American Journal of Computational Linguistics *Microfiche 21*

NEWSLETTER OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS

VOLUME 12 - NUMBER 3 JULY 1975

AMERICAN JOURNAL OF COMPUTATIONAL LINGUISTICS is published by the Center for Applied Linguistics for the Association for Computational Linguistics

EDITOR David G. Hays *Professor of Linguistics and of Computer Science State University of New York, Buffalo*

EDITORIAL STAFF Jeff F. Lesinski *Assistant Pro Tem*
Jacquin Brendle *Secretary*

EDITORIAL ADDRESS Twin Willows, Wanakah, New York 14075

MANAGING EDITOR A. Hood Roberts *Deputy Director, Center for Applied Linguistics*

MANAGEMENT STAFF Nancy Jokovich and David Hoffman

PRODUCTION AND SUBSCRIPTION ADDRESS 1611 North Kent Street, Arlington, Virginia 22209

American Journal of Computational Linguistics *Microfiche 21 : 2*

RECENT COMPUTER SCIENCE RESEARCH IN NATURAL LANGUAGE PROCESSING

ALLEN KLINGER
Computer Science Department
School of Engineering and Applied Science
University of California, Los Angeles

ABSTRACT

The machine translation problem has recently been replaced by much narrower goals and computer processing of language has become part of artificial intelligence (AI), speech recognition, and structural pattern recognition. These are each specialized computer science research fields with distinct objectives and assumptions. The narrower goals involve making it possible for a computer user to employ a near natural-language mode for problem-solving, information retrieval, and other applications. Natural computer responses have also been created and a special term, "understanding", has been used to describe the resulting computer-human dialogues. The purpose of this paper is to survey these recent developments to make the AI literature accessible to researchers mainly interested in computation on written text or spoken language.

1. INTRODUCTION

The computer literature discussed in this paper uses several linguistic terms in special ways, when there is a possibility of confusion, quotation marks will be used to identify technical terms in computer science. The term "understanding" is frequently used as a synonym for "the addition of logical relationships or semantics to syntactic processing". This use is substantially narrower than the word's implicit association with "human behavior implemented by computer" the narrower use is introduced as a neutral reference point. The question of whether a computer porgram can operate in a human-like way is central to artificial intelligence. "Do current 'understanding' program systems show how extended human-like capability can be implemented using computers?" is a related pragmatic question Initially this investigation sought to examine whether programs which "understand" language in the stipulated narrow sense are prototypes which could lead to expanded capability. Unfortunately, "language understanding" and its special subtopic "speeeh understanding" are insufficiently developed to permit profitable discussion of the original question Hence an operational approach to the recent literature is taken here. This paper outlines how "language understanding" research has evolved and identifies key elements of program organization used to achieve limited computer "understanding".

2. LEVEL AND DOMAIN

Current AI programs for language processing are organized by level and restricted to specified domains. This section presents those ideas and comments on the limitations that they entail.

Three principal levels of language-processing software are

1. "Lexical" (allowed vocabulary)
2. "Syntactic" (allowed phrases or sentences)
3 "Semantic" (allowed meanings)

In practice all these levels must operate many times for the computer to interpret even a small portion, say two words, of restricted natural-language input. Programs that perform operations on each level are, respectively,

1. Word in a table?
2. Word string acceptable grammatically?
3. Word string acceptable logically?

A program to detect "meaning" (logical consequences of word interpretations) must also perform grammatical operations for certain words to determine a part of speech (noun, verb, adjective, etc.) One method makes a tentative assignment, parses, then tests for plausibility via consistency with known facts. To reduce the complexity of this task, the designer limits the subset of language allower or the "world" (i.e. the subject) discussed. The word "domain" sums up this concept, other terms for "restricted domain" are "limited scope of discourse", "narrow problem domain", and "restricted English framework"

The limitation of vocabulary or context constrains the lexicon and semantics of the "language". The trend in the design of software for "natural-language understanding" is to deal with (a) a specialized vocabulary, and (b) a particular context or set of allowed interpretations in order to reduce processing time. Although computing results for several highly specialized problems [e.g. 7, 23] are impressive examples of language processing in restricted domains, they do not answer several key concerns.

1. Do specialized vocabularies have sufficient complexity to warrant comparison with true natural language?
2. Are current "understanding" programs, organized by level and using domain restriction, extendable to true natural language?

The realities are severe. Syntactic processing is interdependent with meaning and involves the allowed logical relationships among words in the lexicon. Most natural-language software is highly developed at the "syntactic" level However, the number of times the "syntactic" level must be entered can grow explosively as the "naturalness" of the language to be processed increases. Success on artificial domains cannot imply a great deal about processing truly natural language.

3. PROGRAM SYSTEMS

The systems cited in this section answer questions, perform commands, or conduct dialogues.

Programs that enable a user to execute a task via computer in an on-line mode are generally called "interactive" Some systems are so rich in their language-processing capability that they are called "conversational" Systems that have complicated capabilities and can reply with a sophisticated response to an inquiry are called "question answering". The survey [1] discusses two "conversational" programs ELIZA [2, 3] and STUDENT [4], which answers questions regarding algebraic word problems. SIR [5] answers questions about logic. Both [4] and [5] appear in [6], the introduction there provides a general discussion of "semantic information and computer programs involving "semantics"

The "question-answering" program systems described in [2-5] were sophisticated mainly in methods of solving a problem or determining a response to a statement. Other systems have emphasized the retrieval of facts encoded in English. The "blocks-world" system described in [7] contrasts with these in that it has sophisticated language-processing capability It infers antecedents of pronouns and resolves ambiguities in input word strings regarding blocks on a table. The distinction between "interactive", "conversational", and "question-answering" is less important when the blocks-world is the domain. The computer-science contribution is a program to interact with the domain as if it could "understand" the input, in the sense that it takes the proper action even when the input is somewhat ambiguous. To resolve ambiguities the program refers to existing relationships among the blocks.

The effect of [7] was to provide a sophisticated example of computer "understanding" which led to attempts to apply similar principles to speech inputs. (More detail on parallel developments in speech processing is presented later.)

The early "language-understanding" systems, BASEBALL [9], ELIZA, and STUDENT, were based on two special formats: one to represent the knowledge they store and one to find meaning in the English input. They discard all input information which cannot be transformed for internal storage. The comparison of ELIZA and STUDENT in [1] is with regard to the degree of "understanding" ELIZA responds either by transforming the input sentence (essentially mimicry) following isolation of a key word or by using a prestored content-free remark. STUDENT translates natural-language "descriptions of algebraic equations, ... proceeds to identify the unknowns involved and the relationships which hold between them, and (obtains and solves) a set of equations" [1, p 85]. Hence ELIZA "understands" only a few key words; it transforms these words via a sentence-reassembly rule, discards other parts of the sentence, and adds stock phrases to create the response. STUDENT solves the underlying algebraic problem--it "understands" in that it "answers questions based on information contained in the input" [4, p. 135]. ELIZA responds but does not "understand", since the reply has little to do with the information in the input sentence, but rather serves to keep the person in a dialogue.

