INTEGRATED SYSTEMS FOR NATURAL LANGUAGE PROCESSING

TECHNICAL REPORT NO. NL-8

R. F. Simmons

August 1972

NATURAL LANGUAGE RESEARCH FOR COMPUTER-ASSISTED INSTRUCTION

Supported by:

THE NATIONAL SCIENCE FOUNDATION
Grant GJ 509 X

Department of Computer Sciences

and

Computer-Assisted Instruction Laboratory

The University of Texas
Austin, Texas

ABSTRACT

This report summarizes the discussion of the final meeting of the Conference on Research Trends in Computational Linguistics, held in Washington D. C. March 14-16 1972. A brief background survey of the state of the art in natural language processing and of computational linguistics is then presented and a series of areas for recommended research and development are outlined.

To appear in: Friedman, Joyce and Hood Roberts (eds) Proceedings on Research Trends in Computational Linguistics. (In Preparation).

REPORT OF THE DISCUSSION OF INTEGRATED SYSTEMS FOR NATURAL LANGUAGE PROCESSING

Most recent and notable among integrated systems for language processing are Woods' Natural Language Retrieval System and Winograd's system for understanding commands to a robot for operations on a limited world of colored blocks. Other examples of more or less successful integrated language processing systems include; mechanical translation, natural language data manipulation, robot command systems, natural language CAI, semantic analysis of medical data, the analysis, synthesis and testing of linguistic rules, text analysis, paraphrase and question answering, and text generation. The current ARPA sponsored research toward useful speech recognition systems is the most recent large scale approach to integrate knowledge of semantics, syntax and phonology into a capability for computer understanding of a useful spoken vocabulary. From this inventory of examples we can infer that an integrated language processing system is a processor that organizes a set of component linguistic processes into an effective and potentially practical language processing device. The components may include speech analysis, and synthesis, syntactic, semantic and logical operations.

One aspect of integrated language processing can arbitrarily be dimensionalized as the vertical inter-relation of morphological, syntactic, semantic and pragmatic information. On this dimension interaction among levels is used to resolve ambiguities and to clarify understanding. The most thorough example currently available as a

working program appears to be Winograd's system that uses syntactic semantic and pragmatic information to resolve possible ambiguities of English commands and questions. Problems associated with integration on the vertical dimension include the fact that the complexity of such integrated systems is such that a single person can hardly hold all aspects of interaction in his mind at one time. This level of complexity requires the use of high level languages such as LISP, PLANNER, etc. with the consequence that computation is fairly slow and fast random access memory requirements are quite large. Alternatively, such a system may be built as a team effort, with its attendant communication problems among team members and subprograms. The requirement for careful documentation of such systems was emphasized and suggestions emerged for the use and development of text editing and automated documentation-program packages. Automated flow-charting programs were suggested for application to FORTRAN, ALGOL, and PL1 type programs, and systems for displaying the calling structure of functions and other organizational structures of LISP systems were briefly described.

A second aspect of integrated systems can arbitrarily be dimensionalized as horizontal. On this dimension problems occur at the interface of a given language processing system with the user and with other systems. Careful human engineering of the user interface is required to enable the convenient insertion of lexical entries, grammar rules and semantic and pragmatic information as well as the testing of these data for consistency and accuracy. The need for fast on-line interactive consoles is most obvious here. Problems concerned with evaluation of the effectiveness of language processing systems and

their generalizability emerge when considering horizontal integration. These areas are suggested for continued research.

A contrast emerged between what is currently desirable and what is minimally necessary in the way of hardware for research and development of integrated language processing systems. Ideally, something like the following configuration of hardware is desired:

INPUT	CENTRAL PROCESSOR	OUTPUT
Character Reader	Core Memory ¼-1 million words	Print
Remote Quiet Terminals	Extended Core 1 million words	Voice
Voice Input	Disc 10 million words	Graphics
Graphics	Networked Miniprocessors	

numbers signify order of magnitude only

Software support would include a multi-access timeshared system, basic languages such as FORTRAN, ALGOL, PL1, LISP and SNOBALL. Specialized text editing, parsing, generation and question answering program systems would serve as a computational library for the system.

