TOWARD THE DEVELOPMENT OF A MACHINE WHICH COMPREHENDS

Robert K. Lindsay

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
Carnegie Institute of Technology

May 1961

PREFACE

I wish to express my gratitude to those who have helped me in the development of the ideas put forth in this paper. Special thanks go to Professor Herbert A. Simon, Professor Walter R. Reitman, Professor James G. March, and Dr. Allen Newell.

The work reported in this paper was sponsored by the Ford Foundation, to which I am also grateful.

My appreciation is also extended to my wife, Jane, for much encouragement and many hours of assistance in the preparation of several revisions and the final manuscript.

R. K. L.

May 26, 1961
Austin, Texas

Table of Contents

INTRODUCTION

Psychological theory attempts to explain how thinking--the subject matter of psychology--is possible by a brain composed of single mechanistic elements--the basic assumption of psychology. The problem of programming digital computers to behave in complex fashions is equivalent to this aspect of the psychological problem. Today automata theorists agree that no fundamental barrier blocks the development of machines which can think, by any reasonable definition of the term. However, the precise techniques for implementing general thinking processes have been only partially developed.

An example of a high-level, general thinking process is comprehension: the understanding of passages of a natural language. The problem of constructing a machine which comprehends is of interest for two reasons. First, a machine whose detailed construction was known would aid the development of other types of intelligent automata. Second, such a machine would provide a basis for the study of human higher mental processes. A reading machine--a set of processes sufficient for understanding written language--would provide a basis for both studies.

In this paper, the fundamental aspects of the reading machine problem are discussed. An attempt is made to define meaning and understanding in useful terms. Means of organizing the storage of information are examined in terms of efficiency and power. Finally, a digital computer program is described in detail. This program implements the major features of the reading machine requirements by making some deductions from English text which, though trivial to adult humans, display the kernel of the problem.

To accomplish this task, guideposts from many fields of study are employed: Chapter One reviews the findings of experimental psychology in the area of higher mental processes. These results are particularly helpful as guides for efficient means of organizing complex systems. Chapter Two discusses recent results in the study of complex information processes and communication theory. These results are analyzed in terms of the problem under discussion, and some definitions and strategies are offered for carrying out the development of a reading machine.

Chapters Three and Four are detailed examinations of a computer program which analyzes English syntax and discovers implicit meanings along the lines of the earlier descriptions. It is felt that the exactness of specification required for the construction of this system is necessary for uncovering pertinent questions which do not arise from a more superficial discussion.

Chapter Five analyzes some of the questions raised by the computer system. The final topic is the application of these discussions to the psychological questions which were their source.

CHAPTER ONE

Introduction

The history of thought concerning thought is marked by many false starts and much argument about seemingly fundamental distinctions: insight versus trial-and-error, associationism versus even less well specified but more "complex" processes, and goal direction versus mechanical processing, to name a few. Each of these problems has consumed much space in the literature. Each has succumbed, not by being resolved, but by being viewed in alternate fashions.

Due to the complexity of the thinking processes, experimental investigations have been restricted to the study of simpler tasks which are assumed to be basic to the formation of the higher mental processes. As yet, empirical laws dealing with the most complex functions remain undiscovered; we have not sufficient evidence to begin to distinguish among various methods of accomplishing these complex functions. However, the few direct attacks upon problem-solving, attitudes, and concept formation have, along with some studies of more basic processes, yielded a few broad results which may be used as guides in constructing a theory of thinking. These results are sufficiently well established to provide a set of postulates which must be included in such a theory, but they are weak in two ways. First, they are vague, in the sense that they may enter the theory in a number of distinct formulations. Second, they are incomplete, in the sense that further postulates are required to specify a system whose behavior is comparable to that of the human mind. The strategy here adopted is to accept these results as guideposts in our theorizing. To that end, this chapter is devoted to the discussion of the empirical bases of six major psychological concepts.

The discussion is organized around the principle of limited rationality. According to this principle, a major psychological question is: Given the known, severe limitations on the information processing capacity of the human mind, by what strategies do humans perform the complex functions of which they are known to be capable? The experimental and theoretical literature is viewed in this light.

The concept of limited rationality

Many experiments and theories have dealt with the irrationality of people. Much of this emphasis on irrationality was, of course, sponsored by Freud and his colleagues, who focused upon unconscious motivation as the key to understanding human behavior, and looked upon the rational mind as a psychologically subordinate feature used to placate the conscience. Experimental psychologists, however, have chosen lack of rationality--even in objective, dispassionate, logical tasks--as a major point of concern. Woodworth and Sells (1935) studied the "atmosphere effect" and demonstrated that the wording of the premises of syllogisms affected the reasoning of their subjects, although the changes in wording did not affect the logical form. Luchins (1942) amply demonstrated how the order of presentation of problems could cause many subjects to fail to solve tasks of great simplicity; Maier (1930, 1931) investigated problem-solving activity in a similar light. Studies of prejudice, such as those of Allport and Postman (1945), Levine and Murphy (1943), and Bartlett (1932) have demonstrated how attitudes can distort the logically clear relationships in a stimulus situation, or work to selectively notice, eliminate, or forget other aspects. More recently, Abelson and Rosenberg (1958) have proposed a system of "psycho-logic" as explanatory of human reasoning. This system allows conclusions to be drawn when logically there is not sufficient evidence available. An example given by these authors is: India opposes U.S. Far Eastern Policy. U.S. Far Eastern policy is directed against Communism. Therefore, India is in favor of Communism. In addition to these postulated limits on rationality, Abelson and Rosenberg assert that "We hypothesize that these rules apply only when the individual thinks about the topic, or 'rehearses the arguments' [their emphasis]." The implication of this note is a second limit on human rationality: cognitive activity takes time, and, for humans, time limitations curtail ability.

None of these authors would assert that subjects deny that A implies A or that they disavow any other logical rule, if the statement is presented in the proper form and ample time and description are allowed. Why, then, do subjects persist in being so illogical? All of the answers cited above take the form of hypothesizing some limits on the ability of

humans to process information. These limits can take two forms: first, the persistence to "make sense of" or complete a conceptual system (a conceptual level Gestaltism), and second, the inability to handle more than a small quantity of information at one time. In other words, irrational behavior may result from lack of information and/or from lack of computing ability.

The concept of rationality deserves further consideration. Rationality is a property of the environment; that is, a theory of the perfectly rational man is a theory of how the world operates, and how an organism should behave to adapt perfectly to the world. For example, the move made in a chess game by a perfectly rational player is the optimum move dictated by the rules of chess; all processes which arrive at this result are equally rational. It would seem, if this analysis were true, that logic is something learned (by each individual or by the species, through evolution), as part of the adapting process. Work by Piaget (1931, 1932) with small children lends support to the view that each individual learns logical relationships through experience with his environment.

March and Simon (1958) have examined the concept of rationality as an aid to theorizing about human behavior. They point out that classical economic theory and statistical decision theory are based upon the assumption that people are perfectly rational in a very broad sense: they are completely aware of all data which are relevant to their choices, and they are possessed of unlimited powers of computation. These assumptions are quite opposed to psychological evidence, which may be sufficient to account for the failure of the resulting theories as predictors of human behavior.

An alternative to the usual emphasis upon man's irrationality is proposed by March and Simon. Rather than consider an irrational organism dealing with a rational world, they adopt the strategy of assuming that the human mind is limited in its powers and that due to these limits it operates in a perceived world which is not as complete as the external world. According to this view, learning, thinking, and problem-solving are seen as processes of adaptation. In the context of the limited information and limited ability which the organism brings to the problem situation, behavior is assumed to be rational. When seen in the larger, objective context, the behavior may appear irrational. Emphasis is thereby withdrawn from the question of whether or not the human mind is

rational. Consideration is shifted to determining what information is available to a subject in a problem situation, what specific thought processes the subject has at his command at the time, and what fundamental limits are imposed on the subject's capacities for developing other thought processes. Recent advances in formalizing the concept of information, principally due to Shannon (1948), and other work investigating complex processes, principally due to Newell, Shaw, and Simon (1957), have given these questions meaning which is absent in other formulations of the problem of human cognition. This view of the problems of cognition becomes a useful research strategy and provides a convenient framework for organizing our discussion.

The questions for which a theory of cognition must find answers, then, become the following sort: In what ways are humans limited in their capacity to process information; how do these limits affect the information considered; given such limitations, how are humans capable of complex reasoning? In the remainder of Chapter One the results of psychological investigation and theorizing will be broadly considered in the light of such questions. Several generalizations about the organization of human thought processes will be distilled from the literature for use in subsequent chapters.

Empirical findings concerning cognition

Limits on information processing capacity

Miller (1956) has summarized a large amount of experimental evidence concerning the so-called channel capacity of human subjects. For a wide range of stimulus information, including pitch (Pollack, 1952), loudness (Garner, 1953), taste (Beebe-Center, Rogers & O'Connell, 1955), and points on a line (Hake & Garner, 1951), the channel capacity, measured in terms of the number of alternative stimuli which subjects could distinguish, ranged from 4 to 10. These figures are strikingly low when compared to what could be achieved with present-day technology if an attempt were made to construct a good special purpose machine for any of these tasks.

Experiments on the span of immediate memory, also summarized by Miller, show a range of 5 to 9 items, items being randomly-presented monosyllabic English words, letters and digits, letters, decimal digits, and binary digits, in that order. Although the range of information for these various stimuli is very large, the number of alternatives is again quite small. A large computer could remember indefinitely, without error, hundreds of thousands of binary digits after a single presentation.

Here then we have a large amount of detailed evidence as to the very limited information processing capacity of the human. It seems that a human has the ability to handle less than 10 items of information at one time. This fact is often referred to as a limit on the size of immediate memory.

A second important limitation is of a similar nature, and perhaps in some aspects results from the first. We recognize that humans are also very slow in performing tasks, and unable to consider many alternatives at once. It takes an average subject about 10 trials, depending on conditions, to learn a list of 10 syllables (McGeoch, 1952). Even expert chess players consider in depth only 9 to 10 moves out of the possible set (de Groot, 1946), although they perhaps examine a larger set in order to select those for detailed examination. In summary, not only are humans limited to a small set of items for simultaneous consideration, but they require a relatively long time to make decisions about these items.

How then can a human ever accomplish anything, given these severe limitations on his computing abilities? Much of the literature of cognition might be viewed as a listing of attempts to answer this question.

Hierarchical organization and encoding

A number of studies may be organized around the above question by the postulation of a basic form of memory organization which allows a processing system of limited capacity to execute functions of great complexity. In addition, this form of memory organization accounts for many psychological phenomena, though only in broad terms. The limited psychological evidence available does not rigorously imply such an underlying organization; it is, however, a sufficient explanation for certain data.

A basic feature of the immediate memory is that its size is measured in items, not in information. That is, the system acts as though it will accept a small number of symbols, but the information content of a symbol may be large or small. Thus a subject can remember 8 decimal digits or 8 letters of the alphabet (Pollack, 1953b), although 8 randomly-selected digits amount to 26 bits of information, while 8 randomly-selected letters amount to 35 bits of information.

Given these constraints, it follows that a subject must recode several items of information into some sort of structure which can be stored elsewhere as a unit and which can be named in the sense that a single symbol can now serve as token for the entire group. Perhaps language provides the clearest illustrations of this process. Letters form words and words form ideas in such a fashion that some groups of letters ("meaningful" groups) and some groups of words (ideas) provide much less strain on the cognitive system than do other (meaningless) combinations of the same objective complexity. An interesting illustration pointing up the link between immediate memory and the concept of a name, is given by Humphry (1951) who reports that experienced telegraphers lag the telegraphic signal by as much as 200 dots and dashes, a surprisingly large number, unless it is remembered that this amounts to only 8 or 10 words.

A large body of experimentation has been devoted to the study of transfer of training: learning task A facilitates (or impairs) the learning of task B. Some cases of facilitation may be accounted for by postulating that some processes of use in the first task are also usable as portions of the second task; that is, these subprocesses act as a unit in the sense of being interchangeable.

A common experience observed when learning a new task, such as driving an automobile, is that certain actions require detailed practice. But eventually they become automatic, so that a single command, such as to avoid a hole in the road or turn a corner, can be executed without further explicit consideration of all of the complex error correction necessary. A point seems to be reached where a complex process operates as a unit, and this point frees the limited control system for other use--conscious effort is no longer required. The frequent result of concept

attainment studies, that subjects are able to attain concepts or solve problems--as demonstrated by their overt success--before they can verbalize their methods, may result from the use of previously encoded processes. This explanation places the subject in the position of a computer programmer who can use a general purpose program written long ago even though he no longer can recall the method which the program employs, and hence cannot explain how it solves the problem at hand. This phenomena occurs at very early ages (Roberts, 1932), which illustrates the feasibility of applying the encoding principle to the explanation of extremely large non-conscious deductions as exemplified by the mathematical intuition of trained mathematicians.

If the encoding phenomena results, as we have argued here, from the limited size of immediate memory, we should find that the human memory is organized in a hierarchical fashion, with a small number of branches (5-10) at each node. Mandelbrot (1955) has shown that such an assumption applied to a language coding tree leads to the statistical properties of natural languages as expressed by Zipf (1945). It should be noted further that even if the hierarchical structure were not required by the limited immediate memory, it would still be valuable and perhaps necessary due to the other cognitive limits. Thus the construction of a large system is brought about in a much more efficient fashion by constructing stable sub-assemblies than by attempting to cut it from whole cloth, when the task to be accomplished is large in comparison to the computing power and speed available or when error may enter the system. This situation is indeed exactly of the type which we are discussing: humans create conceptual systems of great complexity in comparison to their capacity. Even if a person were capable of making large sub-assemblies of his concepts, he would proceed at such a rate--judging from our knowledge of his ability at solving problems--that the probability of error or interruption destroying the construction before it became stable would be very large. In this regard it is relevant to note that an error occurring in one letter of a word does not destroy the whole message. Mandelbrot has shown that this situation requires either that all coded words be of the same length (which of course is not true of most natural languages) or that words end with a symbol (the space) which does not occur inside the word. That is, the observed quantizing or grouping of letters and words is required to guarantee a certain level of accuracy in communication.

A hierarchical system of storing information, as well as processes, may also be employed in the human brain, and this view is not in disagreement with psychological observations. Most experimental investigations of concept formation (Goldstein & Scheerer, 1941; Komachiya, 1957; Vinacke, 1951) view the problems presented to the subjects in an hierarchical paradigm. A concept can be considered to be a collection (or the name of a collection) of other concepts, each with some common element (e.g. Hull, 1920) or relationship (e.g. Smoke, 1932, 1935). Meaningful questions can be asked in this context as to the manner in which a collection can be searched for common properties. The problem of strategies of search, again in the light of limits on cognition as determinants of "cognitive strain," have been extensively investigated by Bruner, Goodenough, and Austin (1956). More recently yet, non-psychologists have been concerned with the problem of efficient storage of large quantities of information, such as found in libraries, scientific abstracts, and legal decisions. The hierarchical arrangement is, of course, an obvious consideration here.

Associative organizations

A third property of the human conceptual and mnemic system is generally discussed under the title of associationism. As far back as Aristotle reference can be found to the fact that human thoughts are stored not as one stores data in a computer, or cans on a shelf, but are stored in such a fashion that one thought leads to another. The mechanism is made flexible by the fact that one word or thought is connected with many others. Word association studies, such as those of Kent and Rosanoff (1910) and Jung (1918) demonstrate both the striking consistency and the striking flexibility of these associations. For example, fruit may call forth the association vegetable, food, apple, brute, or a number of other responses; but the connections do not seem at random. The associative nature of memory has been extensively investigated and measured by such techniques as conditioning (Cofer & Foley, 1942; Diven, 1937; Razran, 1935, 1939), sentence completions (Brower & Abt, 1952; Osterweil & Fisk, 1956), and free association (Jung, 1918).

A feature of associationism which is also important is its partial ordering property. Subjects find it much easier to recall a list of memorized words in the order presented than in any other order (see McGeoch, 1952). Backward associations also exist, but these are not mediated with such facility. That the association process is not symmetric is demonstrated by the difficulty of reciting the alphabet backwards. A non-ordered form of information storage, such as the standard computer arrangement, is equally facile in either direction.

Einstellung and heuristics

Luchins (1942) presented his subjects with a number of problems of the form: given three water jars of capacity a, b, and c, how does one obtain x units of water if the jars have no markings for intermediate measures? His results showed that if subjects were given several problems which could be solved by method A (such as fill b and empty it into a once, then into c twice) and then were given problems which could be solved by method A or method B, subjects would persist in the use of method A even though method B might be easier (such as fill a and empty it into c). Furthermore, additional problems which could be solved only by method B required much more time for solution than was the case for control groups. In some cases, the result was very striking: method B required only the filling of one of the jars. Luchins referred to this phenomena as Einstellung.

Other investigators have studied this phenomenon. Maier (1930) demonstrated that subjects had difficulty in using a tool in an unusual fashion--as opposed to the normal set--but could solve problems with it readily if this set were destroyed. In such cases a hint as to the proper direction (as Maier terms it) is a much more powerful aid than is practicing parts of the solution.