Programs with an ability to spout back similar to ELIZA's usually store a body of text and an indexing scheme to it. This approach has obvious limitations and was replaced by systems that use a formal representation to store limited logical concepts associated with the text. One of them is SIR, which can deduce set relationships among objects described by natural language. SIR is designed to meet the requirement that "in addition to echoing, upon request, the facts it has been given, a machine which 'understands' must be able to recognize the logical implications of those facts. It also must be able to identify (from a large data store) facts which are relevant to a particular question" [5].

Limited-logic systems are important because they provide methods to represent complex facts encoded in English-language statements so that the facts can be used by computer programs or accessed by a person who did not input the original textual statement of the fact. Such a second user may employ a completely different form of language encoding. Programs of this sort include DEACON [10, 11] and the early version of CONVERSE [12]. The former could "handle time questions" and used

> a bottom-up analysis method which allowed questions to be nested. For example, the question "Who is the commander of the battalion at Fort Fubar?" was handled by first internally answering the question "What battalion is at Fort Fubar?" The answer was then substituted directly into the original question to make it "Who is the commander of the 69th battalion?" which the system then answered. [7, p. 37]

CONVERSE contained provisions for allwing even more complex forms of input questions (Recent versions are described in [13-15].)

Deductive systems can be divided into general systems which add a first-order predicate-calculus theorem-proving capability to limited-logic systems to improve the complexity of the facts they can "infer", and procedural systems which enable other computations to obtain complex information The theorem-proving capability is designed to work from a group of logical statements given as input (or statements consistent with these input statements) However, facts INCONSISTENT with the original statements cannot always be detected and deductive systems quickly become impractical as the number of input statements (elementary facts, axioms) becomes large [6, 7, 16], since the time to obtain a proof grows to an impractical length. Special programming languages (e.g. QA4 [17, 18], PLANNER [20, 21]), have added strategy capabilities and better methods of problem representation to reduce computing time to practical values

> QA4 (seeks) to develop natural, intuitive representations of problems and problem-solving programs. (The user can) blend ... procedural and declarative information that includes explicit instructions, intuitive advice, and semantic definitions. [17]

However, there is currently no body of evidence regarding the effectiveness of the programs written in this programming language or related ones on problem-solving tasks in general

or "language understanding" in particular. There is a need for experimental evaluation of the strategies that the programming language permits for "language understanding" problems.

Procedural deductive systems facilitate the augmentation of an existing store of complex information. Usually systems require a new set of subprograms to deal with new data:

> each change in a subprogram may affect more of the other subprograms. The structure grows more awkward and difficult to generalize. ... Finally, the system may become too unwieldy for further experimentation. [5, p. 91]

In procedural systems the software is somewhat modular In 19 "semantic primitives" were assumed to exist as LISP subroutines. PLANNER [20] allows complex information to be expressed as procedures without requiring user involvement with the details of interaction among procedures (but [21] reports some second thoughts).

The work of many other groups could be added to this survey. Recent work on REL, building on on [10, 11] is reported in [36, 37]; [24, 25] are relevant collections; and [26] is a survey paper.

4. DEDUCTION

In all of the program systems described thus far, "language understanding" depends on the "deductive capabilities" of the

---

*Some experiments on problem-solving effectiveness of special programming languages in another context appear in [22].

program, that is, its ability to "infer" facts and relationships from given statements. In some cases deduction involves discerning structure in a set of facts and relationships. This section describes how "understanding" programs themselves are structured and how that structure limits their capability for general deduction.

Theorem-proving programs use an inference rule illustrated in [23 p. 61] to deduce new knowledge. A formal succession of logical steps called resolutions leads to the new fact. The example there begins with P1 - P4 given:

P1 if x is part of v, and if v is part of y, then x is part of y;
P2 a finger is part of a hand;
P3 a hand is part of an arm;
P4 an arm is part of a man

A proof that

P9 a finger is part of a man

is derived by steps, such as combining P1 and P2 to get

P6 if a hand is part of y, then a finger is part of y

Unfortunately, it is easy to move outside the domain where the computer can make useful deductions, and the formal resolution process is extremely lengthy and thus prohibitively costly in computer time. In [31, 32] it is shown that some statements ("who did not write ---?") are unanswerable and that there is no algorithm which can detect whether a question stated in a zero-one logical form can be answered. Hence

theorem proving is not e-sential to "deduction" and "understanding" systems, natural or artificial, must rely on other techniques, e.g., outside information such as knowledge about the domain.

In most "understanding" programs, information on a primitive level of processing can be inaccurate; for example, the identification of a sound string "blew" can be inaccurately "blue" Subsequent processing levels combine identified primitives. If parts of speech are concerned, the level is syntactic; if meaning is involved, "semantic"; if domain is involved, the level is that of the "world". Each level can be an aid in a deductive process, leading to "understanding" an input segment of language. Programs NOW EXIST which operationally satisfy most of the following points concerning "understanding" in narrow domains (emphasis has been added)

> Perhaps the most important criterion for understanding a language is the ability TO RELATE THE INFORMATION CONTAINED in a sentence TO KNOWLEDGE PREVIOUSLY ACQUIRED. This IMPLIES HAVING SOME KIND OF MEMORY STRUCTURE IN WHICH THE INTERRELATIONSHIPS OF VARIOUS PIECES OF KNOWLEDGE ARE STORED AND INTO WHICH NEW INFORMATION MAY BE FITTED... The memory structure in these programs may be regarded as semantic, cognitive, or conceptual structures...these programs can make statements or answer questions based not only on the individual statements they were previously told, but also on THOSE INTERRELATIONSHIPS BETWEEN CONCEPTS that were built up from separate sentences as information was incorporated into the structure...

> THE MEANINGS OF THE TERMS STORED IN MEMORY ARE PRECISELY THE TOTALITY OF THE RELATIONSHIPS THEY HAVE WITH OTHER TERMS IN THE MEMORY. [28 pp. 3-4]

This has been accomplished through clever (and lengthy) computer programming, and by taking advantage of structure inherent in special problem domains such as stacking blocks on a table, moving chess pieces, and retrieving facts about a large naval organization.

Program systems for understanding begin with a "front end": a portion designed to transform language input into a computer representation. The representation may be as simple as a character-by-character encoding of alphabetic, space marker, and punctuation elements. However, a complex "front end" could involve word and phrase detection and encoding. The usual computer science term for a computer representation is "data structure" [27] and there are many types. The language processing program DEACON used ring structures [11], a representation frequently used to store queues. In principle a data structure can represent involved associations, but in practice simple order or ancestor relationships predominate Completely different and far more complex types of structure are inherent in natural language. For example, from [28]

> "The professors signed a petition." is not true.

has for valid interpretations:

(a) The professors DIDN'T sign a petition.
(b) THE PROFESSORS didn't sign a petition.
(c) The professors didn't sign a PETITION.
(d) The professors didn't SIGN a petition.