In contrast to the ideal system to support integrated language processors, the minimum requirements can already be found at a number of locations around the country in existing computing and communication networks. The M.I.T. AI laboratory, Bolt Beranek & Newman, Stanford, Carnegie, University of Texas, S.R.I., and the ARPA computing research network are all well known examples of computing facilities adequate to the purpose. Unfortunately, adequate access to such information utilities is administratively limited and very expensive.

A significant question was raised about hardware requirements which led the discussants to conclude that present day hardware systems are theoretically adequate to support research in computational linguistics but, in fact, the research user typically commands less

computer utility in terms of central processors and memory than he did in the mid sixties. The reason is that despite the truly impressive increases in size of memory and processor efficiency, the computing utility is shared much more widely, usually reducing the amount of utility that each person can command. The consequence appears to be that even in the best computation laboratories, computational linguists are hard put to obtain sufficient computing power to construct large integrated systems. For the moment, this lack does not block research; but it does bias the activity toward the development of component software rather than large integrated systems which require more computation utility than is readily available to the researcher.

The suggestion was made that Computational Linguistics research might follow the path customarily used by atomic physicists in concentrating expensive equipment at a few centers. The ARPA network is a good example of this approach in the related field of artificial intelligence research, and it might well be recommended for our own attempts to develop practically useful integrated language processing systems.

Hallway Discussions: The foregoing paragraphs represent the gist of the morning's discussion on Integrated Systems in Computational Linguistics. Behind this discussion lay a considerable amount of common knowledge of the current state of the art, the research problems and the implications of integrated systems for the future of the discipline. In this section an attempt is made to make some of this implicit background explicit to the reader. The effort will of course be significantly biased by the author's perception of the situation.

First, it appears that some significant changes in theoretical approaches to linguistics have been occurring over the past two or three years. It was previously the fashion to look almost exclusively to the transformationalists for a theoretical basis of computational linguistics. Such syntactic analysis systems as those reported by Petrick in a previous chapter showed a majority bias toward transformational deep structure analysis or approximations thereto; but what I noticed particularly was that some of the mature projects such as those reported from Europe, the still-healthy mechanical translation project of Lehmann and Stachowitz (1972), and numerous newer language processing systems around the country including those of Winograd (1972), Thompson (1964), Carbonnel (1970), Heidorn (1971), and Simmons (1972) had broken free of the transformational approach. Some systems such as those of Woods (1970) and Kellogg(1968) while originally based on transformational theory had clearly evolved to the point that essentially only the basic principles of analysis into a deep syntactic structure and some use of selection restrictions and semantic markers remained to show their theoretical ancestry.

Placing these observations in the context of 1) the rapid development of case-structure theory, 2) the recent attention to Halliday's work (1970), and 3) the computational development by Woods (1970) and by Bobrow and Frazer (1969) of the Thorne, Bratley and Dewar (1968) use of finite state network approach to grammar, it is possible to see the emergence of a new theoretical framework for computational linguistics. As it appears to me, the most profitable theoretical approach to parsing today is somewhat eclectically based on the use of some form of process grammar represented either as

rule sets (see Heidorn), as augmented finite state nets (see Woods) or as procedures (see Winograd, Schank).

For structures to represent the underlying meaning of sentences, three approaches are quite prominent. To the extent that a phrase refers to some object in a data structure, it is quite valuable to follow the procedural semantics approach described most throughly by Woods (1968). In this approach the meaning of a phrase is the value returned by a procedure that uses elements of the phrase to identify a data object or set as its referent.

Winograd carried the procedural semantics approach even further by interpreting relational words such as verbs and prepositions as procedures in a robot command language. The MicroPlanner Interpreter of this language is sophisticated enough to accomplish quite significant inferences. Consider, for example, the command, "Put the red block in the blue box", which translates into a procedural expression such as, (PUT B1 B2). If the blue box, B2, already contains an object, the PUT function is able to call other functions to temporarily release the block, B1, remove the object from B2, then pick up B1 again and finally succeed in putting it in the box.

