet able to obtain many kinds of deep structures such as (SOMETHING MODIFIES SOMETHING) for derived forms such as the word, MODIFICATION.

Transformational theory in generating an English-sentence string begins with the generation of a set of underlying phrase-markers whose elements are syntactic and phonological markers and features, then applies transformations to embed and modify the base phrase-markers, and finally transforms the structured set of syntactic and phonological markers to a selection of phonemic elements whose combinations make English words. Katz currently takes the generation of a set of base structures (i.e., underlying phrase-markers) as one of the requirements of his semantic theory. Using a dictionary and a set of projection rules, he derives semantic interpretations in which the elements of phrase-markers are combinations of semantic markers. Kellogg [13] has implemented a recognition scheme for semantic interpretation which, although with some important modifications, largely follows Katz's scheme to successfully translate from a subset of English into an unambiguous logical notation. We take Kellogg's work as a strong empirical

---

[5] See Section II and its references [16-20] for an explication of this point.

indication that Katz's approach is, in the main, a valid and usable system for semantic analysis.

Katz's dictionary includes syntactic and semantic markers, selection restrictions, and distinguishers. The selection restrictions in conjunction with projection rules have the function of restricting combinations of word senses to avoid semantically nonsensical or meaningless statements. Our system also includes syntactic and semantic markers, but the function of selection restrictions and projection rules is accomplished in what we believe is a theoretically simpler and more elegant fashion.

Given an example like the phrase "angry pitcher," Katz might have the following structure of semantic markers and selection restrictions:

ANGRY 1. ADJ (EMOTION . . . .) < ANIMATE . . .>
PITCHER 1. N (ANIMATE, PERSON ... <.. . SR >
2. N (INANIMATE, CONTAINER . . .)
<. . . .SR >.

The operation of a projection rule in this modification example is to allow the combination of angry_1 with pitcher_1 and to prohibit the combination of angry_1 with pitcher_2 by use of the selection restriction < animate > which requires the head of the resulting structure to have the marker "animate."

In contrast, our system, while having similar syntactic and semantic markers, achieves the same effect gained by the above selection restrictions and projection rules by the use of the following SEF:

(ANIMATE MOD EMOTION) .

As long as there is no SEF such as (INANIMATE MOD EMOTION) or (CONTAINER MOD EMOTION), the phrase is restricted to a single interpretation. We thus argue that selection restrictions can be dealt with on the semantic level in the same manner as they are on the syntactic level: by a set of rules governing the legitimate content of phrase structures.

Starting as we do from graphemic representation of words in English-sentence strings, we first replace word elements with sets of syntactic and semantic markers and then derive base structures with the aid of SEFs (essentially a phrase structure component) followed by an explicit transformational component. The resulting highly interrelated base structures are taken in our system as the meaning of the sentence.

Consequently, in a generation system (that we have not yet constructed) we would select a set of base structures whose elements are labels identifying particular sense meanings, transform these in various ways—changing syntactic and semantic markers appropriately—to form a sentence that embeds the set, then find words with corresponding patterns of syntactic and semantic markers, and modify these words by use of syntactic inflectional features to produce a grammatical and meaningful English sentence.

(IMPER (V PREP N) N)
(N (V PREP N) INTRANS)
((N MOD ADJ) V N)

It can be seen that in both analytic and generative approaches in our system there is no obvious requirement for projection rules of the type Katz posits. However, if, as a result of the various transformations, the original set of semantic and syntactic markers is changed to the point that the set no longer corresponds to a word sense associated with a single English word, there is obviously a requirement to discover a combination of two or more existing sense meanings that we can combine to account for the set of markers. If this were required, the rules of combination would probably correspond to Katz's projection rules. However, in our view it is by no means clear that there is any notable difference between such projection rules and other transformational and phrase-structure type rules required for generating sentence strings. In the recognition algorithm there is no obvious need for combining markers associated with word senses to derive the underlying deep structures.

Katz points out [22] that projection rules for combining subject, verb, and object elements into sentence meanings are essentially rules for embedding nominal elements with verbs into structures like sentences. In our structure, any base structure sentence is represented by a triple of sense identifiers[6] (i.e., a sentence) or some combination of sense identifiers and references to other base structure sentences (i.e., a sentence with S as an element). So in this case, too, the function of projection rules in our recognition algorithm is completely served by SEFs and transformational rules.

Conclusions.—As a result of these arguments and our ability to analyze sentences without projection rules, we conclude that at least for a semantic recognition system, the function of selection restrictions and projection rules can be most easily accomplished in the transformed phrase-structure format of SEFs and a generation algorithm.

Second, our experimentation surprises us in indicating that a semantic analysis system is remarkably similar to a syntactic analysis system, except for its augmentation of relatively few syntactic-class markers and rules of combination by a myriad of semantic classes and rules of combination for these. In support of this point it is quite interesting to note that if the system is limited to syntactic classes, it will produce all and only the surface syntactic structures for a sentence quite in the manner of any other good syntactic parsing system. For example, using only syntactic markers, the following analyses emerge for, "Time flies like arrows":

(IMPER(TIME LIKE ARROWS) FLIES) ,
(TIME (FLIES LIKE ARROWS) INTRANS) .
((FLIES MOD TIME) LIKE ARROWS) .

Lest this be taken as a sign of semantic weakness, it should be recalled that the system requires that any two distinguishable word senses have at least one different element in their marker sets. As a consequence, SEF

---

[6] These identifiers point both to a word form and to a unique set of markers.

rules can always be written to restrict the combinations of a word sense with any other word sense. (However, it is possible that SEFs might be required to become complex triples in order to distinguish very fine differences of meaning.)