Iterative substitution of alternatives to deduce overall meaning yields cumbersome processing, especially when there are nested uncertainties. The recursive properties associated with the data structure term "list" [27] are not easily adapted to multiple meanings. Hence, representing linguistic data for computation is an open and fundamental research problem. Nevertheless, the programs which deduce facts from language do so without a clear best technique for computer representation. To do this, restrictions on the language implicit in the input domain are used, and repeated processing by level (lexical, syntactic, semantic) is used in the absence of an efficient representation language. Data structures that facilitate following the language structure are needed Existing programs provide special solutions to the problems of deductive processing in narrow language domains While these programs are not a general breakthrough in representing language data for computation, they demonstrate that current programming techniques enable a useful "understanding" capability Furthermore, there is a real potential for use of the "understanding" in an interactive mode to facilitate use of computers by nonspecialists and to tap the more sophisticated human understanding capabilities

5 INTERACTION

Research and computer program development designed to store multitudes of facts so that they can be accessed [29] or combined [30] and "understood (see pp. 3-10 in [30]) in

linguistic form (see pp. 11-17 of [30]) is highly relevant to recent research programs in text and speech understanding. When such a system is used a user might fail to get a fact or relationship because the natural-language subset chosen to represent his question was too righ--i.e., it includes a complex set of logical relationships not in the computer. Thus a block could result in a human-computer dialogue if the program has no logical connection between "garage" and "car" but only between "garage" and "house" (the program replies "OK" or "???" to user input sentences)

```
I LIKE CHEVROLETS.
OK
CHEVROLETS ARE ECONOMICAL.
OK
MY HOUSE HAS A LARGE GARAGE.
OK
I CAN GET TWO IN
???
```

The computer failed to "understand" that there was no change of discourse subject. This is an example of a "semantic" failure which could be overcome by interaction. That is, the human user would need to input one more meaning or association of a valid word so that computer "understanding" may be achieved. Syntactic blocks may also occur. M. Denicoff pointed out that in [7] 172 different syntactic features were used for a situation where there are no statements with psychological content and no use of simile. If the psychological meanings are added as in [38], these features would not be

enough to describe all the possible meanings of a text drawn from a less artificial source. Indeed, a key problem which formal grammars seem ill-suited for is the reality that many contexts may be simultaneously valid: multiple meanings give natural-language communication the richness of overtones and subtleties--poetry carries this to an extreme.

The above dialogue on "Chevrolets" is an example of what Carbonell [39, p. 194] called "mixed-initiative discourse". This important aspect of interaction is considered in the LISP program DWIM ("Do What I Mean"), which is a useful working tool for text-input error correction precisely because it "understands" the user's characteristics. (For example, typical spelling errors.) This is discussed by Teitelman [40, 41, 42]

> A great deal of effort has been put into making DWIM "smart". Experience with perhaps a dozen different users indicates we have been very successful: DWIM seldom fails to correct an error the user feels it should have, and almost never mistakenly corrects an error. [40, p. 11]

Another limited-discourse interactive program [43] facilitates introduction of expert knowledge on chess. The program uses search with a maximum look-ahead depth of 20 and has backtracking capability; both syntactic and semantic knowledge is incorporated. By grouping similar board positions (i.e., all involving a piece on cell 1, all involving a queen move), it imposes semantic organization on the vast files to be searched and improves syntactic processing speed

6. SPEECH

Publication of [44], which coined the term "speech understanding", initiated the natural next step toward use of the computer's "understanding" capability. The goal of easy interaction with the computer becomes more exciting with speech as input medium. Systems to recognize both text and speech have used syntax and context [45, 46], but [47] added a comprehensive approach using multiple processing levels to resolve ambiguities. In the direct successors of this work [8, 49], the same process of partial acceptance of primitive elements (phonemic candidates from digitized acoustic data) followed by lexical, syntactic, and semantic processing to rank alternatives has shown significant success. Reddy (in a Carnegie-Mellon University film on the Hearsay System) states hat on 144 connected utterances, involving 676 words, obtained from 5 speakers, performing 4 tasks (chess, news retrieval, medical diagnosis, and desk calculator use), requiring 28 to 76-word vocabularies, the computer program recognition, in terms of words spotted and identified correctly, was

a. 89% with all sources of knowledge
b. 67% without use of semantic knowledge
c. 44% without use of syntactic or semantic knowledge

These results were obtained in October 1973, and have been improved since [50]. However, a key limitation of this form of computer speech "understanding" is response rate. Reddy

estimated that the third word-accuracy figure (without use of syntactic and semantic knowledge) would have to be in excess of 90% to allow the program to achieve a near-human response speed.

The nature of computer "understanding" programs leads to problems of combinatoric explosion in number of alternatives and this lessens the usefulness of multilevel program organization (acoustic-phonetic, lexical, syntactic, semantic, domain, and user interactions) as much in speech processing as in text processing. Prototype speech "understanding" systems have been build [49, 50] and newer acoustic-phonetic and syntactic techniques have been incorporated into this work [49, 51, 52], yet it seems clear that the development of theory in prosody and grammar cannot provide a breakthrough to escape the combinatoric explosion. The reason is that the search of parse trees and the use of semantics (look up related words) depend on a single context--both take geometrically increasing amounts of computing time as the number of contexts grows. Furthermore, this increase in time is added onto that which occurs when the size of lexicon is expanded. As words are added, the number of trees that can be produced by the grammar's rewriting rules in an attempt to "recognize" a string expands rapidly. Hence in speech as in text processing, "understanding" exists via computer yet it is not likely to lead to machine processing of truly natural language. Indeed the artificiality of speech "understanding" by computer is

even greater than that of text input. The "moon rocks" text system [33, 35] used a vocabulary of 3500 words, while the speech "understanding" version based on it [51] used only 250 words.

The COMMERCIAL AVAILABILITY of systems that recognize isolated words with 98.5% accuracy [53]* and the need for a rapid human-computer input interface [54] promise that the last word has not been spoken on "understanding". Research and development on language handling systems is continuing in the hope of achieving useful "understanding". Indeed, Stanford Research Institute's Artificial Intelligence Center is basing its current work on the just-mentioned isolated-word recognizer. It is likely that useful developments will occur where language, and probably spoken-language, "understanding" will be exhibited. These developments will occur through careful design of tasks and use of advances in computer technology However, the general problem of machine "understanding" of natural language--whether text or speech--is not likely to be aided by these developments.