Set, or direction, or Einstellung, is another example of a phenomenon which may be thought of as resulting from limited rationality in the sense that it is a technique for avoiding complete crippling by these limits. A good example of this is given by Newell, Shaw, and Simon (1957) in their discussion of the Logic Theorist. The problem to which they addressed themselves was the programming of a digital computer to discover proofs to theorems of symbolic logic. A proof, as considered

in their work, consists of a sequence of logical rules selected from a given set (the axioms and previously proved theorems of symbolic logic). When applied in order to the premises of the theorem under consideration, this sequence yields the conclusion of the theorem. Newell, et al. demonstrated that the discovery of such proofs by exhaustive methods would involve examination of hundreds of thousands of continuations; this task would be too large even for a high-speed computer. One alternative, later demonstrated by Wang (1960), is to select a more efficient algorithm whose application is simple enough to be practical. Rather than take this approach, Newell, Shaw, and Simon studied methods of selection in the examination of possible solution paths, limiting exploration to those routes which have a high probability of yielding success. The techniques for making these selections they called heuristics.

Heuristics, then, are selection strategies which provide tremendous power for problem solvers of limited speed and memory capacities. But selection and persistence are the contributing factors to the seemingly irrational behavior demonstrated by Luchins, Maier, and others. The Luchins experiment may be viewed as a demonstration of humans' ability to develop heuristics, indeed over a very short period. The perverse situation contrived by the experimenter caused trouble for the subjects. But such is the property of selection: one runs the risk of being wrong.

The study of how heuristics are developed is of critical importance to a theory of cognition. Heuristics may assume the form of a priority schedule which determines which aspects of a stimulus situation are to be given first consideration. It is this form whose development has been most extensively studied experimentally. Heidbreder (1946, 1947) and Komachiya (1957) have investigated concept formation as a function of priorities in the consideration of stimulus dimensions and have found definite preferences common to most subjects. Along similar lines, Lindsay & Willis (1959) have gathered evidence concerning priorities given to stimulus dimensions in a complex discriminative reaction task. Here stimuli assumed one of two values for each of seven dimensions. Based on measures of reaction time for each dimension separately, the seven dimensions could be ranked according to difficulty. When required to examine all dimensions in order to make a decision, the subjects did so in order of difficulty, easier dimensions being examined earlier.

Completion or closure

We have already referred to the work of Abelson and Rosenberg as an example of a theory about cognitive activity in a context of incomplete information. Many experimental studies have demonstrated the fact that subjects tend to "make sense" of situations which do not in themselves hang together. Experiments on perception perhaps are more abundant in demonstrating this phenomenon.

Wertheimer (1958) has cited many examples of familiar perceptual situations as evidence for the Gestalt theory of perception. The existence of common stable perceptual organizations, resulting from a variety of complex stimulus patterns exemplifies the notion of stable cognitive organizations. Gibson (1929) has demonstrated one technique for the study of stable perceptual forms. His subjects were asked to make reproductions of geometric figures briefly (1-1/2 seconds) exposed. Reproductions tended toward "familiar" configurations, even when the actual stimuli departed from these forms. Many other studies (e.g. Bobbitt, 1942; Bruner, Postman, & Rodrigues, 1951; Mowatt, 1940) of this closure phenomena illustrate the concept of stable organization, or Gestalt.

The definition of what is closed and what is open is, of course, not an easy task. It must be conceded, however, that some cognitive organizations are stable in the sense of being more readily recalled, more easily manipulated, and likely to be achieved under a variety of stimulus conditions. Other organizations are more subject to forgetting, less readily applied in thinking, and very likely to be spontaneously converted into another organization. The specification of the properties of stable organizations is a prime goal of cognitive theory.

Many recent efforts toward the development of a theory of attitudes (Cartwright & Harary, 1956; Heider, 1946; Osgood & Tannenbaum, 1955) have been primarily concerned with structural balance or stability in the organization of evaluative concepts. Osgood and Tannenbaum (1955) treat certain basic attitudes as dimensions in a factor analytic approach to the study of concepts. A key point in their theorizing is the observed "polarization" of attitudes: evaluations tend to be bi-modal rather than uniform. Theories of attitude change discuss the effects of additional

information upon the existing balance, the most interesting instances being those which involve "incongruity." This occurs when both positively and negatively valued factors enter into a single situation, as when a popular person endorses an unpopular idea. In such cases, the cognitive organization appears to be disrupted.

Summary

Psychological investigation has yielded little in the way of precise measures of cognitive activity, but several factors are well-established properties of cognitive organization.

1. Humans are limited in their ability to process information by an immediate memory system of small capacity (as measured by the number of items which it may hold).

2. Hierarchical organization predominates in the human cognitive system.

3. Memory is associative in nature, and these associations are subject to a partial ordering.

4. Humans exhibit the property that, having discovered a successful method, there is a high probability that they will pursue this method in similar situations.

5. Complex tasks are not attacked exhaustively, but strategies are employed to limit the set of alternatives considered.

6. Cognitive organization favors certain (stable) states over other (unstable) states.

Only in isolated situations are there data sufficiently detailed for precise theorizing. However, in the remaining chapters, we will use the six points discussed above as the basis for a theory of meaning and understanding. Although this theory must await detailed empirical study for verification, it is constructed to agree with the broad facts of cognitive psychology.

CHAPTER TWO

Introduction

Much effort has recently been devoted to the construction and analysis of complex machines whose behavior may be called intelligent. In this chapter certain features of memory organization and information processing are discussed in the light of the psychological evidence presented in the preceding chapter. An examination is made of the problem of developing a machine which can read and understand English text, and definitions are given for meaning and understanding. Finally, a description is given of a type of memory organization whose properties allow efficient use of storage space and provide direct methods for storing implications of the passage read.

Complex information processing

The study of complex systems has been stimulated by the advent of high-speed digital computers. Much of the effort devoted to these large capacity information processing systems has been aimed at developing efficient methods of instructing the machine. These efforts usually take the form of computer programs designed to translate into awkward, detailed machine language some pseudo-language which is of greater convenience to the human operator. Most higher order languages are oriented toward numerical applications, for which present computers were primarily designed and for which they are most directly adaptable.

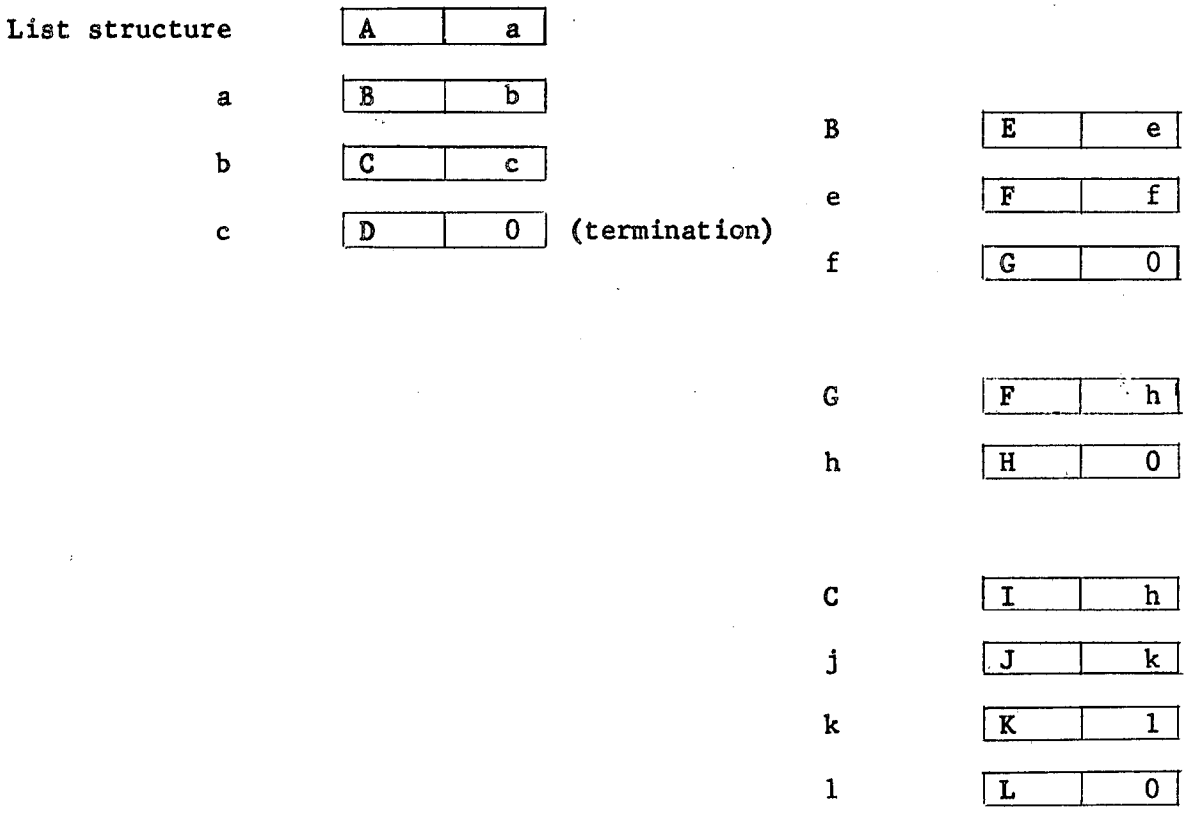
An increasing number of investigators have concerned themselves with intelligent behavior. Most studies can be classified as dealing with information retrieval, language translation, pattern recognition, or problem-solving. These four areas include a vast collection of highly complex problems of a difficulty probably comparable to that which characterizes human endeavors. These problems are basically of a non-numerical nature, but all involve the manipulation of large quantities of symbols.

The problem of instructing computers to handle such tasks is complicated not only by lack of knowledge as to how these tasks should be done, but also by lack of means of instructing the machines to behave in the complex ways desired.

Newell, Shaw, and Simon (1957) showed that computer memory organized into lists of associated cells provided great flexibility and freedom in handling complex systems, as compared with the more usual organization of computer memories wherein each item of information must be referred to by its specific numerical address. In the Newell-Shaw-Simon list structure memory organization, each computer memory location is divided into two parts: one part contains a symbol which is the main concern of the program, and the second part contains the name of the location of an associated memory cell, which in turn names a third cell, and so forth. Such a string of associated symbols is called a list. (See Figure I.) An important feature is that any item on any list may be the name of another list, thus achieving a tree or hierarchical structure. Whereas with the addressing system, symbol locations must be either given or (numerically) computed, with the list-structure memory organization only a starting point need be found to locate a symbol. The exact location in memory generally is not known at all, and the intervening structure need not be specified.

The list structure organization is the basic feature of information processing languages, as characterized by IPL-V (Newell, Tonge, Feigenbaum, Mealy, Saber, Green, & Wolf, 1960). The basic instruction set of these languages deals with operations upon lists or single symbols. Some examples are: erase a list structure (locates and erases all of the cells connecting a large set of symbols, no matter what particular form the organization takes);add a symbol to a given list; compare two symbols; combine two lists. Additional refinements, differing from one language to another, have been incorporated. For details, the IPL-V manual should be consulted; we are here concerned only with the interesting properties of list structures.

Figure I



Example of List Structure Memory Organization

We note that list structures form a basically hierarchical, associative, and partially-ordered organization. Use of a list structure language, therefore, not only provides great convenience, but also incorporates into a program some features which are present in the human cognitive system. This does not, of course, imply that there is a direct correspondence in the central nervous system to the associated memory locations of a list structure computer.

The theories presented in this paper are specified as IPL-V programs, and are put forth in the light of the advantages outlined above.

Scope of the theories investigated

Verbal behavior and language are fertile areas for the study of cognitive processes. It is with words that man communicates his concepts, explains his actions, and to a large extent develops his cognitions. Philosophers may still wonder if all of our thinking is done with words, but they will admit that language holds a special place in cognition.

The present theorizing concerns the processes of reading and understanding passages of English text. Degree of understanding may be operationally defined as the degree of success in answering questions concerned with material which is contained in or may be deduced from the passage. The processes necessary for answering questions may cover a wide range of complexity, including the most abstract thinking of which man is capable. In this initial paper the focus is upon some of the processes and cognitive organizations involved in the performance of even simple instances, in the belief that the most complex examples are likely to make use of these same features. The essence of the problem is the ability of humans to input passages of English and store the information in such a fashion that material which was implicit in the text is now available as well as the explicitly cited material.

Some examples will aid in clarifying our goal and illustrate those facets which are of direct concern. If a human subject reads that John is Mary's brother and that Jeanne is the sister of Mary, he will then be able to conclude that John is Jeanne's brother, that Jeanne is

John's sister, that Mary is John's brother or sister, that Mary is Jeanne's brother or sister, that Jeanne is female and that John is male. None of this information can be obtained by simple examination of the input phrases.

A more complex example is taken from a College Entrance Examination Board test in reading comprehension, reproduced as Figure II. This illustrates the type of questions one might reasonably expect an intelligent high school student to be able to answer. Again the questions do not merely require as answers material which can be obtained directly from the text. They require the making of deductions from the text material, and they may be more readily answered by a subject with some previous knowledge of English history. These two facets are not distinct--both require some sort of history of contact with other sources of information. By the former is meant the use of fairly general processes of deduction which would be applicable to other materials (such as those processes which would enable the reader to deduce from the "tone" of the passage the answer to question four). By the latter is meant the more detailed knowledge obtained from direct experience or from contact with other passages on the same topic (such as the recognition of Cromwell as a former ruler of England). We wish to maintain this distinction, calling the former deductive processes and the latter history.

In summary, our area of concern is the study of those cognitive organizations and deductive processes which enable human subjects to obtain information implicit in a passage of simple prose and which require a minimum of history or familiarity with the topic.

Figure II

Yet, though we believe that the intentions of Cromwell were at first honest, though we believe that he was driven from the noble course which he had marked out for himself by the almost irresistible force of circumstances, though we admire, in common with men of all parties, the ability and energy of his splendid administration, we are not pleading for arbitrary and lawless power, even in his hands. We know that a good constitution is infinitely better than the best despot. But we suggest that, at the time of which we speak, the violence of religious and political enmities rendered a stable and happy settlement next to impossible. The choice lay, not between Cromwell and liberty, but between Cromwell and the Stuarts. That Milton chose well no man can doubt who fairly compares the events of the Protectorate with those of the thirty years which succeeded it, the darkest and most disgraceful of the English annals. Cromwell was evidently laying, though in an irregular manner, the foundations of an admirable system. Never before had religious liberty and the freedom of discussion been enjoyed in a greater degree. Never had the national honor been better upheld abroad, or the seat of justice better filled at home. And it was rarely that any opposition that stopped short of open rebellion provoked the resentment of the liberal and magnanimous usurper. The institutions which he had established were excellent. His practice, it is true, too often departed from the theory of these institutions. But had he lived a few years longer, it is probable that his institutions would have survived him, and that his arbitrary practice would have died with him.

1. The form of government under Cromwell is best described as a
....(A) tyranny
....(B) democracy
....(C) benevolent dictatorship
....(D) constitutional monarchy
....(E) limited aristocracy

2. Milton chose
....(A) the Stuarts
....(B) liberty
....(C) Cromwell
....(D) to remain neutral
....(E) the best possible form of government

3. The Protectorate
....(A) ushered in a thirty-year period of comparative freedom and political stability
....(B) was preceded by a period of unequaled misrule
....(C) represents the lowest ebb of England's political history
....(D) was constantly threatened by rebellion
....(E) was a period during which England merited considerable prestige

4. The author's attitude toward Cromwell's regime is one of
....(A) enthusiastic endorsement
....(B) grudging approval
....(C) noncommittal appraisal
....(D) emphatic distaste
....(E) qualified admiration

5. Which of the following statements best represents Cromwell's approach to government?
....(A) Might is right.
....(B) In union there is strength.
....(C) After me, the deluge.
....(D) The end justifies the means.
....(E) Whatever is is right.

Meaning and understanding

In order to better define the problem and clarify the operation of the computer program to be described later, we will offer definitions for "meaning" and "understanding." Without distortion of intuitive notions, we may say that the term meaning refers to the information content of a message, and that the term understanding refers to the decoding of the message. That is, meaning is a property and understanding is a process. We will now put these interpretations into more exact form.

The basic paradigm of a communication situation consists of the following five parts:

1. A context composed of a set of possible source alternatives. Communication is the process of transmitting the truth-value of these alternatives from one intelligence to another or from one time to another.

2. An encoding scheme which assigns to each alternative a message.

3. A set of symbols, strings of which compose the messages, and an associated communication channel which is capable of transmitting the symbols.

4. A decoding scheme which assigns to each message an alternative selected from:

5. A set of target alternatives. Usually the set of target alternatives is considered to be identical with the set of source alternatives. Unless explicitly noted, we will assume this to be the case and refer to members of this common set as alternatives.

In discussing a communication situation we must distinguish at least six sets of probabilities:

(i) The probabilities $p^*_i(a_i) = p^*_i$ that alternative a_i is true.

(ii) The probabilities $p_i(m_i) = p_i$ that message m_i will be sent.

(iii) The probabilities $p^*_{ij}(a_i | m_j) = p^*_{ij}$ that a_i is true given that m_j is sent.

(iv) The probabilities $p_{ij}(m_i | m_j) = p_{ij}$ that message m_i was sent given that message m_j was received.

(v) The probabilities $P_{ij}(m_i \mid m_j) = P_{ij}$ that message m_i will follow the transmission of the ordered set of messages m_j.