A third finding from this study, though it is not strong enough to be a conclusion, is that wherever an embedded sentence leaves surface traces, the process of recovering that embedded structure rarely requires more than a single transformation. This finding is adequately supported by the examples of embedding in Section IV. It is also apparent that, when (in addition to relative pronouns and inflectional markers such as infinitive, participles, etc.) we consider the derivational affixes such as -ate, -ion, -ly, -ment, etc., there are a great many surface cues that are not yet generally used. Recent work by Givon [26] and Olney et al. [19] suggests how these cues signal embeddings. Studies of anaphoric and discourse analysis also suggest that most deletion transforms usually leave some detectable trace—at least in printed text environments. However, the problem of restoring deletions is a complex and difficult one.

The consequence of these conclusions, if they survive continued study, is that deep underlying structures of sentences with unique identification of word sense in context can be obtained with considerably less mechanism than most previous experience with transformational theory and recognition systems would lead one to believe. This consequence remains as a hypothesis. We can support it further by showing that our approach applies as well to large amounts of textual material supported by large dictionaries as it does in small-scale application to a wide variety of structures.

References †

1. Simmons, R. F. "Answering English Questions by Computer: A Survey." *Communications of the ACM* 8, no. 1 (1965): 53-70.
2. Simmons, R. F. "Automated Language Processing." In *Annual Review of Information Science and Technology,* edited by C. A. Cuadra. New York: Interscience Publishers, 1966.
3. Kuno, S. "Computer Analysis of Natural Languages." Presented at Symposium on Mathematical Aspects of Computer Science, American Mathematical Society, New York, April 5-7, 1966 (available from Harvard Computation Center).
4. Bobrow, D. G.; Fraser, J. B.; and Quillian, M. R. "Automated Language Processing." In *Annual Review of Information Science and Technology,* edited by C. A. Cuadra. New York: Interscience Publishers, 1967.
5. Quillian, M. R. "Semantic Memory." Doctoral dissertation, Carnegie Institute of Technology, February 1966.
6. Abelson, R. P., and Carrol, J. D. "Computer Simulation of Individual Belief Systems." *American Behavioral Scientist* 9 (May 1965): 24-30.
7. Colby, K. M., and Enea, H. "Heuristic Methods for Computer Understandings of Natural Language in Context-restricted On-Line Dialogues." *Mathematical Biosciences* 1 (1967): 1-25.
8. Simmons, R. F.; Burger, J. F.; and Long, R. E. "An Approach toward Answering English Questions from Text." In *Proceedings of the AFIPS 1966 Fall Joint Computer Conference.* Washington, D.C.: Thompson Book Co.
9. Simmons, R. F., and Silberman, H. F. "A Plan for Research toward Computer-aided Instruction with Natural English." SDC document TM-3623/000/00, August 21, 1967.
10. Woods, W. A. "Procedural Semantics for a Question-Answering Machine." In *Proceedings of the AFIPS 1968 Fall Joint Computer Conference.* Washington, D.C.: Thompson Book Co.
11. Schwarcz, R. M. "Steps toward a Model of Linguistic Performance: A Preliminary Sketch." RAND Corporation, memorandum RM-5214-PR, Santa Monica, Calif., January 1967.
12. Kellogg, C. H. "On-Line Translation of Natural Language Questions into Artificial Language Queries." SDC document SP-2827/000/00, April 28, 1967.
13. . "A Natural Language Compiler for On-Line Data Management." In *Proceedings of the AFIPS 1968 Fall Joint Computer Conference.* Washington, D.C.: Thompson Book Co.
14. Bohnert, H. G., and Becker, P. O. "Automatic English-to-Logic Translation in a Simplified Model." IBM, Thomas J. Watson Research Center, Yorktown Heights, N.Y., March 1966. AFOSR 66-1727 (AD-637 227).
15. Allport, F. H. *Theories of Perception and a Concept of Structure.* New York: John Wiley & Sons, 1955.
16. Chomsky, N. *Aspects of the Theory of Syntax.* Cambridge, Mass.: M.I.T. Press, 1965. (M.I.T. Research Laboratory of Electronics, Special Technical Report no. 11.)
17. Bendix, E. H. "Semantic Analysis of a Set of Verbs in English, Hindi, and Japanese." Doctoral dissertation, Columbia University, February, 1965.
18. Gruber, J. S. "Studies in Lexical Relations." Doctoral dissertation, Massachusetts Institute of Technology, 1965.
19. Olney, J.; Revard, C.; and Ziff, P. "Toward the Development of Computational Aids for Obtaining a Formal Semantic Description of English." SDC document SP-2766/000/00, August 14, 1967.
20. Givoń, T. "Some Noun-to-Noun Derivational Affixes." SDC document SP-2893/000/00/, July 20, 1967.
21. Olney, J. C. "Some Patterns Observed in the Contextual Specialization of Word Senses." *Information Storage and Retrieval 2* (1964): 79-101.
22. Katz, J. J. "Recent Issues in Semantic Theory." *Foundations of Language* 3 (1967): 124-94.
23. Kiefer, F. "Some Questions of Semantic Theory." *Computational Linguistics* 4 (1965): 71-77.
24. Green, C., and Raphael, B. "The Use of Theorem-proving Techniques in Question-Answering Systems." In *Proceedings of the AFIPS 1968 Fall Joint Computer Conference.* Washington, D.C.: Thompson Book Co.
25. Upton, S., and Samson, R. W. *Creative Analysis.* New York: E. P. Dutton & Co., 1963.
26. Givon, T. "Transformations of Ellipsis, Sense Development, and Rules of Lexical Derivation." SDC document SP-2896/000/00, July 22, 1967.

---

† Corporation documents for System Development Corporation, Santa Monica (SDC) and IBM, Yorktown Heights, New York, may usually be obtained from the author upon request.