7 CONCLUSIONS

A large body of research in computer science is devoted to language processing. A survey of the program systems that

---

*Threshold Technology Inc. has sold such a system to several users. Their VIP-100 includes a minicomputer dedicated to the recognition task; there are other isolated-word systems [54]

have been reported shows that two main goals have emerged:

1. To enable "intelligent" processing by the computer ("artificial intelligence")
2. To produce a more useful way to access data and solve problems ("man-machine interaction")

Techniques in artificial intelligence and speech recognition have been developed to the extent that prototype computer program systems which exhibit "understanding" have been developed for highly limited contexts. To extend these programs to larger subsets of natural language poses problems, it is unlikely that any of the research directions currently being explored will of themselves "solve" the "natural language problem". (The techniques include, but are not limited to, further developments in artificial intelligence programming languages [17, 18 20, 21, 55]; refinements in theories of grammar; improved deductive ability, possibly by better theorem-proving techniques; and the introduction of stress-related features in the encoding of speech [52]. A useful collection of language models appears in [56].) Nevertheless, prototype systems for "understanding" both text and speech are useful achievements of engineering, and spoken entry of data by humans to computers is beginning to be established by isolated-word recognizers which use a minicomputer dedicated to the task. A multiplicity of purposes beyond this simple but practical task of data entry are mentioned briefly in the

foregoing discussion of "interaction". Developments along the many diverse paths indicated under that heading are likely to be rapid in the future as practical "understanding" of subsets of language becomes part of computer technology For another view of the evolution of that process, see [57].

REFERENCES

1. Nievergelt, J., and J. C. Farrar, "What Machines Can and Cannot Do," *Computing Surveys, 4*, June 1972, 81-96.

2. Weizenbaum, J., "ELIZA--A Computer Program for the Study of Natural Language Communication Between Man and Machine," *Comm. ACM 9*, January 1966, 36-45.

3. Weizenbaum, J., "Contextual Understanding by Computers," *Comm. ACM 10*, August 1967, 474-480.

4. Bobrow, D. G., "Natural Language Input for a Computer Problem Solving System," in M. Minsky (ed.), *Semantic Information Processing*, MIT Press, Cambridge, Mass., 1968, 135-215

5. Raphael, B., "SIR: Semantic Information Retrieval," in M. Minsky (ed.), *Semantic Information Processing*, MIT, Press, Cambridge, Mass., 1968, 33-134, 256-266.

6. Minsky, M. (ed.), *Semantic Information Processing*, MIT Press, Cambridge, Mass., 1968.

7 Winograd, T., *Understanding Natural Language*, Academic Press, New York, 1972.

8. Plath, W., "Restricted English as a User Language," IBM T. J. Watson Research Center, Yorktown Heights, New York, 1972.

9. Green, P. F., A. K. Wolf, C. Clomsky, and K. Laugherty, "BASEBALL. An Automatic Question-Answer," in E. A. Feigenbaum and J. Feldman (eds.), *Computers and Thought*, McGraw-Hill, New York, 1963.

10. Thompson, F. B., "English for the Computer," *Proc. FJCC*, Spartan, New York, 1968, 349-356.

11. Craig, J. A., S. Berezner, H. Carney, and C. Longyear, "DEACON: Direct English Access and Control," *Proc. FJCC*, Spartan, New York, 1968, 365-380.

12. Kellogg, C., "A Natural Language Compiler for On-Line Data Management," *Proc. FJCC*, Spartan, New York, 1968, 473-492.

13. Travis, L., C. Kellogg, P. Klahr, *Inferential Question-Answering: Extending Converse*, System Development Corporation, SP-3679, January 31, 1973.

14. Kellogg, C. H., J. Burger, T. Diller, and K. Fogt, "The CONVERSE Natural Language Data Management System: Current Status and Plans," in J Minker and S. Rosenfeld (eds.), *Proc. Symp. Information Storage and Retrieval*, University of Maryland, College Park, April 1971, 33-46.

15. Kellogg, C. A., *Question-Answering in the Converse System*, System Development Corporation, TM 5015, October 1971.

16. Nilsson, N. J., *Problem-Solving Methods in Artificial Intelligence* McGraw-Hill, New York, 1971.

17. Rulifson, J. F., R. J. Waldinger, and J. A. Derksen, "A Language for Writing Problem-Solving Programs," *Proc. IFIP Congr. 1971* (presented at Ljubljana, Yugoslavia, August 1971).

18. Rulifson, J. F., *QA4 Programming Concepts*, Stanford Research Institute, Artificial Intelligence Group, Technical Note 60, August 1971.

19. Woods, W. A., "Procedural Semantics for a Question-Answering Machine," *Proc. FJCC*, Spartan, New York, 1968, 457-471.

20. Hewitt, C., "A Language for Theorems in Robots," *Proc. Int. Joint Conf. Artificial Intelligence*, Washington, D.C., 1969, 295-301.

21. Sussman, G. J., and D. V. McDermott, "From PLANNER to CONNIVER--A Genetic Approach" ("Why Conniving is Better Than Planning"), *Proc. 1972 FJCC*, AFIPS, Vol. 41, Part II, 1171-1179.

22. Fikes, R. E. "Monitored Execution of Robot Plans Produced by STRIPS," *Proc. IFIP Congr. 1971* (presented at Ljubljana, Yugoslavia, August 1971). Also see R. E. Fikes and N. J. Nilsson, "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence, 2*, 1971, 189-208.

23. Slagle, J. R., *Artificial Intelligence: The Heuristic Programming Approach*, McGraw-Hill, New York, 1971.

24. Garvin, P. L. (ed.), *Natural Language and the Computer*, McGraw-Hill, New York, 1963.

25. Sass, M. A., and W. D. Wilkinson (eds.), *Computer Augmentation of Human Reasoning*, Spartan, Washington, D.C., 1965.

26. Martins, G. R., "Dimensions of Text Processing," *Proc. 1972 FJCC*, AFIPS, Vol. 41, Part II, 801-810.

27. Knuth, D. *The Art of Computer Programming*: Vol. I *Fundamental Algorithms*, Chap. 2 "Information Structure" Addison-Wesley, Reading, Mass. 1968.

28. Shapiro, S.C., *The MIND System: A Data Structure for Semantic Information Processing*, The Rand Corporation, R-837-PR, August 1971.

29. Levien, R. E., and M. E. Maron, "A Computer System for Inference Execution and Data Retrieval, *Comm. ACM, 10*, 11, November 1967, 715-721.

30. Kochen, M., D. M. MacKay, M. E. Maron, M. Scriven, and L. Uhr, *Computers and Comprehension*, The Rand Corporation, RM-4065-PR, April 1964.

31. Kuhns, J. L., *Answering Questions by Computers: A Logical Study*, The Rand Corporation, RM-5428-PR, December 1967.

32. DiPaola, R., "The Solvability of the Decision Problem for Classes of Proper Formulas and Related Results," *J ACM, 20*, January 1973, 112-126.