(vi) The probabilities $P^*_{ij}(a_i \mid a_j) = P^*_{ij}$ that a_i is true given that a_j is true.

The p^*_{ij} refer to the reliability of the encoding system. We will consider the encoding system to be errorless, and assume that message m_i is always assigned to alternative a_i. Thus, $p^*_{ii} = 1$.

The p_{ij} refer to the reliability of the communication channel. We will consider the "noiseless" case where $p_{ij} = 0$ or 1 for all i, j.

The P_{ij} define the statistical properties of the encoding system. In the following discussion we will consider only statistically unrestricted codes where $P_{ij} = p_i$.

A fundamental result of communication theory as presented by Shannon (1948) specifies the size of the symbol set and the average message length required for communicating about a context of N alternatives. Let C_m be called the <u>capability</u> of the communication system. (The term <u>capacity</u> is generally reserved to denote the time rate at which messages may be sent through the channel.) If there are n symbols in the symbol set and these are used to define messages of length M by concatenation of the symbols with equal probability, then:

$$C_M = M \log(n). \qquad \text{(All logarithms are to the base 2.)}$$

If we define the <u>information content</u> of a set of N alternatives as:

$$C = -\sum_{i=1}^{N} p^*_i \log(p^*_i),$$

then a communication system of capability $C_M \geq C$ is required to unambiguously communicate the truth values of the alternatives.

In the typical situation discussed in the literature, the alternatives are assumed to be mutually exclusive, which implies that $\sum p^*_i = 1$. It follows that $P^*_{ij} = 0$ for all $i \neq j$. It is further assumed that $p_i = p^*_i$, so that if a_i is true, m_i will be sent. In the remainder we will consider cases where these restrictions do not hold. This amounts

to allowing situations where communication of message m_i, while insuring the truth of a_i, does not rule out the possibility that some other alternatives are also true. In fact the communication of m_i may increase (or decrease) the probability that other alternatives are true; that is, it alters the subsequent communication situation.

Consider a set of alternatives $a_1, a_2, \ldots, a_N$ with a priori probabilities $p_1^*, p_2^*, \ldots, p_N^*$. Let the set of messages $m_1, m_2, \ldots, m_N$ correspond to these alternatives so that $p_{ii}(m_i \mid m_i) = 1$ and $p_{ii}^*(a_i \mid m_i) = 1$. Further, allow the p_i to be altered by the communication of a message; that is, after m_j is sent, p_j becomes 1 and the p_i^* are replaced by the P_{ij}^* which are not necessarily equal to the p_i^*. We wish to reserve the term meaning for situations where the communication of a message alters the system in this fashion.

Definition: A message m_j has meaning if and only if $P_{ij}^* \neq p_i^*$ and $p_i^* \neq 0$ or 1 for some i.

Definition: A message m_j has definite meaning if and only if $P_{ij}^* = 0$ or 1 and $p_i^* \neq 0$ or 1 for some i.

Definition: A message has connotative meaning if and only if $P_{ij}^* \neq 0$ or 1 and $P_{ij}^* \neq p_i^*$ for some i.

It follows that a message may have definite meaning, or connotative, or both definite and connotative meanings, or no meaning.

Let C be the information content of the set of alternatives A: $a_1, a_2, \ldots, a_N$, and let C_i be the information content of the set A excluding a_i, measured in terms of the a priori probabilities p_j^*. If we call $C_{|m_i}$ the information content of the set A, after message m_j is received, we may prove the following result.

Theorem: If a message m_i has definite meaning but does not have connotative meaning, $C_{|m_i} < C_i$.

Proof: $C_i = -\sum p_k^* \log(p_k^*)$ with summation over all members of A except for $k = i$.

$C_{m_i} = -\sum P_{ki}^* \log(P_{ki}^*)$ with summation over all members of A except for $k = i$.

Since m_i has definite meaning but not connotative meaning, for all k either $P^*_{ki} = 0$ or 1 or p^*_k. Each term in C_i equals the corresponding term in $C_{|m_i}$ when $P^*_{ki} = p^*_k$. Where $P^*_{ki} = 0$ or 1, the corresponding p^*_k are in the range $0 < p^*_k < 1$, since by the definition of definite meaning $p_k \neq 0$ or 1. Hence each term $-p_k \log (p_k)$ in C_i is positive. Each corresponding term in $C_{|m_i}$ is 0, since $P^*_{ki} = 0$ or 1. Hence $C_{|m_i} < C_i$. Q.E.D.

It follows that the communication of a message with definite but not connotative meaning reduces the requirements on the symbol set and communication channel so that fewer symbols or shorter messages may suffice for future communication. Or, if messages (and encoding-decoding schemes) are not altered, the messages are more redundant so that more noise may be tolerated in the channel without increasing the error rate.

To illustrate the foregoing ideas, consider the following example. Suppose all men are named either Bob, George, Sam, Pete, or Roger with equal probability 1/5, and that all women are named either Peggy, Alice, Mary, or Jane with equal probability 1/4. Consider the set of alternatives identified by their corresponding messages given in the following table.

alternative	message no.	a priori probability	probability after (1)	probability after (11)
X is a man	(1)	1/2	1	0
X is a woman	(2)	1/2	0	1
X is Bob	(3)	1/10	1/5	0
X is George	(4)	1/10	1/5	0
X is Sam	(5)	1/10	1/5	0
X is Pete	(6)	1/10	1/5	0
X is Roger	(7)	1/10	1/5	0
X is Peggy	(8)	1/8	0	0
X is Alice	(9)	1/8	0	0
X is Mary	(10)	1/8	0	0
X is Jane	(11)	1/8	0	1

Message (1) is said to have definite meaning since the a priori probabilities of (2), (8), (9), (10), and (11) are not 0 or 1 and the probability of (2) after (1) is received is 0. Message (1) is also said to have connotative meaning since sending message (1) yields probabilities for (3), (4), (5), (6), and (7) which differ from their a priori values but which do not equal 0 or 1.

Message (11) has definite meaning but not connotative meaning since no a priori probabilities are 0 or 1, yet all are altered to 0 or 1. After receiving message (11) the system is fixed, hence no further messages are required to communicate the truth value of all alternatives. This is an extreme case where the meaning of a message eliminates the need for further communication.

As a second example, assume the following probabilities for heights of women and men:

	under 5 feet	5 to 6 feet	over 6 feet
women	0	1	0
men	1/3	1/3	1/3
sex unknown	1/6	2/3	1/6

Assume further the following probabilities for weights of women and men:

	under 100 lbs.	100 to 200 lbs.	over 200 lbs.
women	1/6	2/3	1/6
men	1/3	1/3	1/3
sex unknown	1/4	1/2	1/4

Consider the following alternatives and messages:

alternative	message no.	a priori probability	probability after (1)
X is a man	(1)	1/2	1
X is a woman	(2)	1/2	0
X is under 5 feet	(3)	1/6	1/3
X is between 5 and 6 feet	(4)	2/3	1/3
X is over 6 feet	(5)	1/6	1/3
X is under 100 lbs.	(6)	1/6	1/3
X is between 100 and 200 lbs.	(7)	2/3	1/3
X is over 200 lbs.	(8)	1/6	1/3

The information content of the messages excluding message (1) is

$$C_1 = -(1/2)\ \log\ (1/2) - (4/6)\ \log\ (1/6) - (4/3\ \log\ (2/3)$$
$$= (1/2)\ \log\ 2 + (2/3)\ \log\ 5 + (4/3)\ \log\ 1.5$$
$$= 3.0$$

The information content of the messages excluding message (1) after message (1) is received is

$$C_{|m_1} = 6\ (1/3)\ \log\ (1/3)$$
$$= 2\ \log\ 3$$
$$= 3.16$$

Message (1) has definite meaning, since $P^*_{21} = 0$; but message (1) also has connotative meaning, since $P^*_{31} \neq p^*_3$ and $P^*_{31} \neq 0$ or 1 (and similarly for the remaining messages). In this case we see that it is possible that

$$C_{|m_i} > C_i$$

Definition: A set of messages[m] consisting of two or more messages has implicit meaning if and only if

(i) $P^*_{ij}(a_i|a_j) \neq 0$ or 1 for any a_j with corresponding message m_j belong to [m]; and

(ii) The probability of some alternatives a_i given all members of [m] is 0 or 1.

Since a set of messages having implicit meaning has definite meaning when considered as a unit, we have the following result.

Corollary: If a set of messages m has implicit meaning but not connotative meaning, $C_{|[m]} < C_{[m]}$.

Single messages having definite meaning will also be referred to as having explicit meaning.

The concept of a model

In the preceding section we have considered the addition of "memory" to the communication paradigm, by allowing the transmission of a message to alter the probabilities of the alternatives for subsequent communications. This extension allows us to treat information storage as

a type of communication, where the stored information takes the place of the symbols, messages, and communication channel. Shannon has specified the conditions under which a set of messages and a communication channel are sufficient to insure the existence of an encoding and a decoding scheme which are capable of communicating about a set of alternatives with a given probability distribution. When information storage is viewed as a communication from one time to a later time, Shannon's result specifies the properties of the storage system which allow a factorization of the problem into the following five parts:

(1) A set of input expressions consisting of truth statements for a set of alternatives.

(2) An annexing process which adds an alternative (more precisely, a symbol representing the truth of the alternative) to the storage system.

(3) A map which consists of the storage system itself. It is composed of a set of symbols which represent the truth of the alternatives and storage places for holding symbols.

(4) A retrieving process which may examine the map and determine the truth of alternatives which have previously been stored.

(5) A set of output expressions consisting of truth statements for a set of alternatives. These are usually considered to be identical with the input expressions.

We will refer to the storage map, together with its associated annexing process and retrieving process, as a model.

The essence of understanding is in the above factorization, specifically with regard to the independence of the parts. A model is called unrestricted when

(1) The annexing process is independent of the map; that is, any expression with probability not equal to 0 must be acceptable to the annexing process, regardless of which statements have preceded it.

(2) The retrieving process is independent of the map; that is, it is not altered for different maps, but remains the same after the operation of the annexing process (after the presentation of any expression).

It should be noted that the results of the previous section do not establish the uniqueness of unrestricted models as methods of information storage. Shannon's results, however, show that if a map has a sufficiently large symbol set and enough storage space, an annexing process and a retrieval process can be devised for a set of input expressions.

Consider the following example. The set of input expressions (= the set of output expressions) consists of (1) ab ; (2) ac ; (3) ba ; (4) bc ; (5) ca ; (6) cb. If $P^*_{ij} = 1$ for all i, j and $p^*_i = 1/6$ for all i, messages corresponding to each of these expressions have explicit meaning. Define an annexing process as follows: given ab, ac, ba, bc, ca, or cb, store the symbol # in the map, unless the map already contains # , in which case do nothing. Define the retrieving process as follows: if the map contains # , output ab, ac, ba, bc, ca, and cb; otherwise, no output. In this example, a model consisting of these two processes plus a map consisting of a single symbol # and one storage place is a sufficient model for storing the information about the input expression. Both annexing process and retrieving process are independent of the contents of the map and the order of presentation of the alternatives.

Phillips (1960) has demonstrated a system of the above sort with input expressions consisting of sentences of English. Having been given an input sentence, his program will answer questions about information contained explicitly in the input sentence, but allows the question to differ from the input statement in form. That is, the program recognizes the equivalence of the forms.

We will now consider the case of implicit meaning as previously defined. We have shown that when a set of alternatives has non-zero conditional probabilities among some of the members, it is possible to find an encoding-decoding scheme which reduces the required capability of the channel after a message with definite meaning is sent, or after a set with implicit meaning is sent. If the information is to be stored, it is possible to find a model of limited capability which is sufficient for handling input expressions with definite meaning. After presentation of all members of a set of expressions having implicit meaning, the requirements on the model are reduced. However, storing less than a complete set of implicitly meaningful input expressions may not reduce the required

capabilities of the map at all. The model can be altered to realize the savings only when the final member of an implicitly meaningful set is received.

As an example of this case, consider again the set of input expressions (1) ab ; (2) ac ; (3) ba ; (4) bc ; (5) ca ; (6) cb. If the input expressions were meaningless, six storage places would be required for storing communications about this set. We have seen above that only one storage place is required when each expression definitely means the others. We now will assign implicit meaning to subsets of these expression so that the following implications result:

(1) & (4) $\longrightarrow$ (2) (Interpret ab as a<b, etc.)
(2) & (6) $\longrightarrow$ (1)
(3) & (2) $\longrightarrow$ (4)
(4) & (5) $\longrightarrow$ (3)
(5) & (1) $\longrightarrow$ (6)
(6) & (3) $\longrightarrow$ (5)

If expression (1) is given, only the following sets are permissible: (4), (2); (2), (6); (5), (6). Similarly, given any of the other expressions, only three sets are permissible. Thus only three storage places are required for these expressions under the conditions of meaning specified. If, now, (1) is followed by (4), then only (2) is permissible, hence no additional space is required for storage. In such cases, two storage places suffice. But if (1) is followed by (2), either (4) or (6) is permissible, so three storage places are required. When expressions are presented in an order which minimizes storage requirements (i.e., when no messages are redundant), we will speak of well-behaved communications.

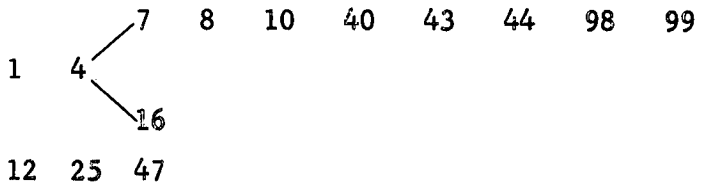
The programs to be described deal with the problems of constructing information storage systems--models--which are unrestricted, but which also allow a savings in storage capability when communications are well-behaved. In other words, the model must make efficient use of storage space, while allowing expressions to be presented in any order without altering the annexing and retrieving processes.

A large class of situations involving implicit meaning is composed of expressions about relationships. The expressions specify that one item bears a certain relationship to another item. Given the relationships among a group of items, the properties of the relationships may imply the manner in which other items are related. We may use the items (or names for them) as the symbols of a storage map, and store the implications by associating the storage places in efficient ways.

As an example, consider the relationship "greater than." One model of the relationship is a geometric one, the involved items being numbers. The map for this model consists of the spatial orientation of the numbers along a line: 1 4 7 8 10 40 43 44 98 99. An annexing process might say: If the new number x is given to be greater than a current resident of the map, and another current resident of the map is given to be greater than x, place x to the right of the first number and to the left of the second number. Thus if we were given "17 is greater than 10 and 40 is greater than 17," the map would be altered to: 1 4 7 8 10 17 40 43 44 98 99. The retrieving process for this model would be simply: If x is to the right of y, then x is greater than y. The retrieving process is unaltered by the use of the annexing process, the altered map now holds the new information, and the combination of map and the retrieving process makes available all deductions implied by the statement which introduced 17: 17 is greater than 1, 4, 7, 8 and 10; 40, 43, 44, 98 and 99 are greater than 17.

If we are to permit as input expressions all statements of the form $x > y$, the above model must be altered if it is to be unrestricted, since the expressions are now limited to certain orders of presentation. Thus if we were given "16 is greater than 4" we could not find a unique place for it on the straight-line map as it now stands. If we were given "47 is greater than 25 and 25 is greater than 12" it would be impossible to relate the newly given numbers to any of the previous map contents. A sufficient model for this example would require (1) provision for keeping more than one straight-line map in the storage system, and (2) provision for connecting two straight-line maps when expressions relating them are received. One scheme for doing this is a two-dimensional array consisting of straight-line maps placed above and below one another, and positioned from left to right according to given inter-relationships between maps.

Continuing with the example, as modified by the expressions given in this paragraph, we would have



The terms "to the right of" and "to the left of" as occurring in the annexing and retrieving processes must also be generalized.

There are many models for a given relationship. One question which we might ask about them is: how many words of computer memory are needed per item stored? A general model for any relationship could be constructed as follows. Construct a matrix of all items, so that the entry for row x and column y will give the relationship of x to y, the entry for row y and column x will give the relationship of y to x, etc. The retrieving process is very simple in this primary model: to find the relationship of x to y, locate the entry in row x and column y. The annexing process is not always so simple. If no implications arise from the addition of new information, the annexing process is trivial. If, however, the implications are many and complex, the annexing process also becomes complex. We might say in such cases that the map is not doing its share of the work. It is then necessary to devise maps which are more clever, and to do so without unduly complicating the retrieving process.

Models and stable cognitive organizations

As outlined in Chapter One, psychological theorizing may be viewed as the discovery of those processes which enable an organism of limited cognitive power, operating in an environment often of insufficient information, to behave rationally. We have noted that cognitive organizations are often functionally equivalent to the list structure format, as evidenced by the hierarchical, associative, and partially ordered arrangement of "thoughts." We are also aware of the limited immediate storage facilities of the human intellect. These factors, combined with the

realization that sentences are read word by word, with little reviewing (Dingman, 1958), force the conclusion that stable suborganizations are built up as the serial process progresses, in order that the sentence may be efficiently encoded to allow for known reading rates (Laycock, 1958). These stable suborganizations, also found to be of importance in larger contexts (see above) are fundamental to the discussion. In particular we are concerned with the general properties of these organizations (those things which make them stable and useful) and their form in particular contexts. We have chosen the general term model to refer to these structures, and we will subsequently examine models for syntactic analysis and an example of an hypothesized model for semantic analysis. Just as the sorting net is a fundamental organization for perception in the information processing theory of verbal learning (Feigenbaum, 1959), so the model is the fundamental memory organization.