Another approach that is theoretically satisfying but computationally cumbersome is to represent the meaning of a sentence as a statement in symbolic logic. Such a statement is then taken as a theorem whose truth value with reference to a set of axioms that comprise the data base can be determined by use of resolution techniques. (See Green & Raphael,1968, Palme 1971, and Sandewall 1970).

A third approach --one that is applicable to text analysis-- is to resolve the sentence into its deep-case representation which

is conceptualized as an event description in terms of a verbal relation and its arguments. The meaning of this event description is a series of propositions implied by the event. (See Schank, Lehmann and Stachowitz, and Simmons).

Since the third approach is rather new, an example will best communicate it:

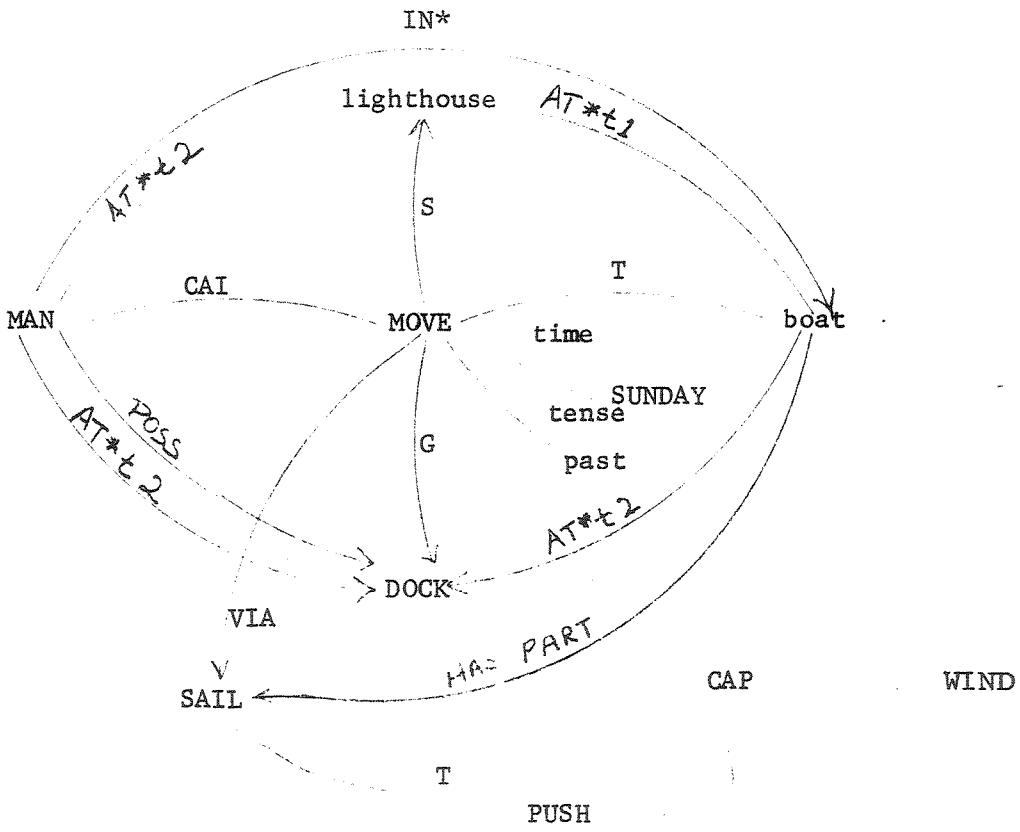
The man sailed a boat from the lighthouse to his dock.

A form of its deep case structure is as follows:

MOVE	via sail
CAUSAL ACTANT1	the man
THEME	boat
SOURCE	the lighthouse
GOAL	his dock

Some of the propositions implied by this structure can be seen in the diagram of Figure 1. Each node-arc-node in that diagram represents such a proposition. For example, "MAN Causal Actant of MOVE" shows that the man instigated the action; "MOVE THEME BOAT" shows the boat received the action; "MAN Possess DOCK" represents the meaning of "his dock"; "BOAT At, t1 LIGHTHOUSE" shows an initial state of the event; and "MAN At, t2 DOCK" shows that the position of the man at the end of the event was that of the dock. The other arcs can be interpreted in similar fashion. The entire set of such arcs provides a first level of inferential meaning for such purposes as question answering and problem solving.