33. Woods, W. A., and R. M. Kaplan, *The Lunar Sciences Natural Language Information System*, BBN Report 2265, Cambridge, Mass., September 1971.

34. Woods, W. A., *An Experimental Parsing System for Transition Network Grammars*, BBN Report 2362, Cambridge, Mass., May 1972.

35. Woods, W. A., R. M. Kaplan, and B. Nash-Webber, *The Lunar Sciences Natural Language Information System: Final Report*, BBN Report 2378, Cambridge, Mass., June 1972.

36. Dostert, B. H., and F. B. Thompson, *The System of REL English*, California Institute of Technology, REL Report 1, September 1971.

37. Dostert, B. H. "REL--An Information System for a Dynamic Environment," *REL Report No. 3*, California Institute of Technology, December 1971.

38. Charniak, E., "Jack and Janet in Search of a Theory of Knowledge," *Proc. Int. Joint Conf. Artificial Intelligence*, Stanford, Calif., 1973.

39. Carbonell, J. R., "AT in CAI: An Artificial Intelligence Approach to Computer-Assisted Instruction," *IEEE Trans. Man-Machine Systems* MMS-11, December 1970, 190-202.

40. Teitelman, W., "Do What I Mean: The Programmer's Assistant," *Computers and Automation*, April 1972, 8-11.

41. -----, "Toward a Programming Laboratory," *Proc. Int. Joint Conf. Artificial Intelligence*, Washington, D. C., 1969, 8-11

42. Teitelman, W., D. G. Bobrow, A. K. Hartley, and D. L. Murphy, *BBN-LISP TENEX Reference Manual*, Bolt Beranek and Newman, Cambridge, Mass., 1972.

43. Zobrist, A. L. and F. R. Carlson, Jr., "An Advice-Taking Chess Computer," *Scientific American*, 228, June 1973, 92-105.

44. Newell, A., et. al., *Speech-Understanding Systems: Final Report of a Study Group*, National Technical Information Service, Springfield, Virginia.

45. Duda, R. O., and P. E. Hart, "Experiments in the Recognition of Hand-Printed Text: Part II-Context Analysis," *Proc. FJCC*, Spartan, New York, 1968, 1139-1149.

46. Alter, R., "Utilization of Contextual Constraints in Automatic Speech Recognition," *IEEE Trans. Audio Electroacoustics*, AU-16, March 6-11, 1968.

47. Vicens, P., "Aspects of Speech Recognition by Computer," Ph.D. Dissertation, Stanford University, April 1969. (Also available U. S. Dept. of Commerce Clearinghouse for Federal Scientific and Technical Information, AD687720)

48. D. R. Reddy, L. D. Erman, and R. B. Neely, "A Model and a System for Machine Recognition of Speech," *IEEE Trans. Audio Electroacoustics, Special Issue on 1972 Conference on Speech Communication and Processing, Vol. AU-21*, pp. 229-238, June 1973.

49. V. R. Lesser, R. D. Fennell, L. D. Erman, and D. R. Reddy, "Organization of Hearsay II Speech Understanding System," *IEEE Transactions on Acoustics, Speech, and Signal Processing, (Special Issue on IEEE Symposium on Speech Recognition), Vol. ASSP-23*, pp. 11-24 February, 1975.

50. *IEEE Transactions on Acoustics, Speech, and Signal Processing, (Special Issue on IEEE Symposium on Speech Recognition) Vol. ASSP-23.* February 1975.

51. W. A. Woods, "Motivation and Overview of SPEECHLIS: An Experimental Prototype for Speech Understanding Research," *IEEE Transactions on Acoustics, Speech, and Signal Processing, (Special Issue on IEEE Symposium on Speech Recognition)*, Vol. ASSP-23, pp. 2-10, February, 1975.

52. W. A. Lea, M. F. Medress, and T. E. Skinner, "A Prosodically Guided Speech Understanding Strategy," *IEEE Transactions on Acoustics, Speech and Signal Processing (Special Issue on IEEE Symposium on Speech Recognition)*, Vol. ASSP-23, pp. 30-38, February 1975.

53. T. B. Martin, "Applications of Limited Vocabulary Recognition Systems," *IEEE Transactions on Acoustics, Speech, and Signal Processing, (Special Issue on IEEE Symposium on Speech Recognition)*, Vol. ASSP-23 February 1975.

54. Turn, R. A., S. Hoffman, T. Lippiatt, *Potential Military Applications of Speech Understanding Systems*, The Rand Corporation, R-1434, June 1974.

55. Feldman, J. A., J. R. Low, D. C. Swinehart, and R. H. Taylor, "Recent Developments in SAIL--An Algol-Based Language for Artificial Intelligence," *Proc. 1972 FJCC*, AFIPS, Vol. 41, Part II, 1193-1202.

56. Schank, R. C. and K. M. Colby, eds., *Computer Models of Thought and Language*, W. H. Freeman and Company, San Francisco, 1973.

57. Wilks, Y., "Do Machines Understand More Than They Did?", *Nature*, Vol. 252, 22 November, 1974, pp. 275-278.

CURRENT BIBLIOGRAPHY

The selection of material through the current second year of AJCL's existence remains tentative. A survey of subscriber-members will be included in the last packet mailed during 1975 to establish patterns of coverage for future years.

Categorization of entries deepens as the field defines itself and the collection of literature against which new items can be matched increases. The advice of members is welcome.

Many summaries are authors' abstracts, sometimes edited for clarity, brevity, or completeness. Where possible, an informative summary is provided.

The Linguistic Documentation Centre of the University of Ottawa provides many entries; by editorial accident, some of the entries recently received from that source remain to be included in the next issue. AJCL gratefully acknowledges the assistance of Brian Harris and his colleagues.

Some entries are reprinted with permission from Computer Abstracts.

See the following frames for a list of subject headings and items with extended presentation or review.

SUBJECT HEADINGS

THEORETICAL ISSUES IN NATURAL LANGUAGE PROCESSING

AN INTERDISCIPLINARY WORKSHOP IN

COMPUTATIONAL LINGUISTICS
PSYCHOLOGY
LINGUISTICS
ARTIFICIAL INTELLIGENCE

Cambridge, Massachusetts
June 10-13, 1975

EDITORS:

Professor R. Schank
Department of Computer Science
Yale University
10 Hillhouse Avenue
New Haven, Connecticut 06520

and

B. L. Nash-Webber
Bolt Beranek and Newman Inc
50 Moulton Street
Cambridge, Massachusetts
02138

AVAILABLE FROM: Center for Applied Linguistics
1611 North Kent Street
Arlington, Virginia 22209 *PRICE:* $7.50

ABSTRACTS FOUND ELSEWHERE ON THE MICROFICHE

THE PRAGUE BULLETIN OF MATHEMATICAL LINGUISTICS

22

Universita Karlova

Praha 1974

TABLE OF CONTENTS

COMPUTATIONAL ANALYSES
OF ASIAN & AFRICAN LANGUAGES

A new journal
Mantaro J. Hashimoto, Editor

Project on Computational Analysis
National Inter-University Research Institute
of Asian & African Languages & Cultures

4-51-21 Nishigahara, Kitaku, Tokyo
114 Japan

No. 1 March, 1975

TABLE OF CONTENTS

SYNTAX, SEMANTICS, AND SPEECH

William A. Woods
Bolt Beranek and Newman Inc.
Cambridge, Mass 02138

Report No. BBN 3067 April 1975

Acquaints speech researchers in the state of the art in the conceptual development of, and the new perspectives they place on, parsing, syntax and semantic interpretation. Includes the Chomsky hierarchy of grammar models, non-determinism in parsing and its implementation in either backtracking or multiple independent alternatives, predictive vs. non-predictive parsing, word lattices and chart parsing, Early's algorithm, transition network grammars, transformational grammars and augmented transition networks, procedural semantics, selectional restrictions and semantic association.