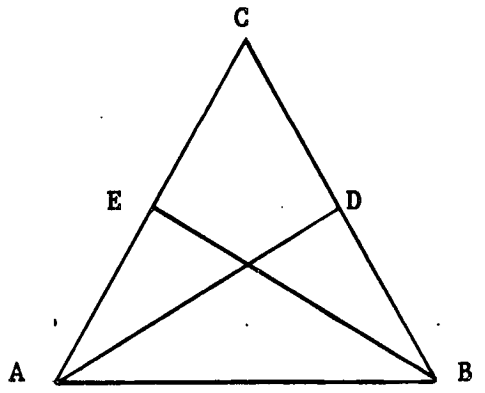
The concept evolves from considerations of the following sort. There are many ways of saying the same thing that are recognized as equivalent by humans. We consider comprehension to be the mapping of these diverse forms of expression onto a common organization. These common organizations are familiar from introspection as well as from the lay vocabulary: they are the mental pictures so commonly discussed. They display not only isolated concept, but the relationships between these concepts. And it is these relationships which supply the implicit information which we seek.

Geometric figures provide an important illustration of the model concept. Figures underlie reasoning about spatial relations by providing a picture, or storage system, which holds relationships of objects in space and conveniently makes deductions from given relations. Gelernter (1959) has used such a model as the basis of his plane geometry theorem proving program. Given the premises of a theorem, the program constructs a diagram appropriate to the situation; that is, all points mentioned in the premises are assigned Cartesian co-ordinates selected so that the figure which they define satisfies the conditions of the premises. The diagram thus constructed also illustrates relationships which were not explicitly given by the premises but were implied by

combinations of them. For example, consider the premises

segment AE = segment EG
segment BD = segment DC
angle ABE = angle EBC
angle BAD = angle DAC

A diagram satisfying the conditions of the premises is



This diagram also has segment AC = segment BC and angle ABC = angle BAC, although these conditions were not explicitly part of the requirements of the premises. The proof that these conditions do in fact follow from the premises is one of the most involved proofs given by Euclid.

Checking all possible sequences of axioms and theorems is far too large a task to yield valid proofs in reasonable time periods. Gelernter's program makes use of properties of diagrams in choosing sequences for testing. He reports that the program performs about as well as a high school student if it uses only the heuristic: "attempt to prove only those subgoals which are true in the diagram."

SAD SAM

The theory of understanding is specified as an IPL-V computer program. The problem of reading and understanding is usually divided into two parts: the syntactic problems and the semantic problems. This division is decidedly artificial. People do not learn these functions separately, nor do they perform their feats of language behavior in this two step fashion. Rather, they learn to obtain information from spoken and written language while they are learning to recognize correct forms for the presentation of the information; and further, understanding aids in the recognition of form, while formal structure is necessary for comprehension. Put differently, the existence of accepted forms, more or less standard, enables a person to obtain meaning from sequences of words; and the existence of external situations, more or less meaningful, enables a person to discover and recognize equivalent forms.

Nonetheless, it is convenient to factor the problem into these two parts as a starting point. Even though we are side-stepping some of the more subtle problems of interaction, at worst we may still view the two aspects as separate tasks, amenable to the problem solving powers of humans. Thus, syntactic information is often sufficient of itself for a unique syntactic analysis, and subjects may be studied when given only part-of-speech information. Similarly, grammatically incomplete passages -- even single words -- contain information which a subject may extract; this task too is open to experimental investigation.

First, we will consider the organization of words into their intended syntactic relations -- diagramming or constructing a syntactic model of the sentences. Second, we will consider the extraction and storing of semantic meaning of sentences dealing with a single, restricted topic -- constructing a semantic model of the situation discussed in the passage. The program for the first task will be called the Sentence Appraiser and Diagrammer (SAD), that for the latter will be called the Semantic Analyzing Machine (SAM).

As has just been pointed out, the learning of the reading processes comes about by the inter-relation of the syntactic and semantic aspects. In the performance of the reading function, the distinction is less clear, mainly because the analysis proceeds sequentially and both functions are carried out together. It is only after a good deal of learning has taken place that students can be taught the distinction and made to concentrate on one aspect or the other. But in reading a sentence, a striking feature is the sequential analysis; the relations of the words are discovered and the information which they convey is used to model the situation as the words are perceived, before the entire sentence is completed. The process is flexible, tentative hypotheses -- both as to which function, e.g., noun or adjective, and as to which sense a perceived word should take -- are constantly formed. They are accepted or rejected on the basis of the words which follow. It is this inter-relationship in performance which this first program misses.

Syntactic relations

Chomsky (1956) has studied the complexity of English grammar. He has shown that no finite-state Markov process that produces symbols with transition from state to state can serve as an English grammar, and further, that such processes that produce n-order statistical approximations to English do not come closer, with increasing n, to matching the output of an English grammar. Chomsky suggests a phrase structure analysis which resolves sentences into clauses, clauses into noun phrases and verb phrases, and so on until the function of each word is determined. This type of analysis is the basis of a syntactic analysis routine by Harris (1959). Harris's program scans the sentence in its entirety to locate phrases, and then analyzes these. Sager (1960) has devised a modification of this process which preserves the left-to-right scanning inherent in the procedure of humans. These procedures are fairly successful in yielding the main features of sentence structure when applied to scientific writing.

The syntactic portion of the present program is most similar to Sager's technique, in that it uses a left-to-right analysis. In addition to using a restricted vocabulary, it also uses a limited syntactic description

for each word, whereas the programs of Sager and Harris approach the complexity of categorization devised by modern linquists. The present program attempts to make a complete analysis of a sentence relating each word to every other, while using only limited syntactic information. The aim here is to devise a program which behaves in the way humans behave, rather than solve the parsing problem by using the computer's speed and accurate memory.

One word or group of words may be related in various ways to another word or group of words. A noun and verb may be, for example, subject and predicate, or subject and object; two nouns may be referent and object, or subject and object; two adverbs may modify the same verb, or one may modify the other. Furthermore, sets of relationships may imply further connections, as when noun a is the referent of noun b which in turn is the referent of noun c. We wish to be able to store the words of a sentence in such a manner that connections, such as between a and c, are contained in the arrangement. This is readily accomplished by using a list format which will be described in detail in the next chapter.

The task of parsing English sentences is complicated because the set of input expressions -- grammatically-correct sentences -- is infinite, and because a complete set of grammar rules has not been devised. The sentences themselves do not directly state syntactic relationships. This information must first be extracted from the sequence words. This is done by assigning to each word a set of syntactic functions, only one of which may be used at each instance, but any of which may be used by the given word from time to time. The job of the annexing process is to determine which syntactic function each word assumes in each instance, and to determine the relations between words given the limitations imposed by usage. In this portion of the program, the annexing process bears the greatest burden.

Family relations

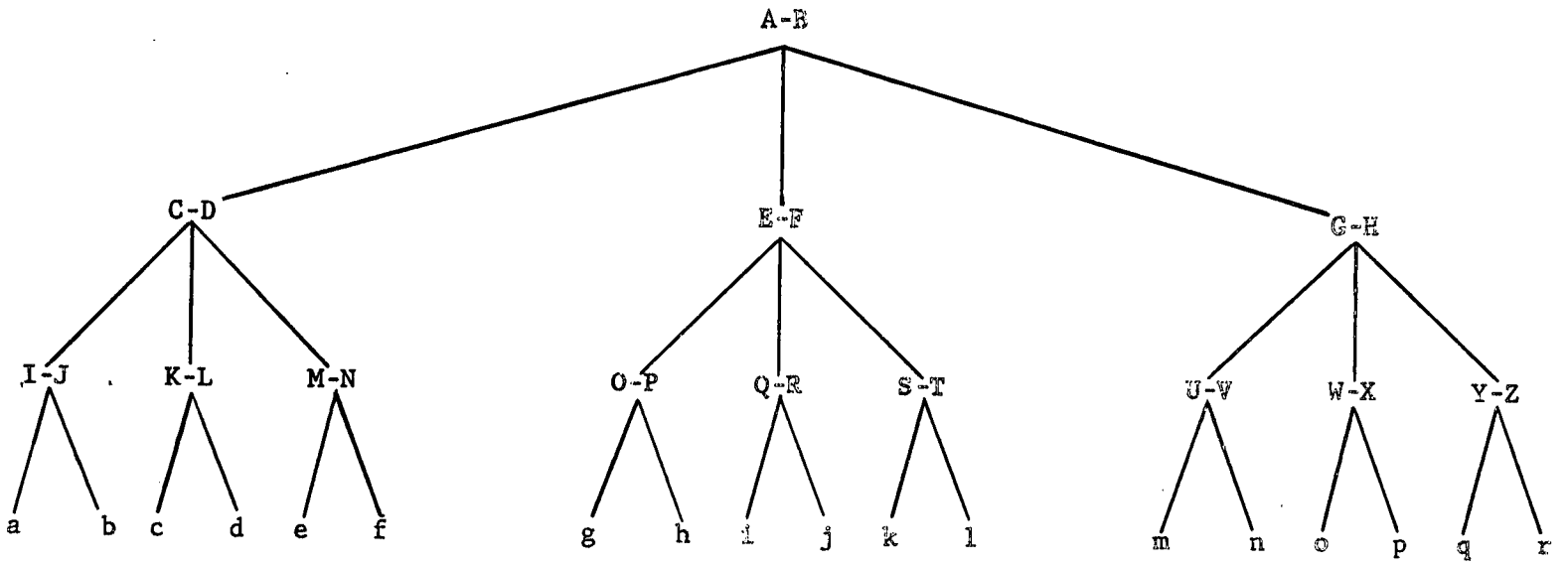
For exposition, consider the ideal family tree, one with no complications of multiple marriages, incest, etc. We will begin with the primary model. A sample tree and its primary map are illustrated in Diagram III and Diagram IV, respectively. For N people in such an ideal tree the total number of computer

words needed to store the map is 2N(N-1) or, for this special case 3,784. Note that storage requirements increase with N^2. The annexing process, in order to add one new person to the map of N, will require 2N(N+1)(N+1-1) - 2N(N-1) = 4N additional words. (Certain savings in space could be realized by avoiding the use of list structures, but all calculations are made with list structure storage as indicated in each example.) If the statement which introduced the new person related him to x, this relationship may be entered immediately. The other entries, such as connecting the new person with y, require an examination of the relation between x and y and a table look-up. This must be carried out for each of the other people in the map. In effect, the annexing process is doing much of the work of the retrieving process; the latter is quite trivial, finding R(x,y) requiring only the locating of the list for x, locating the word holding y, and finding the symbol in the next cell.

An immediate and trivial modification of the primary model is illustrated in Diagram V. Here a savings in space is realized by naming the list of offspring and using this name in association with each parent, thus avoiding duplication of the list. The total number of words required for the map of N people is 2M(N-2) + 4 + 2(N-1)S, where M is the number of marriages and S is the number of single people. For our example, this amounts to 2,644 words. The annexing process remains as complex, although it need not execute so many operations. The retrieving process is identical with that of the primary model.

The first model to make use of properties of the family relations is labeled Relational Model A and appears in Diagram VI. This model takes advantage of the fact that the husband-wife relation is symmetric, thereby eliminating redundancies found in the primary map. Each person has associated with him a list. The first item on his list is the name of the list of his spouses, the second item names the list of his children, the third item names his father, the fourth names his mother. If S_i is the number of spouses for person i, and C_i is the number of children of person i, this map requires a total of $\sum_{i=1}^{N} (4+S_i+C_i)$ computer words. For our example this is 262.

Figure III



Example of Ideal Family Tree

Figure IV

A: B
husband
C
daughter
D
son-in-law
E
son
F
daughter-in-law
G
son
H
daughter-in-law
I
grandson
J
granddaughter-in-law
K
grandson
L
granddaughter-in-law

B: A
wife
C
daughter
D
son-in-law
E
son
F
daughter-in-law
G
son
H
daughter-in-law
.
.
.

C: A .....
mother
B
father
D
husband
E
brother
F
sister-in-law
G
brother
H
sister-in-law
.
.
.

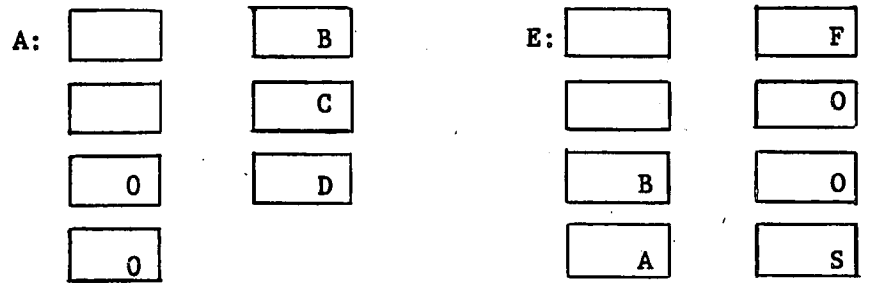
Primary Map

Figure V

A:	B	B:	A	C:	A
90000	husband		wife		mother
	C		90000		B
	daughter				father
	D				D
	son-in-law				husband
	E				E
	son				brother
	F				F
	daughter-in-law				sister-in-law
	G				G
	son		brother		brother
	H				H
	daughter-in-law				sister-in-law
	I				.
	grandson				.
	.				.
	.				

Modified Primary Map

Figure VI



Relational Map A

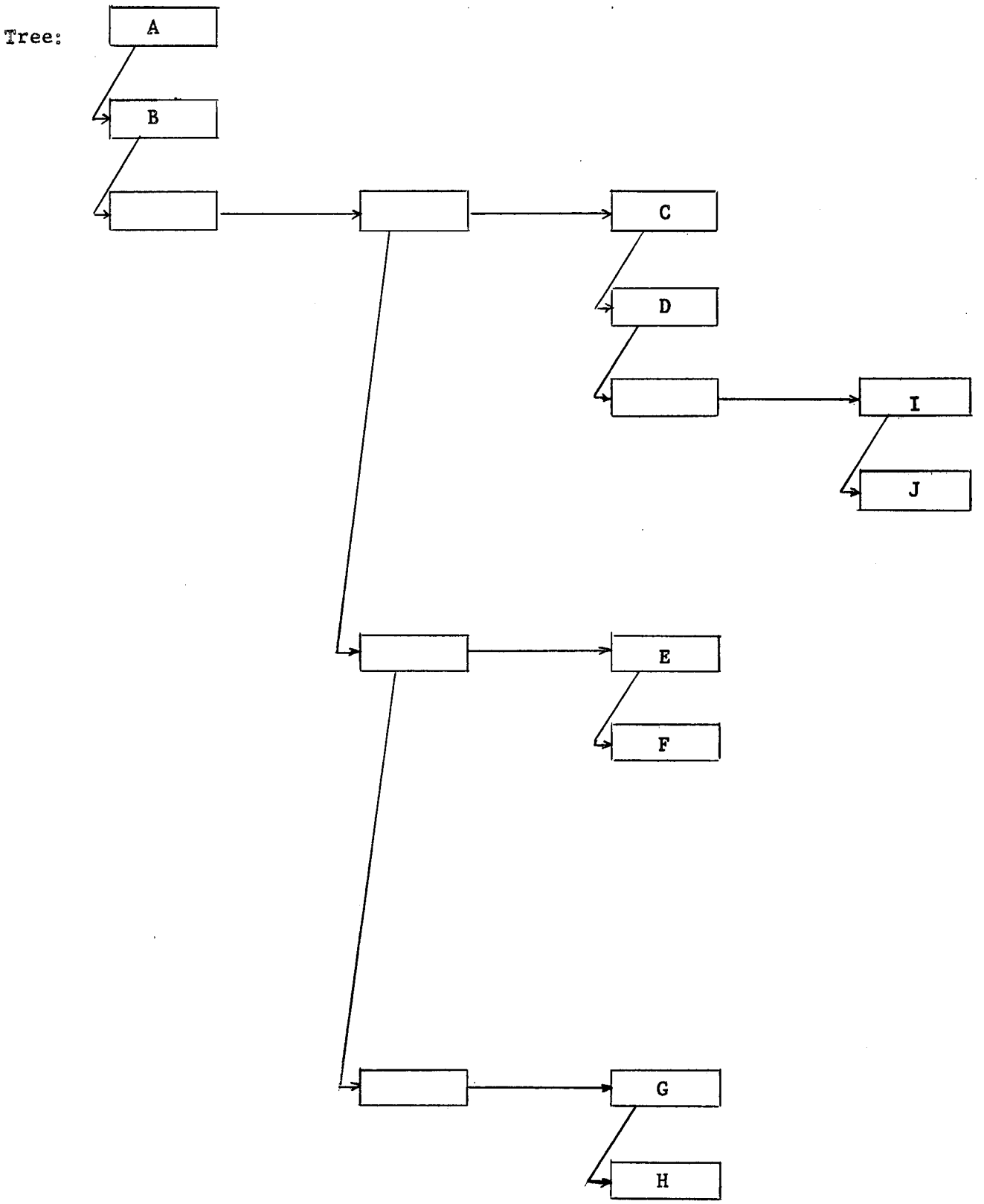
The annexing process is now quite simplified: the new name is entered on the list for the other person mentioned in the statement introducing the new information; a list is created for the new name, the entry for the other person named in the statement is immediate, but some calculation is required for the other entries. The retrieving process, on the other hand, is becoming increasingly complex: a path must be found linking the two names whose relation is requested; the shortest such path is chosen and a table consulted to determine the name for the particular combination of parent-child and husband-wife relations represented by the path.