These three approaches to representing the meaning of English sentences are quite obviously the current computational nominations for semantic deep structures underlying the surface form of sentences. Each has proved useful in some integrated system application. There is enough similarity among the approaches to suggest that they may in



Graph of Sentence Meaning

Figure 1

fact be different points of view and that a given semantic structure has procedural, logical and propositional descriptions.

The means for computing these underlying structures are remarkably similar at a certain level of abstraction. The surface structure of an English sentence is analyzed with the aid of a grammar, a lexicon and a parsing algorithm to produce noun phrase and verb phrase components. Either at the point of recognizing a phrase, or later, a series of operations are applied to transform each phrase into the desired form of semantic structure. Although syntactic analysis remains a difficult, indeterminate, recursive process, the clear definition of the desired semantic structure is used successfully to limit the indeterminacy of the process. The validity of the resulting semantic structure is measured by its usefulness in question answering, paraphrase, and inferential capabilities. Rationalist arguments about the exact form and time of application of transformational rules are essentially irrelevant in this empirical approach, and the definition of semantic deep structure is influenced most strongly by pragmatic considerations.

What we see in looking briefly at the development of integrated systems for understanding English is that computational linguistics is scientifically past its adolescence. While not being "school-bound", it is nevertheless deeply influenced by such syntactic theorists as Harris, Chomsky, Halliday, Thorne and others. It reflects semantic theories of Tarski, Carnap, Katz, Woods, and most recently Fillmore, Chafe and Grimes. Its methods clearly demonstrate the usefulness of computational theories of Thompson, Kuno and Oettinger, Woods and others. The field is also fortunate in that a group of psychologists are exploring the applicability of its case models and procedural

semantic structures as models of human memory processes. (See Frijda 1972, Collins and Quillian 1969, Norman 1972, and Frederikson 1972 among others). And Carroll (Chapter) in reporting another panel suggests other valuable relations between computational linguistics and psychology.

But post-adolescence is not maturity. Computational realizations of natural language processing capabilities are impressive in their ability to understand small subsets of natural language but in most cases a great deal of development is required to achieve practical applications.

Most existing systems are dealing only with single sentences and only the most recent few resolve some simple cases of pronominal and anaphoric reference. Wilks (1968) Heidorn (1971) and Su (1971) have studied paragraph or larger units of text, but these appear to be initial explorations of a very large area of computational linguistic research. Few of the systems have utilized peripheral storage to allow for growth of the large lexicons and grammars that are usually required for practical applications. The task of producing large, consistent lexicons and grammars has only occasionally been attempted. (See Woods 1970, Sager 1970, and Lehmann and Stachowitz for examples of such attempts.)

The capability to analyze unedited text -- essential for many mechanical translation and information retrieval applications -- is far out of reach at present, depending on the development of large computational lexicons and grammars for a large stable system with much use of peripheral storage. An area in which very little work has so far been accomplished is that of developing convenient means

for nonlinguists to communicate lexical and grammar structures to the system.

But applications of this post-adolescent field of computational linguistics are a most active area of research. Early work on stylistic and content analysis (See Sedelow, 1972, Stone 1962) has resulted in tools frequently used by behavioral scientists and humanities scholars for a wide variety of purposes. Another early development of keyword analysis of sentences is still a major tool for information retrieval applications. Several efforts at natural language data management (See Woods, Thompson and Kellogg for examples) have explicitly practical goals. A most recent applications area is that of Computer Assisted Instruction with studies by Wexler (1970), Carbonnel (1970), Brown (1972), Simmons (1972) and others. The attempt in this area is to use language processing and question answering technology to develop some approximation to a tutorial system that can understand a student's questions and statements, and answers to questions. The systems available at this writing are purely experimental -- sometimes impressive in the depth of their understanding of a few concepts -- but lacking both breadth of language processing ability and depth of peripheral memory-system engineering. Even more important, almost nothing has so far been learned on an experimental basis of how to use a language processor as part of an effective tutorial system. A pressing need is upon us to substitute experimental evidence for armchair theorizing about the educational use of natural language tutorial systems.