IMPROVING METHODOLOGY IN NATURAL LANGUAGE PROCESSING

William C. Mann
USC Information Sciences Institute
Marina Del Rey, California

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 126-129.

Process models are rigorous, process specifications are made very explicit, and complexity is handled by use of computers. A methodology should be reliable, efficient and have integrative power. The distinctive strengths of the current computer oriented methodology are (a) the complexity of data and theory is easy to accommodate, (b) time sequence and dependencies are preserved, and (c) a diversity of hypotheses can be tested. Weaknesses are (a) experiments often take years to perform, (b) the activity is treated as a programming exercise with the status of data and program unclearly defined and (c) in attempting to be general on a particular phenomenon, significant others are missed. As whole systems are produced, they are difficult to disseminate and judge. A system may process its examples, but it is hard to determine if it is ad-hoc and tuned to the examples.

SOME METHODOLOGICAL ISSUES IN NATURAL LANGUAGE UNDERSTANDING RESEARCH

W. A. Woods
Bolt Beranek and Newman, Inc
Cambridge, Mass

In: R. Schank and B.L. Nash-Webber eds. Theoretical Issues in Natural Language Processing, 1975, 134-139.

There are two tasks for which methodologies are used, (a) building intelligent machines, and (b) understanding human language performance. Both depend on the development of a 'device-independent' language understanding theory. For theoretical studies, a methodology should be cognitively efficient and should deal effectively with the problem of scale--having a large number of facts embodied in the theory. Studies should be performed in the context of total language understanding; isolation of components limits scope. Intuition on human language performance is a good guide to computational linguistics.

SPEECH RECOGNITION BY COMPUTER: A BIBLIOGRAPHY WITH ABSTRACTS

D. W. Grooms
National Technical Information Service
5285 Port Royal Rd.
Springfield, Virginia 22161

Report No. Com-74-11435/6, September 1974. Price: $20.00

Contains 142 abstracts covering recognition, synthesis, and the acoustical, phonological and linguistic processes necessary in conversion of various waveforms. Retrieved using the National Technical Information Service on-line search system.

FUZZY LOGIC FOR HANDWRITTEN NUMERICAL CHARACTER RECOGNITION

P. Siy and C. S. Chen
Akron University

IEEE Transactions on Systems, Man and Cybernetics, SMC-4; 570-575, 1974

Considers characters as a directed abstract graph, of which the node set consists of tips, corners, and junctions, and the branch set consists of line segments connecting pairs of adjacent nodes. Classification of branch types produces features which are treated as fuzzy variables. A character is represented by a fuzzy function which relates its fuzzy variables, and by the node pair involved in each fuzzy variable. After producing a representation of an unknown character recognition occurs when a previously learned character's representation is isomorphic to the unknown.

A MEANS OF ACHIEVING A HIGH DEGREE OF COMPACTION ON SCANDIGITIZED PRINTED TEXT

R. N. Ascher and G. Nagy
IBM Corporation

IEEE Transactions on Computers, C-23; 1174-1179, 1974

A 16:1 compaction ratio was achieved by storing only the first instance of each pattern class and thereafter substituting this exemplar for every subsequent occurrence of the symbol. Proposed are refinements to yield a 40:1 ratio.

THE MORPHOLOGY OF CHINESE CHARACTERS
A SURVEY OF MODELS AND APPLICATIONS

William Stallings
Center for Naval Analyses
Arlington, Virginia

Computers and the Humanities 9, 1: 13-24, 1975

Various proposals are discussed, principally (1) Rankin, who has a two-level grammar, the first gives the strokes and rules for combination and the second explicates the order, with a recursive definition of subframes. (2) Fujimara has an inventory of strokes and operators. For each stroke 3 functional points are isolated and operators define the linking by reference to these points. Applications include keyboard input, storage and retrieval or characters, and automatic recognition. There are two different approaches. One seeks a logically efficient system; the other one that seems natural to a user of the language.

CHINESE CHARACTER RECOGNITION BY A STOCHASTIC SECTIONALGRAM METHOD

Y-L. Ma
National Taiwan University

IEEE Transactions on Systems, Man and Cybernetics, SMC-4; 575-584, 1974

An approach to recognition of a block picture by comparing it with stochastic sectionalgrams obtained by grouping many samples. To calculate the risk, the absolute values of the differences between the stroke-occurrence probabilities of corresponding quanta in the two sectionalgrams are summed one of these two sectionalgrams being derived from the input pattern and the other from the prototype pattern. The smaller the sum of these differences is, the more accurate the input pattern recognition.

COMPUTER IDENTIFICATION OF CONSTRAINED HAND PRINTED CHARACTERS WITH A HIGH RECOGNITION RATE

W. C. Lin and T. L. Scully
Case Western Reserve University
Cleveland, Ohio

IEEE Transactions on Systems, Man and Cybernetics, SMC-4, 497-504, 1974

Hand printed on a standardizing grid made of twenty line segments, yielding twenty features, and input using a television camera, 49 character classes were recognized at a greater than 99.4% rate. Feature values calculated utilizing a Gaussian point-to-line distance concept were used in a weighted minimum distance classifier. All character-dependent data are obtained through training techniques Both statistical linear regression and averaging methods are used to obtain the parameters defining each character class in feature space.

THE PHRASAL LEXICON

Joseph D. Becker

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 60-63.

We speak mostly by conjoining remembered phrases. Productive processes have secondary roles of adapting old phrases to new situations and of gap filling.

PROGRAMS FOR LINGUISTIC STATISTICS
PART 1: WORD ROOTS IN SCIENTIFIC AND TECHNICAL RUSSIAN

[Programme zur Sprachstatistik. Teil 1:
Wortstämme in russischen naturwissenschaftlichen und technischen Fachsprachen]

S. Halbauer

Angewandte Informatik, 16: 469-470, 1974

Description of a program, written in machine language, that searches for words containing a fixed stem from Russian mathematical texts.