We are now on the road to finding an efficient model for representing family relations. Of course, it is well known that the common family tree is a very good way of storing this information. It should now be clear why it is good.

1. The small amount of storage space required. Family relations, though they have many names, are all composites of two elementary relations -- that of husband-wife and that of parent-child. Hence, in addition to symbols for the people involved (the letters in Diagram III) we need only two other symbols (the horizontal dashes and near-vertical lines of Diagram III) one for each of the elementary relations.
2. The simplicity of the annexing process. Local information is sufficient for the placing of a new person, that is, giving the person's relation to only one other person is sufficient to find a place for him in the tree.
3. The independence of the retrieving process. This process is unaltered by changes in the tree. It consists of finding a path from x to y, this path consisting of a sequence of the two elementary relations. Corresponding to each possible sequence is a name. For example, parent-parent-child-child=cousin.

One computer representation of the family tree is given in Diagram VII and it is called Relational Model B. Note that storage requirements increase with N. In this case only 87 computer words are required for the map of our example. The annexing process is relatively simple: the other name mentioned in the introducing statement is located in the tree and the new name is inserted in the appropriate cell. (Actually, this process is not sufficient to handle all possible orders of input expressions, such as those involving the relation "aunt". See the discussion of unrestrictedness on pages 27, 30-31, and again

Figure VII



Relational Map B

on page 75.) The retrieving process to find R(x,y) must locate the lowest node which contains both x and y, then proceed to find the shortest path and do a table look-up similar to that required in Relational Model A. The locating of the lowest node could be simplified by a trivial addition to the map, namely, adding another word at the end of each list of children, this word containing the name of the list of those children's parents. This would enable the retrieving process to travel up the tree from each of the given names, rather than requiring that the tree always be entered from the origin. This would turn out to be quite useful when we are dealing with non-ideal trees -- those with more than one beginning or complex cross-references due to re-marriages, etc.

Summary

A model is a cognitive organization for information storage which enables the efficient organization of information. As such it is basic to such general deductive techniques as reasoning by analogy, a process which involves the recognition of similarity between relations, here being accomplished by the matching of the form of two well-specified list structures (maps). It is also expected that both annexing and retrieving processes will prove to be similar (for a variety of contexts) in the sense that they can be applied to various maps. For example, those properties of the processes for the above discussed family relations which make use of the symmetric properties of the husband-wife relation should also be of use in storing and retrieving other symmetric relations. Other relational properties, such as transitivity and reflexivity, will lead to common features for the model.

CHAPTER THREE

Introduction

This section contains a description of the Sentence Appraiser and Diagrammer and examples of the operation of this portion of the program.

The input to this program consists of passages of text written in Basic English, a system extracted from normal English by C. K. Ogden (1933). It was designed primarily to introduce a student of the English language to a self-contained yet uncumbersome context for initial study. The output of the program consists of one list structure per sentence; each of these list structures is intended to be isomorphic to a sentence diagram as it might be constructed by a high school student after making a syntactic analysis of the input. SAD uses only limited information about the words of the passage. This information does not include semantic content, and hence certain types of structure can not be unequivocally determined. Within these limitations, however, great flexibility of the sentence structure is allowed, although the complexity is not equivalent to that which can be understood by the average high school student.

Basic English

The second edition of Webster's New International Dictionary has over 600,000 meaningful symbols listed in its 3,214 pages, if all entries, names, and abbreviations are included. It is safe to say that no one person would recognize all of these, yet the vocabulary of the average man would be a strain for any present-day computer high-speed memory. Since the main interest here is not in the handling of large vocabularies, but in the use of even a small one, the vocabulary size has been limited. English is very generous in providing synonyms, so it is possible to reduce the considered vocabulary to reasonable proportions.

A reading vocabulary of 10,000 words will suffice for much elementary English prose, if they are the proper 10,000 words. C. K. Ogden and I. A. Richards have gone even further by making certain words use meanings which they already possess in situations where a synonym might more commonly be found, thus eliminating more synonyms. Ogden has carefully selected 850 words which can do the work of approximately 10,000. This was done with the prime purpose of developing more convenient teaching methods, hence some further simplifications in syntax have been incorporated and the resulting system is called Basic English. The vocabulary and syntax of this system have been adopted for the purposes of the present work since the system is already complete and consistent and since several books have been "put into" Basic and are available for experimentation.

The 850 words of Basic are not the most frequently used 850 English words. The strategy of choice was rather to find the 850 most useful words. The word "chair" occurs more frequently in English text than does the word "seat," yet the latter is included in Basic while the former is not. The reason for this is that "seat" (or some combination of other Basic words) can replace "chair" in all its uses, whereas the reverse is not true (e.g., seat of the pants). The most striking feature of Basic is the reduction to only 16 verbs plus two auxiliaries. This is possible through the use of slightly more complicated verbal patterns, in conjunction with the use of prepositions as parts of verbs--a very common practice in English. Anything written in Basic, of course, is readily understood by anyone who can understand the same passage in English. All simplifications are eliminations of superfluous complications; none are contradictions of standard English grammar or semantics.

A passage in Basic requires more words than the same passage written with a 10,000 word vocabulary. However, since Basic words are generally short words, and since most passages may be carefully edited while the meaning is preserved, Basic translations are roughly comparable in length to their sources.

When it comes down to actually saying anything, most of the Basic authors have decided to complement the original 850 with various other words and phrases. Odgen has included in the system what he calls international words--those which are used in most languages--such as alcohol, cocktail, bar, vodka, rum, nicotine, cigarette, and tapioca. Also included

are numerals and measurement, currency, and calendar words. In addition there are certain idioms which are accepted, and certain hyphenated words which are listed separately. Pronouns are declined in full, and verbs conjugated in full; nouns may be pluralized. All of the words and phrases, except for the idioms, are listed and defined in all forms in which they may be used in The Basic Words by C. K. Ogden. There are over 1400 in all. In addition, some words may take the prefix un- and/or the suffixes -s, -s', -'s, -ly, -ed, -ing, or -er, increasing the number of distinct symbols still further. The total remains under 3,000.

The simplifications of Basic English are mainly in quantity, although some rules of syntax are also simplified. The real difficulties, however, remain: multiple meanings, multiple syntactic usage, flexibility of form, idiomatic usage, flexibility of punctuation, and so on.

Syntactic models

SAD is divided into two parts. The first merely converts the Basic English text, punched into IBM cards approximately as it would be typed, to lists of IPL symbols, one list per sentence. The symbol for a word is an A-region symbol, except in the case of words with one of the suffixes -ing, -ed, -er, -'s, or -s' in which cases a B-region symbol is substituted. The mechanics for this transformation are embodied in a simple 650 program and will not be further discussed. The second part of the program is written in the IPL-V language and will be discussed in the terms of the preceding section, since it is to be viewed as a model for the storage of syntactic relationships. The programming to be discussed comprises the annexing process of this model; the list structure format or sentence diagram is the map; the retrieving function may be of many kinds, depending upon the questions which may be asked of the system. This section will discuss one such retrieving function; its use and the use of some others will be examined in the following section, where they will be viewed as portions of the annexing process for the semantic model.

The annexing process for SAD

This program accepts as input lists of A-region and B-region symbols each of which corresponds to one of the 1413 words in the Basic Dictionary, or to a prefix or suffix, or to one of the punctuation marks . ; - -- , ( ) / (exclamation point) or $ (question mark). The program is divided into three parts: the history package, the part of speech routines, and the working storage.

The History Package

This portion of the program consists of information which is acquired by a human subject from experience with the particular language, and enables recognition of certain word combinations and types. Some of the major parts are:

N11	=	List of prepositions associated with each verb.
C2	=	List of idioms.
C1	=	List of verb forms: legitimate sequences of verbs and auxiliaries, coupled with information about transitivity, voice, tense, mood, etc.
X16	=	List of parenthetic expressions.
X17	=	List of dates and places.

In addition to these lists, the history package contains the dictionary of words, punctuation, and parts of speech. Each of the A- and B-region symbols names a list of parts of speech. These parts of speech are designated by P-region symbols. The list for each word has been determined by the assignment given in The Basic Words, and the order of their occurrence is in the order of frequency of use as given in that dictionary. The parts of speech are those commonly studied in introductions to syntactic analysis and the collection is not as elaborate as those proposed by linguists, whose categories number about eighty. This simplification is partly due to the use of Basic English, which requires less elaborate analysis (there are only 16 verbs, for example, whereas 23 English verb classifications can be distinguished); it is mainly motivated, however, by the desire to study the organization of the system in simplified form without the refinements necessary for the construction of a sophisticated performance program.

Part of Speech Programs

Each of the P-region symbols is the name of a program. Progress through the sentence is accomplished by obtaining the P-region symbol for the first word, executing the program for that symbol, then obtaining and executing the routine for the next word, and so forth. These programs are not functions in the usual sense, but are operations which alter the internal state of the computer by changing the syntactic map and/or temporarily remembering certain information so that the decision to alter the map may be made in the future. One important function of the part of speech routines is to detect unpermitted sequences and choose an alternative part of speech designation for the current word or an immediately preceding word.

Routine P0 is a special routine which reads the sentences one at a time, initializes the working storage cells, and transfers to the first word. The routine for a period (or other terminal punctuation) is P1, which prints the sentence map and erases the working storage.

Working Storage

A series of storage cells designated by the regional symbol X are used as working space. Two of them, X0 and X1, are used to keep track of progress through the sentence. P0 loads the sentence into X0, so that initially this "push-down" cell holds the stack of A-region and B-region symbols which comprise the sentence, with the first word on the top. As the program proceeds through the sentence, words are taken from the top of X0 and placed in X1. Thus the current word and the immediately preceding word are immediately available in the context of the part of speech routines. The verb for the action of moving the sentence in this fashion is "to slinky," and the process is referred to as such in the program listings. At any time, then, X0(0) contains the current word, X0(1) contains the next word, and X1(0) contains the previous word. When examining the word "boy" in the sentence "Every good boy does fine," the cells would look like this:

X0	X1
boy	good
does	every
fine	

The symbol which has been designated "good" (which is A448) would be a list structured as follows:

A448 P21
P23 0

where P21 is the name of a program which is executed when the current word is an adjective, and P23 is the name of a program which is executed when the current word is a noun.

Cells X2, X3, X4, X5 and X6 are storage cells for phrases currently being built up. They may be viewed as micro-maps whose structures reflect the relationships of adverbs and their modifiers, adjectives and their modifiers, nouns and their modifiers, verbs and their auxiliaries, and prepositions and their objects. Each adverb, adjective, and noun is formed into a phrase, which is a two symbol list; the second symbol is the adverb, adjective, or noun itself, the first symbol names a list of modifiers. (See Figures IX, X and XI.) Each symbol of the list is an adjective or adverb phrase, as the case may be, which in turn has its own list of modifiers. The cell X2 holds adverbial phrases until referents for them are determined. X3 has a similar function for adjectives, and X4 is for nouns. X5 is a temporary collection point for parts of the verb. X6 holds prepositional phrases until referents are found. The prepositional phrase is also a two symbol list (See Figure VIII); the first symbol is the preposition, and the second names a list of objects, each symbol of which is a noun phrase. Cells X2, X3, and X4 can hold, in addition to their respective phrases, the comma symbol, P3, or any conjunction. When the contents are called for, all entries are combined with the conjunctions and punctuation available.

Just as referents may occur after their modifiers, modifiers may occur after their referents. To handle this latter situation another set of cells is used to hold the names of phrases which either need or can use modifiers or objects which have not yet been encountered. These are cells X7 for verb phrases needing predicate adjectives, X8 for prepositional phrases needing referents, X9 for prepositional or verbal phrases needing objects, X10 for verbal phrases needing subjects, X11 for verbal phrases needing adverbs, and X13 for referents needing prepositional phrases.

Figure VIII

9-0	Axxxx		preposition
	9-1	0	list of objects
9-1	0		List of objects
	9-x		noun phrase or conjunction
	.		.
	.		.
	.		.
	.	0	.

Format of Prepositional Phrase

Figure IX

9-0	9-1		Adverb phrase
	Axxxx	0	adverb
9-1	0		List of modifiers
	9-x		adverb phrase or conjunction
	.		.
	.		.
	.		.
	.	0	.

Format of Adverb Phrase

Figure X

9-0	9-1		Adjective phrase
	Axxxx		adjective
9-1	0	0	List of adverb phrases
	9-x		adverb phrase or conjunction
	⋮		⋮
	0		

Format of Adjective Phrase

Figure XI

9-0	9-1		Noun phrase
	Axxxx		noun
	9-x		noun phrase
	⋮		⋮
	0		
9-1	0		Modifiers
	9-2		adjectives
	9-3	0	prepositional phrases
9-2	0		List of adjectives
	9-x		adjective phrase or conjunction
	⋮		⋮
	0		
9-3	Axxxx		preposition
	9-x		list of objects
	Axxxx		preposition
	9-x		list of objects
	⋮		⋮

Format of Noun Phrase

In each cell the top entry is a Y-region symbol which designates the state of demand, Y0 if nothing needed, Y1 if needed, or Y2 if not needed, but usable. The second symbol in each of these cells names the appropriate phrase.

Each of these working cells, then, represent expectations set up by what has come before. For example, when a preposition is encountered, and if it is not part of the verb, X9(0) is loaded with the symbol Y1 which indicates that an object is needed. In X9(1) is placed the name of the prepositional phrase requiring the object. When a noun is encountered, a check is made of X9(0) to see if it is needed as an object, and, if it is, it is assigned according to the contents of X9(1). After the assignment is made, X9(0) is loaded with the symbol Y2, which indicates that an object is usable, that is, the preposition may take a compound object, though it need not.

Working storage cell X12 holds either X2, X3, X4, X5, X6, or C0, depending upon which of these lists was most recently altered, that is, which type of phrase was most recently constructed. The use of this information is not apparent without detailed examination of the program, but it is important in making decisions when multiple syntactic functions are possible. Certain sequences are not allowed, and X12 keeps track of these.

X15 holds conjunctions. The use of these is not determined in the conjunction routine, but is decided by the local context, including information from X12.

X18 is used for the handling of appositions. When an apposition is signalled by appropriate punctuation, such as left parenthesis, dash, or sometimes comma, the previous context is sealed off by preserving all working cells and loading them with the symbol Y4. The next portion of the sentence is then treated in isolation, with provision made for reference between levels. When the apposition ends, as signalled by a right parenthesis, dash, or comma, the apposition context is cleaned up and the previous context is restored. X18 holds symbols indicating the depth to which the appositions have been carried, thus allowing appositions within appositions, and so on, to any depth.

The syntactic map for SAD

The syntactic map, or sentence diagram, is composed of the micromaps which are developed in the working storage cells X2, X3, X4, X5, and X6. The format for the syntactic map is outlined in Figure XIII. C0 is a list of clauses. Each symbol in C0 is either the name of a clause, a punctuation mark such as a comma, or a conjunction. The last two entries are the Y4 symbol which is an internal punctuation mark used in making erasures. Each clause is a four symbol list; the first symbol always names the list of subjects, the second names the main verb phrase, the third names a list of indirect objects, and the fourth names a list of direct objects (or predicate nominatives). The list of subjects and the list of objects are describable lists, each of whose symbols names either a noun phrase or a verbal phrase (i.e., a gerund or infinitive construction). These forms are discussed above and illustrated in Figures XI and XII.

The only additional feature of the completed map is the provision for cross-referencing of noun phrases. If two or more noun phrases refer to the same object or person, this is noted by appending the names of equivalent noun phrases at the bottom of the given phrase. For example, the sentence, "John saw his brother who was on his way home," has two clauses: "John saw his brother," and "who was on his way home." These two clauses are connected by the common reference of the object of the first--brother--and the subject of the second--who. This connection is denoted in the map of the sentence by a cross-referencing of the noun phrases for the two nouns. If the name of the phrase for "brother" is 9-0 and the name of the phrase for "who" is 9-1, these two phrases would appear thus in the completed map (they are here taken out of context):

	9-0	0 brother 9-0	0
and	9-1	0 who 9-1	0

Figure XII

9-0	9-1	Verbal phrase
	Axxxx	verb or auxiliary
	Axxxx	verb or auxiliary
	.	.
	.	.
	. 0	.