Recommended Research Emphases: In the foregoing, I have outlined the area of integrated systems research as central to the discipline

of computational linguistics which is based on syntactic and semantic theories of classical linguists and logicians and computational theories of its own. I have described it as an empirically oriented area with psychological implications for understanding human thought processes and one with potentially great -- but still unrealized -- applications to socially valuable systems. The meeting's discussion on integrated systems was rich in its suggestions for research and in its discussion of current difficulties in accomplishing it. Other areas requiring research become apparent in considering the state of the art.

I have attempted to summarize these suggestions in outline form. Each of the subtopics in the outline is an area where research or developmental attention is needed and can be expected to advance the state of the art.

1. Managing Complex Systems
    a. High level languages
    b. Team efforts
        1) communications among members
        2) communications among system components
    c. Documentation and Programming aids
        1) text and file editing
        2) automated flow-charting systems
        3) displays of structure of subsystems
2. Interfaces
    a. User
        1) convenient methods for inserting data
        2) testing, revising data
        3) evaluation of language processor
            a) choice of standard dimensions for evaluation
    b. Peripheral systems
        1) speech analysis

2) speech synthesis

3) peripheral memory devices

4) text scanners

3. Computing Utility

   a. Central repository of systems, data, algorithms

   b. Distant and local access to a very large computing system (or network).

4. Theoretical Research

   a. syntactic structures for computation

   b. semantics of case structures, procedures, etc.

   c. psychological studies of memory structures

   d. logical structure of sentence meanings

   e. speech

5. Computational Linguistic Research

   a. discourse structure

   b. anaphoric reference

   c. computational structure of the English Lexicon

   d. development of computational lexicon and grammar for large subsets of English (and other languages)

   e. computational structure of speech

6. Systems Research

   a. large language processing systems with indefinitely extendable peripheral storage capabilities

   b. list processing languages that can effectively use peripheral storage devices.

7. Applied Natural Language Systems

   a. computer assisted instruction

   b. text-based question answering

   c. Data-base question answering

   d. natural language systems for describing algorithms and processes to a computer, i.e. Natural Language Programming Systems

   e. systems for conversing with a user about a topic

   f. systems for generating essays from a data base

   g. simulations of personality and belief structures

8. Inferential Capabilities

   a. verbal problem solving

   b. theorem proving

   c. deductive question-answering

   d. paraphrase

9. Esoterica
   a. emotional content of language and of speech.
   b. interface with visual data.
   c. interface with auditory data
   d. stylistics, rhythm, rhyme etc.
   e. semantic classification systems.

REFERENCES

Bobrow, Daniel and Frazer, J. B. "An Augmented State Transition Network Analysis Procedure", Proceedings of 1st IJCAI, 1969, pp 303-316.

Brown, John S., Burton, R. B. and Zdybel, Frank "A Model Driven Question-Answering System for a CAI Environment", University of California, Irvine, Department of Information Science, Tech. Report #13, January 1972.

Carbonell, J. R. "AI in CAI: An Artificial Intelligence Approach to Computer Assisted Instruction", IEEE Transactions on Man-Machine Systems December 1970, Vol. MSS-11, No. 4, 190-202.

Carnap, R. Introduction to Semantics, Harvard University Press, Cambridge, Mass. 1946.

Chafe, W. L. Meaning and the Structure of Language, University of Chicago Press, Chicago 1970.

Collins, A. M. and Quillian, M. R., "Retrieval Times from Semantic Memory", Journal of Verbal Learning and Verbal Behavior, 1969, Vol. 8, pp 240-247.

Fillmore, C. J., The Case for Case, in Bach and Harms (see McCawley).

Frederiksen, Carl H., Representing Logico-Semantic Features of Written Knowledge Acquired from a Discourse, University of California, Berkeley, Department of Psychology (ms from the author). 1972.

Frijda, N. H. "The Simulation of Human Memory" Psychol. Bull., January 1972.