AUGMENTED PHRASE STRUCTURE GRAMMARS

George E. Heidorn
Computer Sciences Dept.
IBM Watson Research Center
Yorktown Heights, NY

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1-5, 1975.

Augmented phrase structure grammars consist of phrase structure rules with embedded conditions and structure building actions Data structures are records consisting of attribute-value pairs. Records can be actions, words, verb phrases, etc. There are three kinds of attributes: relations, whose value is a pointer to other records; properties, with values either numbers or character strings; and indicators, whose values have a role similar to linguistic feature. Structure building rules have a left part indicating the contiguous segments that must be present for a structure building operation, given in a right part, to apply.

DIAGNOSIS AS A NOTION OF GRAMMAR

Mitchell Marcus
Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 6-10.

The hypothesis is that every language user knows as part of his recognition grammar, a set of highly specific diagnostics that he uses to decide deterministically what structure to build next at each point in the process of parsing a sentence. This theory rejects 'backup as a standard control mechanism for parsing. A grammar is a set of modules. The parser works on two levels, a group level and a clause level. Group level modules work on a word buffer and build group level structures. Modules have a pattern, a pretest procedure and a body to be executed if the pattern matches and the pretest succeeds. If the parser fails, it keeps the structure constructed to date, and makes whatever substructures it can from the remaining part.

Grammar : Parser 39

SOME PROGRAMMING ASPECTS OF COMPUTERS WITH NATURAL LANGUAGE

William White
National Institutes of Health
Division of Computer Research and Technology
Bethesda, Maryland

Journal of Clinical Computing, 3, 100-102, 1973

A morphological analyzer is written in PL/1 using a recursive macro actuated generator. Called with a word as argument it returns a stem, part of speech, possible transformations, and semantic information.

Topic Analysis

Brian Phillips
Department of Information Engineering
University of Illinois at Chicago Circle

Doctoral dissertation, State University of New York, Buffalo, 1975

A theory for the structure of discourse is developed. It is shown that propositions of a coherent discourse must be logically connected and exhibit a hierarchic thematic structure that has a single root. An example of a logical connective is 'Cause'; a theme is a generalized pattern that is associated with a single word, e.g., 'poison' is describable as 'Someone ingests something that causes him to become ill'. A theme applies to a discourse if its definiens matches part of the discourse. The topic of a coherent discourse is its matrix theme; an illformed discourse has no topic.

Not all discourse structure is expressed. If omitted, it must be inferrable. The process of inference requires a store of world knowledge - encyclopedic knowledge. An encyclopedia is described that contains all the devices required by the discourse analysis problem. In fact, the encyclopedia is a general model for human cognition and is applicable to many diverse cognitive tasks. The encyclopedia is a directed graph. Categories of nodes and arcs, and of processes, are presented in detail.

On "FUZZY" ADJECTIVES

Fred J. Damerau
IBM Watson Research Center
Yorktown Heights, N.Y.

Report No. RC 5340 March 27, 1975

Discusses some of the problems that arise when the concept of a linguistic variable is combined with the concept of a fuzzy set: the range of the numerical base variable, in ordering usage, is not fixed for a given linguistic variable. Does not explain the computation of values of compound expressions from the values of their components. Not all adjectives can be related to an underlying numerical base. Other features involved in a complete analysis are: average value, typical value, observed value, standard deviation of values and polarity.

[Distribution limited prior to publication.]

User's Guide to the SOLAR Theoretical Backgrounds File

Timothy Diller and Tom Bye
System Development Corporation
Santa Monica, California 90406

Report No. TM-5292/002/00 April 1975

For each analysis in the semantic analysis file the author's theoretical orientation, his assumptions, and his notational conventions are entered on this file. The data fields are: identifying number, document source, related sources, words analyzed, conventions, theoretical basis including - acknowledgements, assumptions, stated purpose, and limits, a SOLAR critique, and the name of the person responsible for the entry. This file is available via on-line queries or in a listing format. The file can be searched using the identifying number on document source fields. Other fields can be searched using a string-matching facility.

USER S GUIDE TO THE SOLAR SEMANTIC ANALYSIS FILE

Tom Bye, Timothy Diller, and John Olney
System Development Corporation
Santa Monica, California 90406

Report No. TM-5292/001/00 April 1975

This file contains formal descriptions of word meanings, including qualifications, informal explanations, and criticisms of descriptions. The words used are found in the lexicons of the Speech Understanding Research groups being sponsored by ARPA. The semantic analysis produces 23 data fields for each word, of which the following are searchable: word, domain analysis number, source part of speech and components Other fields can be searched using a string matching facility. This file is available via on-line queries or in a listing format.

Semantics - Discourse 42

USER'S GUIDE TO THE SOLAR BIBLIOGRAPHY FILE

Timothy Diller
System Development Corporation
Santa Monica, California 90406

Report No. TM-5292/000/02 December 1974

This file provides the citations to the documents referenced in other SOLAR files. Thirty data fields are used, of which the following are searchable: author, year, index term, document type, subject ID, document number, and Bell ID. Other fields can be searched using a string-matching facility. This file available via on-line queries or in a listing format including an author, keyword and sequence number index.

PRIMITIVES AND WORDS

Yorick Wilks
Istituto per Gli Studi
Semantici e Cognitivi
Castagnola, Switzerland

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 38-41.

If semantic primitives are seen as essentially different from words, this leads to attempts to justify them directly, usually psychologically. Otherwise the justification is merely that they work. Primitives can be taken as a small natural language, with no essential difference between primitives and words. But the set of primitives cannot be extended indefinitely, otherwise the distinction between the representation and the natural language will be lost. If it is not possible to escape from natural language into another realm, one cannot separate semantic representation from reasoning as is attempted. It is probably more sensible to say that natural language understanding depends on reasoning rather than vice-versa.

META-COMPILING TEXT GRAMMARS AS A MODEL FOR HUMAN BEHAVIOR

Sheldon Klein
Computer Sciences Department
University of Wisconsin
Madison

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 84-88.

A key feature of the system is that the semantic deep structure of the non-verbal, behavioral, rules may be represented in the same network as the semantics for natural language grammars, and, as a consequence, provide non-verbal context for linguistic rules. The total system has the power of at least the 2nd order predicate calculus.

THE PRIMITIVE ACTS OF CONCEPTUAL DEPENDENCY

Roger C. Schank
Yale University
New Haven, Connecticut

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 34-37.

Canonical representations of conceptualizations are composed of an ACTOR, an ACTION and a set of ACTION dependent cases. The 12 primitive actions are ATRANS, transfer of possession; PTRANS, transfer of physical location; MTRANS, transfer of information; PROPEL, application of physical force; MBUILD construction of new conceptual information; INGEST, taking in of an object by an animal; GRASP, to grasp; ATTEND, to focus sense organ on an object; SPEAK, to make a noise; MOVE, to move a body part; EXPEL, to push something out of the body; and PLAN, which characterizes the ability to form a course of action that leads to a goal.