9-1	Txxxx	Verb information list
	Txxxx	
	Txxxx	
	Txxxx	
	Txxxx	
	Txxxx	
	Txxxx 0	

Format of Verbal Phrase

Figure XIII

C0	9-1		List of clauses
	9-x		clause or conjunction
	.		.
	.		.
	.		.
	9-x		
	Y4		
	Y4	0	
9-1	0		Clause
	9-2		subject
	9-x		verb phrase
	9-3		indirect object
	9-4	0	direct object
9-2	0		List of subjects
	9-x		noun phrase or conjunction
	9-x		.
	.		.
	.		.
	.	0	.
9-3	0		List of indirect objects
	9-x		noun phrase or conjunction
	9-x		.
	.		.
	.		.
	.	0	.
9-4	0		List of direct objects
	9-x		noun phrase or conjunction
	9-x		.
	.		.
	.		.
	.	0	.

Format of Syntactic Map

In addition to this use of cross-referencing, as derived from relative pronouns, other cases are possible. One such case is in the equivalence of nouns and their pronouns. This possibility has not been explored and no such cases are handled in the present version of SAD. Another case is included in the SAM routine, which amends the syntactic map in order to construct the semantic map of family relationships. This is done by SAM for instances of equivalence denoted by the verb "to be," as in the sentence, "John is my brother." Here "John" and "brother" refer to the same person, and SAM makes note of this by the above mentioned form of cross-referencing. This will be more fully discussed in the next chapter.

The retrieving process for SAD

The syntactic map encodes syntactic information in such a manner that grammatical relationships may be retrieved by a set of processes which are independent of the specific contents of the map, but depend only on the general format. This enables the answering of many questions about the structure of a sentence which has been stored in such a fashion. One such set of questions might be "What are the objects of the clause whose subject is A0?" Subjects may be easily generated by the following routine:

```
K1    40W0        9-0
      10C0
9-0   12H0
      60W0
        J81       9-1
9-1     J60
      709-3       9-2
9-3   12H0
        J81
        K0        9-1
9-2   30H0
        J60
      70 0        9-0
```

The subroutine K0 in the above process finds a noun in (0) and the name of the clause in which that noun appears as subject in W0(0); it restores H0. If we wish to find the objects of a clause whose subject is A0, the routine K0 might be as follows:

```
K0   10A0
       J2
     70 0
     11W0
     30W0
       J6
     30H0
       J6
     30H0
       J84        0
```

Execution of the routine K1 would bring to (0) the name of the list of objects of the first clause whose subject is A0 and set H5 +, or, if none existed it would set H5 - and yield no output.

Routines could be devised to answer many such questions. One such retrieving process was coded for use with SAM. This routine, N1, constructs the list M0, which contains all noun phrases in the diagrammed sentence, together with a list of directions for locating each of them. The format of M0 is given in Figure XIV. Each symbol on M0 names a one-symbol, describable list. The symbol on each of these sublists is the name of a noun phrase. The description list is not used in the conventional sense, but names a list whose symbols are the J80's. Inputting C0 to (0) and executing the J80's in the order of their occurrence will yield the noun phrase. This recovery list then contains a great deal of information about the use of the given noun phrase. As one example, it may be noted that the second symbol on the recovery list indicates whether the noun phrase is part of the subject, verb, indirect object, or object portion of a clause.

N1 proper finds the main noun phrases of the diagrammed sentence. These locations are then used as input to N3 which enters the given phrase on M0 and generates and enters all noun phrases which are parts of the modifiers for the input. N3 is used recursively for this function. The list M2 is used as data for the construction of the recovery lists; it is a describable list of the J80's in increasing order.

Chapter Four discusses the use of this SAD retrieving process viewed as an annexing process of the semantic model developed in SAM.

Figure XIV

M0	0		List of nouns
	9-0		noun and recovery list
	⋮		⋮
	⋮	0	⋮
9-0	9-1		recovery list
	9-x	0	noun phrase from C0
9-1	0		Recovery
	J8x		"Find" primitive IPL operation
	J8x		
	⋮		⋮
	⋮	0	⋮

Format of M0

Figure XV

M10	0		Condensed list of nouns
	9-0		nouns and modifiers
	⋮		⋮
	⋮	0	⋮
9-0	9-1		modifiers
	9-2	0	nouns
9-1	0		List of modifiers
	9-x		modifier list from C0
	⋮		⋮
	⋮	0	⋮
9-2	0		List of equivalent nouns
	Axxxx		noun or pronoun
	Axxxx		noun or pronoun
	⋮		⋮
	⋮		⋮
	⋮		⋮
	⋮		⋮

Format of M10

Limitations on SAD

SAD is not capable of generating a syntactic map of all grammatically correct English sentences. The limitations on vocabulary have been discussed at length above. In addition to these, there are two classes of limitations for these routines: those which result from the limited information supplied as input, and those which result from the lack of generality of the P-routine annexing processes.

Input information limitations

SAD does not discriminate each word which may be used as input, but knows them only by their syntactic categories. One exception to this rule is in the case of the preposition "to" which is distinguished from other prepositions in order to determine the presence of infinitive constructions. Aside from this, the input to SAD may be considered as a sequence of the following syntactic categories: articles, adjectives, adverbs, nouns, prepositions, verbs, auxiliary verbs, pronouns, and relative pronouns.

This limited information is not always sufficient for the determination of the correct sentence diagram. For example, the sequence "article-noun-preposition-article-noun-preposition-noun" could represent either "the man in the house of glass" or "the man in the house with glasses." The two phrases have different syntactic forms, but the decision between them depends upon detailed information about the meanings of the words involved.

A second example is the sequence "noun-verb-noun-conjunction-noun-conjunction-noun-verb-noun" which could represent the sequence "Jack gave candy and gum and Jill gave toys" or the sequence "Jack gave candy and Jill and John gave toys." Although commas are often employed to make such meanings clear, the omission of this punctuation is permissible whenever the meaning is clear to the average reader. Hence, in many instances of English text, sequences occur which are ambiguous when reduced to syntactic categories. Whenever such is the case, SAD can not guarantee a correct parsing of the sentence.

The parsing routine, however, always makes a decision. In the case of determining the referent of a prepositional phrase, the decision is always to use the noun immediately preceding the preposition if one exists. As seen above, this decision will at times lead to an incorrect parsing; however, some word sequences exist whose syntactic category sequence is noun-preposition-noun and for which the above rule yields a correct parsing. In fact, the parsing is correct for most such sequences taken from English text.

Lack of generality

The generality of SAD can only be determined by extensive exploration of the program as it analyzes long passages of text taken from unedited sources. Some limitations are, however, apparent without such experimentation.

One of the most fundamental of these limitations is in the recognition of the length of appositions whose termination is not explicitly marked. In the sentence, "The man who came to dinner was not welcome," the clause in apposition, "who came to dinner," clearly terminates before the main verb "was." Although SAD would note the start of the apposition with the relative pronoun "who," cues, recognizable to SAD, which would signal the termination of the apposition at the proper place are not present. In general, the termination must be signalled by a comma, dash, right parenthesis, or period, and the beginning must be signalled by a comma, dash, left parenthesis, or relative pronoun.

In addition, SAD does not handle quotations, interjections, or the special constructions frequently used with questions.

Example of the operation of SAD

As an example of the parsing procedure we will consider the sentence, "They were doing well till the coming of Sam who is Barbara's brother." The first phase of the program converts this string of symbols into the following sequence: A1156, A1260, A301, A1257, A1170, A1147, A228, A743, A1830, A1276, A562, A1831, B12, A157. The dictionary, stored

in the computer, has the following sequence of part of speech routines associated with these symbols: P27, P44/P25, P44/P25, P21/P23, P22/P21, P24/P26, P21, P44/P25/P21/P23, P24, P23, P48/P27, P44/P25, P21, P23.

P0 stores the A-region symbols in X0 and X1, with the part of speech information (adjective) for the 's suffix (B12) taking precedence over the part of speech information (noun) for the root Barbara (A1831). The first part of speech for the first word is found and the routine executed.

P27 (pronoun): The noun routine is executed. Checks for unpermitted sequences are made and none are found. A check is made of X10 and X9 and the presence of Y0 in each indicates that no subject or object is needed. A noun phrase is formed and stored in X4, and also in X13(1). Y1 is placed in X13(0) to indicate that the noun is available as the referent for a preposition. The next word is found and its part of speech routine is executed.

P44 (aux. verb): The verb is saved in X5 and the next word is checked. It is found to be another auxiliary verb. The routine is executed again.

P44 (aux. verb): The verb is saved in X5 and the next word is checked. It is found to be a verb. The routine for verbs is executed.

P25 (verb): The verb is placed in X5 with its auxiliary and the combination is found in the verb table. A list of descriptive information is placed on top of the verbs. A check is made for associated prepositions, but none are found. The routine for a complete verb form is called.

P45 (complete verb): The contents of X5 are combined into a verb phrase and the phrase is placed in C0. No adverbs are available to modify it. X4 is found to contain a noun, and this is used as the subject. The need of adverbs is marked on X11. The verb information list is checked and the possibility of an indirect object is by-passed. Need of an object is marked, and the verb phrase is made available as the referent of a preposition, thus erasing the possibility of the first word of the sentence being used as referent for future prepositions.

P22 (adverb): The immediately preceding word is found not to be either an adjective, a noun, or a pronoun, so the current word is assumed to be correctly described as an adverb. No others are available in X2 to modify it. It is used where needed, that is, to modify the preceding verb phrase.

P24 (preposition): Since this is not the preposition "to," it is formed into a prepositional phrase and it is found from an examination of X13(0), which contains a Y1, that the phrase has a referent. X13(1) holds the verb phrase, to which is now attached the new prepositional phrase. Y2 is placed in X13(0) indicating that the verbal phrase may accept other prepositional phrases. The need of objects by the preposition is marked in X9 and adverbs are marked as no longer needed. Thus, succeeding adverbs will not be used to modify the main verb, but will be saved to modify adjectives, other adverbs, or future verbs, unless such do not appear before the end of the sentence.

P20 (article): The article is entered into X3, without being formed into phrases.

P44 (aux. verb): The word is placed in X5. The next word is not a verb or adverb so the current contents of X5 is treated as a verb.

P25 (verb): The verb is placed in X5 and the contents of X5 (only the present verb) is looked up in the table. It is not found, so the verbal routine is executed.

P46 (participle, gerunds): The next word is checked to determine the function of the verbal. Next is not an adjective, so verbal is not used as adverb. It is formed into a phrase, which allows it to accept objects and adverbs and prepositional phrases. The previous need of an object is noted, so the verbal, in the form of a verb phrase, is used as a noun.

P23 (noun): The above verbal phrase is used as the object of the preposition "till."

P24 (preposition): The verbal phrase, when used as a noun, was made available as the referent for a preposition. The present word is used to supply this modification. Its need of an object is marked.

P23 (noun): This noun is supplied as the needed prepositional object.

P48 (relative pronoun): An immediately preceding referent is found. (Otherwise this word would have been used as a pronoun). A noun phrase is formed of the relative pronoun, and cross-referencing is made between it and the phrase for the immediately preceding noun. The relative pronoun is made available as a referent and stored in X4. The start of an apposition is noted by placing Y4 punctuation in the working cells and marking X18.

P44 (aux. verb): It is stored in X5 and the subsequent words are checked. Since the next word is not an adverb nor a verb, the current contents of X5 are treated as a verb.

P25 (verb): The single word in X5 is found to be a complete verb form.

P45 (complete verb): It is formed into a phrase, and entered in C0. No adverbs are available for it. The contents of X4 (the relative pronoun phrase) are used as the subject. The verb is a form of "to be", so the usability of a predicate adjective or predicate nominative is noted.

P21 (adjective): No illegal sequences are found. Predicate adjective not needed, so the adjective phrase is saved in X3.

P23 (noun): No illegal sequences are found. Noun is modified with the preceding adjective, and is made available as a referent. No subject, object, or predicate nominative is needed, so the noun phrase is saved in X4.

P1 (period): It is discovered that the current context is for an apposition. Since it must end now, the right parenthesis routine is called.

P9 (right paren.): A check is made of working storage to find if any words have not yet been assigned a place in the structure. The modified noun phrase is found in X4 and is used where usable, as the predicate adjective for the verb of the subordinate clause. X18 is popped up, and the previous context of working storage is restored. The next word is sought, but since X0, X1 cells were slinkied backward before entering P9, the next word in this case is a period.

P1 (period): It is discovered that the current context is that for the main sentence, so the working cells are cleaned up by using their contents where usable. In this case there is nothing left over. The diagram is printed.

The completed diagram appears in C0 as follows, with local symbols as generated in the output.

C0	9-0	
	9-1	
	Y4	
	Y4	0
9-0	0	
	9-2	
	9-3	
	9-4	
	9-5	0
9-1	0	
	9-6	
	9-7	
	9-8	
	9-9	0
9-2	0	
	9-10	0
9-3	9-11	
	9-12	
	doing	
	were	0
9-4	0	0
9-5	0	0
9-6	0	
	9-13	0
9-7	0	
	9-14	
	is	0
9-8	0	0
9-9	0	
	9-15	0
9-10	0	0
	they	0
9-11	0	
	9-16	
	9-17	0
9-12	verb information	
9-13	0	
	who	
	9-36	0
9-14	verb information	
9-15	9-18	
	brother	0
9-16	0	

	9-19	0
9-17	till	
	9-20	0
9-18	0	
	9-21	
	9-22	0
9-19	0	
	well	0
9-20	0	
	9-23	0
9-21	0	
	9-24	0
9-22	0	0
9-23	9-25	
	9-26	0
9-24	0	
	Barbara	0
9-25	0	
	9-27	
	9-28	0
9-26	0	
	9-29	
	9-30	
	9-31	
	9-32	0
9-27	0	
	9-33	0
9-28	of	
	9-34	0
9-29	0	0
9-30	0	
	9-35	
	coming	0
9-31	0	0
9-32	0	0
9-33	the	0
9-34	0	
	9-36	0
9-35	verb information	
9-36	0	
	Sam	
	9-13	0

Summary

SAD operates by the execution of a sequence of programs which corresponds to the sequence of part of speech information which is given as input. The process proceeds in a left-to-right fashion, with deviations from one word at a time analysis being made in special cases. These deviations never leave the immediate context of the word under consideration.

The analysis makes use of eleven working storage cells in a manner which allows efficient storage of intermediate results while permitting flexibility in leaving open verious possibilities for the sentence development. Small structures are constructed during progress through the sentence in order that a single symbol, designating a stardard format which retains information about the relation of several words, may replace the several word-symbols. Thus the small number of working storage cells can handle elaborate developments. Further, decisions as to syntactic usage are made immediately and changed only when other information forces the change. This procedure decreases the number of possibilities which must explicitly be kept open simultaneously.

Yngve (1960) has devised a model for the synthesis of English sentences. His model exhibits preferences for word orders which are characteristic of human speakers. These preferences are due to the use of strategies designed to overcome the handicap of a small immediate memory. The present work makes use of similar features for the analysis of sentences, and hence is closely related to Yngve's work.

CHAPTER FOUR

Introduction

As an example of the extraction of implicit meaning from a passage of English text, this chapter describes in detail the SAM program which makes deductions about family relations. The input to SAM is the syntactic map, C0, constructed by SAD, and the list of nouns, M0, which is produced from C0 by the retrieving function N1. From these is constructed a list, M20, which contains all explicitly stated information about family relations. This information is then used to construct the family tree, T0.

Semantic models

The annexing process for SAM

Routine N1 has been discussed in the preceding chapter as an example of one retrieving process of the SAD program. N1 is also a part of the annexing process for SAM. The complete sequence from C0 to the family tree, T0, may be considered as a retrieving process of the syntactic model or as an annexing process of the semantic model, since information is obtained from the syntactic map and used to construct the semantic map.

After the formation of the list of nouns M0, as illustrated in Figure XIV, the syntactic map C0 is modified by the interpretation of certain word meanings. At this point, the words cease to be considered solely by their syntactic category membership. Routine N5 uses the syntactic information available from M0 to discover subject-object combinations whose main verb is a conjugation of "to be". When these are found, the subject and object are marked equivalent by the same scheme of cross-referencing as used with relative pronouns and their referents.

Routine N4 forms the next step in the sequence from C0 to M20. A condensed list of nouns, M10, is formed from M0 and C0. The format of M10 is illustrated in Figure XV. Each symbol on M10 names a list which contains the equivalent nouns and a combined set of all modifiers of the nouns. The head of each of the modifier lists is altered from 0 to the name of the specific noun which was modified in order to carry along this information to M10.

As an example of this procedure, consider the sentence, "John is Mary's brother". After N5 has been executed, the altered syntactic map will look like this:

	C0	9-0	
		Y4	
		Y4	0
	9-0	0	
		9-1	
		9-2	
		9-3	
		9-4	0
	9-1	0	
		9-500	0
	9-2	0	
		9-6	
		is	0
	9-3	0	0
	9-4	0	
		9-700	0
	9-500	0	
		John	
*		9-700	0
	9-6	verb information	
	9-700	9-800	
		brother	
*		9-500	0
	9-800	0	
		9-9	
		9-100	0
	9-9	0	
		9-11	0
	9-100	0	0
**	9-110	brother	
		Mary	0

* Indicates changes made by N5.
** Indicate change made by N4.
Local names of the form 9-XXX indicate common reference between list structures.