Green, Cordell and Raphael, B., "The Use of Theorem Proving Techniques in Question Answering Systems", Proc. ACM National Conference, 1968, pp 169-181.

Grimes, Joseph E. The Thread of Discourse, Cornell University, Ithaca, N. Y. (preprint from the author).

Halliday, N. A. K. "Functional Diversity in Language as seen from a consideration of Modality and Mood", Foundations of Language, Vol. 6, 1970, pp 322-361.

Heidorn, George E., Natural Language Inputs to a Simulation Programming System , Naval Postgraduate School, Monterey, California, December 1971.

Kellogg, C. H. "A Natural Language Compiler for on-line Data Management", AFIPS Conference Proceedings, Thompson Book Co. Vol. 33, 1968, FJCC.

References continued

Lehmann, W. and Stachowitz, R., Feasibility Study on Fully Automatic High Quality Translation. (mimeo) Ling. Res. Ctr., University of Texas, Austin 1972.

Norman, Donald A., "Memory, Knowledge and the Answering of Questions", Center for Human Information Processing, University of California, San Diego, La Jolla, California, May 1972.

Palme, J., "Making Computers Understand Natural Language", In Findler, N. and Meltzer, B. (Eds.) Artificial Intelligence and Heuristic Programming, Edinburgh University Press, Edinburgh, U. K. 1971.

Sager, Naomi. "An Application of Syntactic Analysis to Information Retrieval", String Program Report No. 6, NYU Linguistic String Program, New York, N. Y., April 1970.

Sandewall, E. J., "Representing Natural Language Information in the Predicate Calculus", Stanford University, Computer Science Department, Report #166, Palo Alto, 1970.

Schank, Roger, "Identification of Conceptualizations Underlying Natural Language", In Schank, R. and Colby, K. (Eds.) Computer Cognition, Freeman Press. (In Press).

Sedelow, S., and W. Sedelow., "Stylistic Analysis." In H. Borko (Ed.). Automated Language Processing, Wiley, New York, Ch. 6. (In press).

Simmons, R. F., "Linguistic Analysis of Constructed Student Responses in CAI", In Holtzman W. (Ed.)

Simmons, R F., Semantics Networks: Their Computation and Use for Understanding English Sentences, University of Texas, Department of Computer Sciences, Austin, Texas, May 1972.

Stone, P. J., Bayles, R. F., Namerwirth, J. Z., and Ogilvie, D. M. The general inquirer: a computer system for content analysis and retrieval based on the sentence as a unit of information. Behav. Sci., 7, 4 (1962), 1-15.

Su, Stanley Y. W. A Computational Model of Paragraph Production, Technical Report No. 71-102, Nov. 1971, Center for Information Research, University of Florida, Gainsville, Florida.

Tarski, A. "The Semantic Conception of Truth", Phil. and Phenom. Res., Vol. 4, 1944.

Thompson, F. B. Semantic counterpart of formal grammars. TEMPO General Electric Co., Santa Barbara, Calif.
---et. al. DEACON breadboard summary. RM64TMP-9, TEMPO General Electric Co., Santa Barbara, Calif., Mar. 1964.

References continued

Thorne, J., Brately, P., and Dewar H. "The Syntactic Analysis of English by Machine" In Michie, D. (Ed.) Machine Intelligence, 3, pp 281-309. 1968.

Wexler, J. D. "A Generative Teaching System that Uses Information Nets and Skeleton Patterns" Ph.D. Dissertation, Computer Sciences Department, University of Wisconsin, Madison 1970.

Wilks, Yorick. "Computable Semantic Derivations" System Dev. Corp. Sp3017, January 1968.

Winograd, T. A Program for Understanding Natural Language, Cognitive Psychology, 1972, 3, 1-191.

Woods, W. A. "Procedural Semantics for a Question Answering Machine" AFIPS Conference Proceedings: Thompson Book 6. Vol. 33, 1968, FJCC.

Woods, W. A. "Transition Network Grammars for Understanding Natural Languages" Comm ACM, Vol. 13 #10, October 1970.