The list of nouns will look like this:

M0	0	
	9-0	
	9-1	0
9-0	9-2	
	9-500	0
9-1	9-4	
	9-700	0
9-2	0	
	J80	
	J81	
	J81	0
9-4	0	
	J80	
	J84	
	J81	0

The condensed list of nouns will be:

M10	0	
	9-0	0
9-0	0	
	9-1	
	9-2	0
9-1	0	
	9-800	0
9-2	0	
	John	
	brother	0

The final step in extracting explicit information is the formation of the list of elementary relations, M20. The format of M20 is given in Figure XVI. Basic English permits only eight words pertaining to family relations. These are father, mother, brother, sister, offspring, brother-in-law, sister-in-law, and married. Programs have been written which give meaning to the first five of these. Routine N6 takes in turn each of these five, or any other set as specified by the contents of list M15, and locates each occurrence on M10. The modifiers of the located relation word are then generated. If the proper name of a person (not a pronoun) occurs as an adjective or as the object of a preposition, then the proper name from the modifier is paired with each proper name in the list of equivalent noun forms. These two proper names are coupled with the relation word and the three-word list is entered on M20.

Figure XVI

```
M20          0                          List of elementary relations
             9-0
             9-1
              .
              .
              .
              .          0

9-0          Axxxx                                  first name
             Arrrr                                  relation word
             Ayyyy      0                           second name

9-1          Axxxx                                  first name
             Arrrr                                  relation word
             Ayyyy      0                           second name
              .
              .
              .
```

69

Figure XVI

```
M20          0                          List of elementary relations
             9-0
             9-1
              .
              .
              .
              .          0
```

Consider our previous example as the program continues. The first relation word from M15 is father. This is not found on M10, nor is mother. But brother is located (9-0 from M10). The list of modifiers is examined and each modifying word is checked to find any proper names. The adjective phrase 9-11 (from C0) is located; the adjective is the proper name Mary, and the head of the phrase is brother, indicating that the phrase "Mary's brother" occurred in the sentence. The list of equivalent noun forms (9-2 from M10) is tested for proper names, and John is located. Thus a list of John-brother-Mary is formed and entered on M20. No other proper names are found, so the other relation words are examined. Neither sister nor offspring occur on M10, so the extraction of explicitly mentioned family relations is completed for this sentence.

Formation of the semantic map proceeds from the list of elementary relations. Each of the proper names in a given relation is located in the tree. The selection of an annexing routine depends upon the relation and upon the outcome of this search.

R0 is executed if the relation is brother and neither of the two proper names currently resides in the tree.

R1 is executed if the relation is brother and the first name but not the second is a tree resident. If the second is resident, but not the first, they are reversed and R1 is executed, since brotherhood is symmetric.

R2 is executed if the relation is brother and both names are already residents of the map. This involves forming the union of the parents of the two individuals, if the parents are different. All of the offspring are combined into one list and the names of the parents are placed together. This accomplished by the subroutine S10.

R3 is executed if the relation is offspring and neither name is a resident. If the relation is father or mother and neither name is resident, the names are reversed and R3 is executed.

R4 is executed if the relation is offspring and the second name, or parent, is a resident but the first name is not. If the relation is father or mother and the first name, or parent, is a resident but the second is not, the names are reversed and R4 is executed.

R5 is executed if the relation is offspring and the first name, or offspring, is a resident but the second is not. If the relation is father or mother and the second name, or offspring, is a resident but the first is not, the names are reversed and R5 is executed.

R6 is executed if the relation is offspring and both names are resident. If the relation is father or mother and both names are resident, the names are reversed and R6 is executed.

The semantic map for SAM

The format for the family tree is illustrated in Figure XVII. SAM is not restricted in the order in which the elementary relations are presented, nor is it required that all individuals be descended from a common ancestor. Furthermore, more than one family may be discussed. To allow for these possibilities, a list is kept of entries to family trees. It is this list, T0, which is consulted to find if a given name has previously been entered in relation to another. Every time a new family unit is constructed, its name is added to T0, and the names of units which may now be reached by way of the new unit are deleted from T0.

Each basic family unit is represented by a six-symbol describable list. The first symbol is the name of one parent, the second symbol is the name of the other parent; the fourth symbol is the name of the family unit for the parents of the first parent, the fifth symbol is the name of the family unit for the parents of the second parent. Any of these four symbols may be 0 if the information is not known. Thus, given as input the location of any person, the location of his parents can be found by executing J83. The sixth and final symbol on the list is the name of the list itself. This enables traveling "up" the tree, as well as the convenient location of spouses. The third symbol on the family unit list is the name of a list of offspring. Each symbol on the offspring list is the location of the name of the offspring in its own family unit. That is, each symbol on the offspring list is the name of the first or second list cell on another six-symbol family unit list.

Figure XVII

9-0	0	0	Name of family unit
	Axxxx		name of first parent
	Ayyyy		name of second parent
	9-1		list of offspring
	9-xxxx		parents of first parent
	9-yyy		parents of second parent
	9-0	0	
9-1	0		List of offspring
	9-2		location of offspring
	9-x		.
	.		.
	.		.
	.	0	.
9-2	Acccc		offspring
	Adddd		spouse of offspring
	9-4		list of children
	9-0		
	9-x		parents of Adddd
	9-3	0	
9-3	0	9-2	Family unit of offspring Acccc
9-4	0		List of grandchildren
	9-x		location of grandchild
	9-5		.
	9-x		.
	.		.
	.		.
	.	0	.
9-5	Aeeee		grandchild
	9-x		list of great-grandchildren
	9-x		parents of spouse of Aeeee
	9-2		
	9-6	0	
9-6	0		Family unit of grandchild
	Affff	9-5	spouse of Aeeee

Format of Family Unit List

An example of the operation of SAM

Consider a continuation of the example discussed earlier in this chapter. Assume that preceding sentences have been processed and that the resulting family tree entry list is structured as follows:

```
TO      0
        9-0
        9-1         0
9-0     0
        Peter
        0
        9-2
        9-100
        0
        9-0         0
9-1     0
        Alice
        0
        9-3
        9-110
        0
        9-1         0
9-2     0
        9-5         0
9-3     0
        9-7
        9-120
        9-130
9-4     0
        Jane        9-5
9-5     John
        9-140
        0
        9-0
        9-4         0
        0
        Jane
9-6     0           9-7
9-7     Mary
        Joe
        9-150
        9-1
        9-160
        9-6         0
```

*9-100 = parents of Peter
9-110 = parents of Alice
9-120 = location of an offspring of Alice
9-130 = location of an offspring of Alice
9-140 = location of list of offspring of John and Jane
9-150 = location of list of offspring of Mary and Joe
9-160 = location of parents of Joe

* Local names of the form 9-XXX are equivalent in both T0 structures.

If the sentence, "John is Mary's brother" is processed, M20 will hold the elementary relation John-brother-Mary; both are residents of the tree, and the relation is brother, hence R2 is executed. This routine will combine the family units for the parents of John and the parents of Mary into one unit. The resulting tree will be as follows:

T0	0	
	9-0	0
9-0	0	
	Peter	
	Alice	
	9-1	
	9-100	
	9-110	
	9-0	0
9-1	0	
	9-2	
	9-4	
	9-120	
	9-130	0
9-2	0	
	Jane	9-3
9-3	John	
	9-140	
	0	
	9-0	
	9-2	0
9-4	0	9-5
9-5	Mary	
	Joe	
	9-150	
	9-0	
	9-160	
	9-4	0

In addition to the implications of brotherhood which are captured by the revised map, it is to be noted that the new map also carries the implication that Peter and Alice are married.

Efficiency versus unrestrictedness

If a map is designed to take full advantage of the reduced storage requirements for storing input expressions with explicit and implicit meaning, it may not be an unrestricted map. That is, some sequences of input expressions may not be allowed. In Chapter Two we examined an example of a map for the relationship "greater than". The most efficient map for storing this information is the single straight-line form. However, to make the map unrestricted we found it necessary to replace this form with a less efficient form. We may now examine the question of efficient versus unrestricted maps for family relations.

Implicit meaning

Minimum use of space can only be realized when all members of a set of input expressions with implicit meaning are presented before any of the implications are presented. If communications are received in any other order, space is wasted. We have avoided a true waste in the storage system used by the SAM program by allowing erasures -- returning to a common stockpile all space which is no longer needed. For example, erasures are made when the following expressions occur in the given order: John is Joe's brother; Peter is Al's brother; Alice is John's mother; George is Al's father; Joe is Peter's brother. Before the last expression, Alice and George each have their own family unit. With appearance of the last expression R2 is executed, since both Joe and Peter have been mentioned before. The family units of Alice and George are combined into a common list. If the relation of Joe to Peter had been given before mention was made of both Alice and George, it would have not been necessary to temporarily use the subsequently erased space.

Connotative meaning

When an expression or set of expressions has connotative and definite meaning, similar problems of efficiency may be encountered. For example, if we wished to add the word "aunt" to the acceptable vocabulary, the present system would need revision. Consider the set of expressions: Polly is the aunt of Jane; Alice is Jane's mother; George is Jane's father; Sam is George's brother;

June is Polly's daughter. This set has the implicit meaning that Alice is June's aunt, among other things. The set also has connotative meaning, since the probability that Polly is Alice's sister, the probability that Polly is George's sister, the probability that Polly is Alice's sister-in-law, and the probability that Polly is George's sister-in-law all become 1/2, while the probabilities of all other relations between Polly and George and Alice become 0. The position for Polly's family unit is not fixed definitely, yet some definite implications are made. The present SAM system does not adequately handle these situations.

Interaction among models

If it were known in advance exactly what information would be required, a reading machine of maximum efficiency could be created. This machine would look for and store only the selected information and would ignore all else. It is the inaccuracy of expectations which places a larger burden upon the storage system and forces the consideration of efficient storage schemes such as models.

In considering the reading situation we are dealing with a great over-burdening of computing capacity of computer or human. A passage of text may contain much more information and many more implications than can be handled completely. Therefore some selection is necessary. First, selection of which maps will be constructed, and second, selection as to how much information should be stored in one map and how much in another.

The SAD SAM combination illustrates a factorization of the reading problem into only two information storage models, yet even here some important questions of processing efficiency are raised. For example, cross-referencing of nouns is done partly in the syntactic model and partly in the semantic model: relative pronouns are handled in the former, while "to be" is handled in the latter. The decision as to which procedure is more efficient depends upon the use to which the system will be put. If information concerning other relations is to be called for, and if the models for the other relations also make use of the cross-referencing of "to be", it would be more efficient to have this done in the syntactic analysis. If, as is the case with the present system, this

cross-referencing is not needed elsewhere, it is not necessary that it be performed in the syntax program.

The problem is, however, complicated by the fact that a function may be more readily performed as part of one model than as part of another. This could lead to serious processing inefficiencies due to the factorization of the task. For example, the formation of the list of noun phrases, M0, could more readily be performed by the syntax program than by the semantic program. This is because the retrieving of this information from the syntactic map is made difficult by the flexibility of the format of this storage. But a judgment as to which is more efficient again depends upon the use to which the storage system will be put. If there were no need for this list, the syntax program would be unduly encumbered by requiring it to construct such a list for every sentence.

The situation is closely related to the phenomenon of "set", particularly as influenced by instructions given to an experimental subject. A technique for developing expectations to be used in efficiently factoring the problem would be an important addition to the present system.

Summary

SAM constructs a family tree from the map C0 through interpretation of the words father, mother, sister, brother, and offspring. That is, with each of these words is associated a program; these programs are annexing processes for the semantic map. The explicit relations from which the family tree is constructed may be presented in a variety of ways. The name of one of the individuals involved must occur as the object of the preposition "of" whose referent is one of the relation words; or the name must be used as an adjective modifying one of the relation words, as results when the phrase "X's brother" occurs in the sentence. The name of the other individual must be equivalent to the relation word which is the referent of the "of" phrase, or it must be equivalent to the relation word modified by the possessive noun. The equivalence may have resulted from the

use of a relative pronoun, a phrase in apposition, or the verb "to be"; or it may result from a combination of these constructions. An example of the latter case is the phrase "John, who is the brother of Mary". "Brother" is marked as equivalent to "who" by way of the verb "is"; "who" is marked as equivalent to "John" by way of the relative pronoun routine. The result is the equivalence of John and brother.

CHAPTER FIVE

Introduction

In the foregoing chapters we have discussed reading and understanding as processes of efficient storage and retrieval of information. Meaning was defined in terms of the semantic redundancy of the language; that is, not in terms of the statistical properties of the message symbols, but in terms of the implications of the messages. The present chapter discusses this view of these problems with regard to two theoretical qualities. First, we will discuss the generality of the theoretical position by pointing out other cognitive processes to which it is applicable and by elaborating upon another example at a descriptive level. Second, we will indicate some techniques for verification of the theory, pointing out some areas where future research will find the theories testable.

Further examples of the model concept

One form of efficient encoding has been briefly discussed above in conjunction with the geometry theorem proving machine of Gelernter. The diagram associated with the theorem proved to be a great aid to the theorem proving process by providing some "automatic" deductions from the premises. Such information storage techniques provide a basis for understanding in a wide class of reasoning tasks and are not limited to formal theorem proving. The memorizing of a set of directions for getting from one geographical location to another may be greatly simplified by constructing a "mental picture" of the terrain rather than memorizing each step by rote. Indeed, all problems of complex coordination and locomotion may be cast in the framework of such visual models.

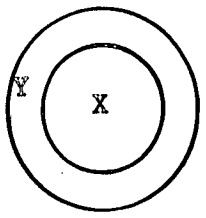
The visual model illustrates a second aspect of efficiency attained by such techniques. Since many tasks may be placed in the visual model frame, we may consider general annexing and retrieving processes which apply to all visual maps, and hence may be used for a variety of special tasks. To illustrate the generality of visual models and the efficiencies which are realized by such partitioning of the information storage process, we will consider in some detail an area of psychological interest which initiated our discussion. The problem of syllogistic reasoning, as explored by Woodworth and Sells (1935), will be discussed.

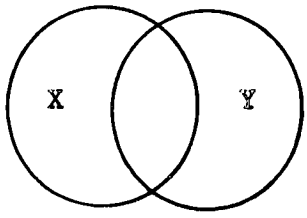
The input expressions (premises) presented to the subjects in the Woodworth and Sells experiments were of the following forms:

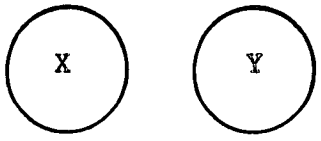
i All X are Y
ii Some X are Y
iii No X are Y
iv All X are not Y
v Some X are not Y
vi No X are not Y

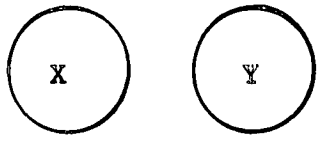
After having been given two such expressions, the subjects were asked, in one way or another, to decide upon the validity of other input expressions (conclusions). The task of memorizing the premises of all possible syllogisms together with the valid conclusions of each would be formidable. The subject is faced with the problem of encoding the problem in such a fashion that it can be handled with limited processing ability.

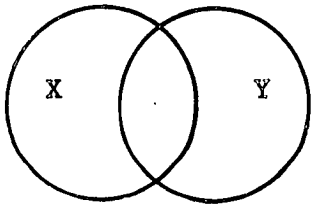
A very useful aid in working with such problems is the use of Venn diagrams. These are simple drawings which "represent" or store the information contained in the input expressions. The Venn diagrams for the six types of expressions are as follows:

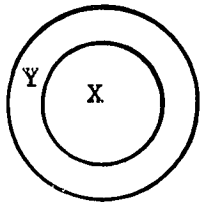
i 

ii 

iii 

iv 

v 

vi 

Syllogistic reasoning may be mediated by combining the basic diagrams of the two premises according to the following rules:

A. For each object mentioned (the X and Y of the premise), draw one circle. The area within the circle corresponds to the object (for example, X) and the area outside of the circle corresponds to the negation of the object (for example, not X).

B. The circles for the two objects mentioned in a premise are positioned according to the corresponding Venn diagram for that premise.

C. The relationship of a new circle to a circle already present but not mentioned in the premise is determined by possible arrangements which are allowed geometrically.

D. If more than one relationship is possible, make separate diagrams for each.

These rules define the annexing process for this model. Note that this process is defined for all orders of presentation of the premises.

The retrieving process is defined as follows:

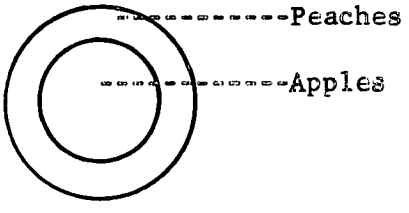
A. If the circle for one object, X, lies wholly within that of a second, Y, for all the diagrams, conclude that All X are Y and that No X are not Y.

B. If the circle for one object, X, intersects the circle for another object, Y, in one diagram, and if the circle for the first object does not lie wholly outside of the circle for the second object in any diagram, conclude that Some X is Y.

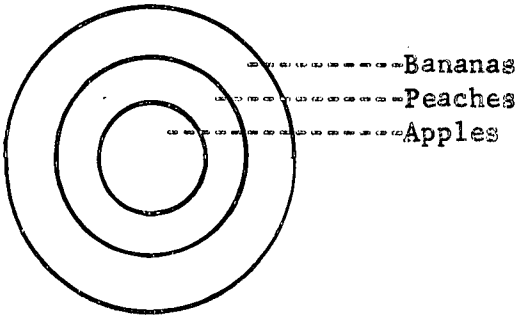
C. If the circle for one object, X, intersects the circle for a second object, Y. in some diagrams, and if it does not lie wholly within the circle for the second object in any of the diagrams, conclude that Some X is not Y.

D. If the circle for one object, X, lies wholly outside of the circle for a second object, Y, in some diagrams, and it neither intersects nor lies wholly within in any diagrams, conclude that All X is not Y and that No X is Y.

As an example of the use of this model, consider the premise All Apples are peaches. This would be represented as



Adding the premise All peaches are bananas would yield



From this diagram the retrieving process would yield, in addition to the premises, No apples are not peaches, No peaches are not bananas, All apples are bananas, and No apples are not bananas.

It should be noted that Venn diagrams do not make use of all the properties of a Euclidian plane (e.g., the metric properties). Perhaps other models, employing alternate storage techniques, would make more efficient use of memory space. Venn diagrams are valuable, however, because they make use of well-developed annexing and retrieving processes which adult subjects have at their command. By using visual maps where ever possible it is unnecessary to develop and store input-output processes specific to each task. Thus, overall efficiency of memory allocation may be improved, even though each map may be inefficient.

As it stands, such a technique for handling syllogistic reasoning is too powerful to be a theory of the behavior of untrained subjects. A theory which is of much greater accuracy is the theory of the atmosphere effect as put forth by Woodworth and Sells. Cast in the terms of the model concept, the atmosphere theory would have a very simple map requiring only two memory cells and four symbols. The annexing process might be outlined as follows. Initially, fill the two cells with the symbols "positive" and "universal." Whenever the word no or not is encountered, place the symbol "negative" in the first memory cell, erasing the previous contents. Whenever the word some is encountered, place the symbol "particular" in the second memory cell, erasing the previous contents. The retrieving process would then be as follows. If the memory cells contain the symbols "positive" and "universal," mark valid all statements of the form All X are Y. If the memory cells contain the symbols "positive" and "particular,"

mark valid all statements of the form Some X are Y. If the memory cells contain the symbols "negative" and "universal," mark valid all statements of the forms All X are not Y, No X are Y, and No X are not Y. If the memory cells contain the symbols "negative" and "particular," mark valid all statements of the form Some X are not Y.

A theory of syllogistic learning may now be more precisely phrased. We need to account for the transition from the state where input expressions create no map at all to the state where they create the atmosphere map, and then to the state where they create the visual Venn map. The problem is more precisely defined in the sense that the steps along the way may be precisely specified, thus allowing the precise definition of the transition (learning) processes.

If visual models are as ubiquitous as we propose here, a fundamental problem of cognitive theory is the specification of the means of storing such maps in the head. In the light of present knowledge, it seems unlikely that this is accomplished by the activation of patterns of neurons which are isomorphic to the drawings one would make on a blackboard: mental pictures are not etched in the head. The specification of a sufficient list structure storage system for visual maps would provide a valuable link between the actual picture and the associative connections of the brain.

Verification of the theories

The computer programs described in this paper are characterized by their great detail of specification as compared to other forms of psychological theorizing. In principle they allow exacting predictions of the performance of individual subjects. This fact, of course, does not preclude the possibility of discovering, through Monte-Carlo simulation, the statistical properties of the theories. However, a more fruitful approach would make use of the predictive facilities for individual subjects and specific tasks. Several of these possibilities will now be briefly outlined.

Order of words read

A sentence is scanned by the SAD SAM program in a left-to-right fashion. When, however, certain improbable sequences are encountered, the program may skip forward one or more words before making a decision, or may return to an earlier word and rebuild from there. The exact sequence of words examined by the program is quite easily determined.

Comparable data for human performances can best be gathered by use of an eye-movement camera. Such data would bear upon several aspects of the theory. Shifts from the normal left-to-right sequence would, under the interpretation of the theory, indicate difficult points in the sentence analysis. It could be readily determined whether or not these points correspond to the hypothesized difficulties. Additions to the list of (subjectively) improbable sequences could also be determined. By this process, very specific weaknesses could be located.

The reversals of the programmed scanning sequence are now relatively independent of their particular causes: the program always reverts to certain anchor points in the sequence and rebuilds from there. Eye-movement data may reveal variation in length of reversal, as well as indicate specific places for beginning the reconstruction.

These data may also shed light upon the relationship of the syntactic to the semantic aspects. After locating a sentence which produces reversals or hesitations in the scanning by a subject, other sentences composed of identical part-of-speech sequences, but differing in the meanings of the words, would be presented to the same subject. Differences in the scanning sequence for these syntactically identical sentences would be due either to errors in the syntactic categorizations or differences in word meanings. Thus the problem of determining the effect of semantic aspects upon the sentence parsing could be attacked.

Relative reading rates

The part-of-speech programs differ in complexity from category to category. Eye-movement data, as well as overall reading rates, would

measure the accuracy of the theoretically defined differences in difficulty. For example, articles should be quickly scanned, while nouns should require more time. In this fashion, the partitioning of the processing between various words could be verified.

In addition to variations in complexity, a given part-of-speech program may involve more or less processing depending upon where it is used. For example, a noun may initiate only a few operations if it occurs at the beginning of a sentence, whereas if it lies on the border between clauses, or in a sequence of nouns forming a compound noun phrase, or in an improbable sequence of nouns and adjectives, the processing may be quite involved. These differences should show up as increased delays in eye-movements and reading rates.

Analysis of errors

Both sentence diagrams and semantic maps constructed by the program are created in detail. The SAD program does more than locate main verb and subject. Every word is placed in relationship to every other word. This provides ample possibility for comparison with sentence diagrams produced by subjects with various degrees of experience with English. Gross measures of numbers of errors are of course also possible, but comparisons of clauses and individual phrases would also yield additional sources of verification.

The analysis of errors could also lead to more sophisticated comparisons. By imposing restrictions upon the processing powers of the program, the number and types of errors will be altered. This would provide one method for adjusting the theory to coincide with data. Such adjustments will also affect other aspects of performance, such as scanning, so that non-trivial tests of the theory can be made. The constraints upon processing powers may further be chosen in terms of their psychological reasonableness. For example, the complexity of substructure allowed and the number of working cells may readily be altered in the present system.

Errors arise in the performance of SAD if the working storage is not sufficient to handle the sentence complexity. There are two ways in which the working storage may be overtaxed. First, if a second instance of a given structure is encountered before the requirements of the first instance are satisfied, the program will never be able to complete the structure first begun since its "memory" of that structure will have been erased. Second, a structure for which SAD does not have built-in working storage provision may be encountered, in which case the program will not be able to recognize words as belonging to one structure and will assign the words to different structures. These errors should be readily recognized in human subjects.

Other tasks

The behavior of subjects working with problems related to but not identical with the reading situation can also be used as data with which to verify the theory. For example, since SAD SAM actually uses only limited information about each word, subjects could be supplied with only this same information rather than with words. A sequence of part-of-speech symbols, presented one at a time in order, would allow the subject to generate sentence diagrams without knowledge of word meanings and without finer syntactic discriminations. A comparison of human performance on this task with human performance in the complete situation would aid in determining the extent to which parsing is controlled by the missing information. In addition, data would be obtained for testing the theoretically most often accepted analyses in situations of ambiguity. Finally, requesting the subjects to construct partial diagrams as the sequence is presented, and allowing them to modify these after new symbols are added, would allow comparisons with the sub-structures developed by SAD in its working storage.

Similarly, changes in the task would provide additional information about semantic interpretation. SAM ignores much of the semantic information contained in a sentence when it derives a semantic map. By eliminating words, or substituting part-of-speech symbols for some words, the validity of the selective perception of SAM could be tested. Word order is also extensively used by SAM in determining relationships. By altering word order of test sentences, measures of the value of this variable could be obtained from human subjects.

Summary

We have indicated the general value of the type of theory discussed in this paper, with special reference to the specific programs presented earlier. The model concept appears to be of wide use in analyzing cognitive activity. Furthermore, the detail of the results generated by the simulations allows many opportunities for verification of the hypothesized processing. In addition to suggested extensions of the project, several specific recommendations for psychological experimentation have been suggested.

In view of the great complexity of language and verbal behavior, as indicated by Chomsky (1956), Penfield and Roberts (1959), Brown (1958), and others, the problems of meaning and understanding may prove to be unavoidable obstacles to the construction of machines which are capable of handling mechanical language translation, abstracting of technical articles, and library search. The study of simulation techniques in conjunction with cognitive psychology provides a means for attacking these obstacles.

References

Abelson, R. P., & Rosenberg, M. J. Symbolic psycho-logic: a model of attitudinal cognition. Behav. Sci., 1958, 3, 1-13.

Allport, G. W., & Postman. L. L. The basic psychology or rumor. In Transactions of the New York Academy of Sciences, Series II, 1945, 8, 61-81.

Bartlett, F. C. Remembering. Cambridge, England: Cambridge University Press, 1932.

Beebe-Center, J. G., Rogers, M. S., & O'Connell, D. N. Transmission of information about sucrose and saline solutions through the sense of taste. J. Psychol., 1955, 39, 157-160.

Bobbitt, J. M. An experimental study of the phenomenon of closure as a threshold function. J. exp. Psychol., 1942, 30, 273-294.

Brower, D., & Abt, L. E. Progress in clinical psychology (Vol. I.) New York: Grune & Stratton, 1952.

Bruner, J. S., Goodenough, J., & Austin, G. A. A study of thinking. New York: Wiley, 1956.

Bruner, J. S., Postman L., & Rodrigues, J. Expectation and the perception of color. Amer. J. Psychol., 1951, 64, 216-227.

Cartwright, D., & Harary, F. Structural balance: a generalization of Heider's theory. Psychol. Rev., 1956, 63, 277-293.

Chomsky, A. N. Three models for the description of language. IRE Trans. inform. Theor. IT-2, No. 3, September 1956.

Cofer, C. N., & Foley, J. P. Mediated generalization and the interpretation of verbal behavior: I. Prolegomena. Psychol. Rev., 1942, 49, 513-540.

De Groot, A. D. The thought processes of the chessplayer. Unpublished partial translation from Het Denken Van De Shaker, Amsterdam, 1946.

Dingman, H. F. Factor analysis of eye-movements with reading comprehension scores. Percept. mot. Skills., 1958, 8, 37-38.

Diven, K. Certain determinants in the conditioning of anxiety reactions. J. Psychol., 1937, 3, 291-308.

Feigenbaum, E. A. An information processing theory of verbal learning. Santa Monica, California: The RAND Corporation, no. P-1817, 1959.

Garner, W. R. An informational analysis of absolute judgments of loudness. J. exp. Psychol., 1953, 46, 373-380.

Gelernter, H. L. Realization of a Geometry Theorem Proving Machine. Proceedings of the UNESCO Conference on Information Processing, 1959.

Gibson, J. J. The reproduction of visually perceived forms. J. exp. Psychol., 1929, 12, 1-39.

Goldstein, K., & Scheerer, M. Abstract and concrete behavior: an experimental study with special tests. Psychol. Monogr., 1941, no. 2. (Whole No. 239).

Hake, H. W., & Garner, W. R. The effect of presenting various numbers of discrete steps on scale reading accuracy. J. exp. Psychol., 1951, 42, 358-366.

Harris, Zellig. A computer program for syntactic analysis. Report No. 14 to The National Science Foundation, 1959.

Heidbreder, E. The attainment of concepts. I. Terminology and methodology. II. The problem. J. gen. Psychol., 1946, 35, 173-189, 191-223.

Heidbreder, E. The attainment of concepts. III. The process. J. Psychol., 1947, 24, 93-138.

Heider, F. Attitudes and cognitive organization. J. Psychol., 1946, 21, 107-112.

Hull, C. L. Quantitative aspects of the evolution of concepts, an experimental study. Psychol. Monogr., 1920, 28, no. 1 (Whole No. 123).

Humphry, G. Thinking: an introduction to its experimental psychology. New York: Wiley, 1951.

Jung, C. G. Studies in word-association. London: William Heinemann, 1918.

Kent, G. H., & Rosanoff, A. J. A study of association in insanity. Amer. J. Insanity, 1910, 67, 37-96.

Komachiya, M. A note on concept attainment. J. Psychol., 1957, 43, 261-263.

Laycock, F. Flexibility in reading rate and Einstellung. Percept. mot. Skills, 1958, 8, 123-129.

Levine, J. M., & Murphy, G. The learning and forgetting of controversial material. J. abnorm. soc. Psychol., 1943, 38, 507-517.

Lindsay, R. K., & Willis, R. Discriminative reaction time of human subjects. Unpublished manuscript, Grad. Sch. Ind. Admin. Carnegie Inst. Techn., Complex Inform. Proc. paper no. 18., 1959.

Luchins, A. S. Mechanization in problem-solving: The effect of Einstellung. Psychol. Monogr., 1942, 54, no. 6 (Whole No. 248).

McGeoch, J. A., & Irion, A. L. The psychology of human learning. (2nd ed.), New York: Longmans, Green, 1952.

Maier, N. R. F. Reasoning in humans. I. On direction. J. comp. Psychol., 1930, 10, 115-143.

Maier, N. R. F. Reasoning in humans. II. The solution of a problem and its appearance in consciousness. J. comp. Psychol., 1931, 12, 181-194.

Mandelbrot, B. On recurrent noise limiting coding. In Proceedings of the symposium on information networks. Ann Arbor, Michigan: Edwards Brothers, Inc., 1955.

March, J. G., & Simon, H. A. Organizations. New York: John Wiley and Sons, 1958.

Miller, G. A. The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychol. Rev., 1956, 63, 81-97.

Mowatt, M. H. Configurational properties considered 'good' by naive subjects. Amer. J. Psychol., 1940, 53, 46-69.

Newell, A., Tonge, F. M., Feigenbaum, E. A., Mealy, G. H., Saber, N., Green, B. F., & Wolf, Alice K. Information processing language V manual. Santa Monica, California: The RAND Corporation, no. P-1897, 1960.

Newell, A., Shaw, J. C., & Simon, H. A. Empirical explorations of the logic theory machine. Proceedings of the Western Joint Computer Conference, IRE, February, 1957.

Ogden, C. K. Basic English, a general introduction with rules and grammar. 4th ed., London, Kegan Paul, Trench, Trubner & Co. Ltd., 1933.

Ogden, C. K. The basic words. London, Kegan Paul, Trench, Trubner & Co. Ltd., 1942.

Osgood, C. E., & Tannenbaum, P. H. The principal of congruity in the prediction of attitude change. Psychol. Rev., 1955, 62, 42-55.

Osterweil, J., & Fiske, D. W. Intra-individual variability in sentence completion responses. J. abnorm. soc. Psychol., 1956, 52, 195-199.

Phillips, A. V. A question-answering routine. Cambridge, Mass.: Mass. Inst. Technol. Comput. Center. Artificial Intell. Project memo 16, 1960.

Piaget, J. The child's conception of physical causality. New York: Harcourt, Brace, 1930.

Piaget, J. The language and thought of the child. New York: Harcourt, Brace, 1932.

Pollack, I. The information of elementary auditory displays. J. Acoust. Soc. Amer., 1952, 24, 745-749.

Pollack, I. The information of elementary auditory displays. II. J. Acoust. Soc. Amer., 1953, 25, 765-769. (a)

Pollack, I. The assimilation of sequentially encoded information. Amer. J. Psychol., 1953, 66, 421-435. (b)

Razran, G. H. S. Salivating and thinking in different languages. J. Psychol., 1935, 1, 145-151.

Razran, G. H. S. A quantitative study of meaning by a conditioned salivary technique (semantic conditioning). Science, 1939, 90, 89-90.

Roberts, K. E. The ability of preschool children to solve problems in which a simple principle of relationship is kept constant. J. genet. Psychol., 1932, 40, 118-135.

Rosenberg, M. Cognitive structure and attitudinal affect. J. abnorm. soc. Psychol., 1956, 53, 367-372.

Sager, Naomi. Procedure for left-to-right recognition of sentence structure. Unpublished manuscript. U. of Penna. Transformations and discourse analysis project paper no. 27, 1960.

Shannon, C. E. A mathematical theory of communication. Bell Syst. tech. J., 1948, 27, 379-423, 623-656.

Smoke, K. L. An objective study of concept formation. Psychol. Monogr., 1932, 42, no. 4 (Whole No. 191).

Smoke, K. L. The experimental approach to concept learning. Psychol. Rev., 1935, 42, 274-279.

Vinacke, W. E. The investigation of concept formation. Psychol. Bull., 1951, 48, 1-31.

Wang, H. Toward mechanical mathematics. IBM J. Res. Develpm., 1960, 4, No. 1, 2-32.

Wertheimer, M. Principles of perceptual organization. In Beardslee, D. C., & Wertheimer, M. (eds.) Readings in perception. Princeton, N. J.: D. Van Nostrand, 1958, p. 115-135.

Woodworth, R. S., & Sells, S. B. An atmosphere effect in formal syllogistic reasoning. J. exp. Psychol., 1935, 18, 451-460.

Yngve, V. H. A model and an hypothesis for language structure. Proc. Amer. phil. Soc., 1960, 104, No. 5.

Zipf, G. K. The repetition of words, time-perspective, and semantic balance. J. gen. Psychol., 1945, 32, 127-148.