Semantics - Discourse 44

COMMENTS ON LEXICAL ANALYSIS

George A. Miller

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 30-33.

An analysis of the verb 'hand' is paraphrased as: 'S had Y prior to some time t at which X used his hand to do something that caused Y to travel to Z, after which Z had Y' The analysis includes a discussion of the subsumed concepts HAPPEN, USE, ACT, CAUSE, ALLOW, BEFORE, TRAVEL, and AT.

A SYSTEM OF SEMANTIC PRIMITIVES

Ray Jackendoff
Department of English
Brandeis University

In R. Schank and B.L. Nash-Webber, eds. Theoretical Issues in Natural Language Processing, 1975, 24-29.

Primitive functions GO, BE and STAY can be extended from a positional interpretation to possessional and identificational interpretations. Two kinds of cause are distinguished, CAUSATIVE and PERMISSIVE. Inference rules based on the form of semantic representations derive logical entailments. e.g. CAUSE (X,E)-- E.

COMPUTATIONAL UNDERSTANDING

Christopher K. Riesbeck

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 11-16.

Comprehension is a memory process; breaking computational understanding into subproblems of parsing and semantic interpretation has hindered progress with much effort wasted on the construction of parsers. A system is described in which a monitor takes words from a sentence one at a time, from left to right. From a lexicon expectations of the word (or its root) are added to a master list of expectations. If an element of the master list evaluates to 'true', programs associated with the element are executed. The final structure built by the triggered expectations is the meaning of the sentence.

DOES A STORY UNDERSTANDER NEED A POINT OF VIEW?

Robert P. Abelson
Yale University
New Haven Connecticut

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 140-143.

Reasoning may be propositional or by mental simulation using visual imagery. In the latter situation, do people include acts and objects not present in a given story, but necessary to carry out the simulation. This has not yet been experimentally tested. Experiments have shown that a listener may simulate a story from the point of view of an observer or of a participant in the story. One problem that this raises for AI, if a program can construct an interconnected structure from the text, is the non-uniqueness of this meaning representation. Another problem is that programs should not be designed to preserve all details, but then, what should be forgotten; point of view may be useful here

BRIDGING

Herbert H. Clark
Stanford University
Stanford, California

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 169-174.

Listeners draw inferences from what they hear, but different listeners can make different inferences. One kind of inference in comprehension is in the context of given-new information: the speaker tries to construct the given and new information of each utterance, so that the listener is able to compute unique antecedents for the given information, and so that he will not already have the new information attached to the antecedent. Inference mechanisms include direct reference, identity, pronominalization, epithets, set membership, indirect reference by association, indirect reference by characterization, reasons, causes, consequences, and concurrences. Bridging inferences need not be determinate, but in discourse they seemingly are, and further, are the inferences with fewest assumptions. Both backward and forward inferences are possible, but only the former are determinate.

COMPUTERS AND NATURAL LANGUAGE

A. W. Pratt, M. G. Pacak, M. Epstein and G. Dunham
National Institutes of Health
Division of Computer Research and Technology
Bethesda, Maryland

Journal of Clinical Computing, 3, 85-99, 1973

The Systematized Nomenclature of Pathology (SNOP), in use at NIH, consists of about 15,000 entries in four lists: topography, morphology, etiology, and function. Only a few binary relations on terms are needed; e.g., location of morphology, (lesion) at topography (body site). Numerous relations on the primary relational triples evidently have to be defined.

GENERATION AS A SOCIAL ACTION

Bertram C. Bruce
Bolt Beranek & Newman
Cambridge, Mass 02138

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 64-67.

Generation is a two stage process. The first formulates a plan and the second expresses these intentions; there is feedback between the stages. Intentions can be encoded by (i) establishing presuppositions, (ii) by linguistic conventions, and (iii) by discourse structure. A Social Action Paradigm is a model of the flow of social actions.

THE BOUNDARIES OF LANGUAGE GENERATION

Neil M. Goldman
Information Sciences Institute
University of Southern California

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 74-78.

In generating natural language from a conceptual structure words and syntactic structure must be deduced from the information content of the message. Words are accounted for by a pattern matching mechanism, a discrimination net. The case framework of verbs is one source of knowledge for choice of syntactic structure.

Semantics - Discourse : Expression 48

SPEAKING WITH MANY TONGUES: SOME PROBLEMS IN MODELING SPEAKERS OF ACTUAL DISCOURSE

John H. Clippinger, Jr.
Teleos
Cambridge, Mass 02138

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 68-73

In therapeutic discourse the subject is not so much generating discourse as regulating it. Statements are made, retracted, qualified and restated. The ERMA model simulates this. It has five stages, represented as CONNIVER contexts. The discourse stream has its source in a special program and then flows back and forth between the contexts before achieving its final expression. Each context determines suitability for expression; whether it should be censored or passed on with suggestions for modification. Concepts are represented by means similar to Minsky's frames.

CONSIDERATIONS FOR COMPUTATIONAL THEORIES OF SPEAKING: SEVEN THINGS SPEAKERS DO

John H. Clippinger, Jr.
Teleos
Cambridge, Mass 02138

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 122-125.

Technological computational linguistics is primarily concerned with software technology whereby computers can use and process natural language. Descriptive computational linguistics uses the computer as a means of developing an accurate and empirically valid model of linguistic and cognitive behaviors of human speakers. There is no inherent representation of intentions in the former, and experience is that it cannot easily be generalized to the latter. One problem of modeling is that important things are often hidden by their familiarity.

CREATIVITY IN VERBALIZATION AS EVIDENCE FOR ANALOGIC KNOWLEDGE

Wallace L. Chafe
Department of Linguistics
University of California
Berkeley

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 144-145.

Both propositional and non-propositional knowledge must exist. Interpretive processes during perception individuate and categorize objects. If an object cannot be categorized then the object will be stored with analogic information. During verbalization analogic images will be compared with available category prototypes to decide on the best match for use in the utterance.

REPRESENTATION AND UNDERSTANDING

STUDIES IN COGNITIVE SCIENCE

edited by

Daniel G. Bobrow
Xerox Palo Alto Research Center
Palo Alto, California

Allan Collins
Bolt Beranek and Newman
Cambridge, Massachusetts

Academic Press
1975

Dedicated to the memory of

JAIME CARBONELL, 1928-1973

I. THEORY OF REPRESENTATION

II NEW MEMORY MODELS

The preface is reprinted on the following frames by permission.

Preface

Jaime Carbonell was our friend and colleague. For many years he worked with us on problems in Artificial Intelligence, especially on the development of an intelligent instructional system. Jaime directed the Artificial Intelligence group at Bolt, Beranek, and Newman (in Cambridge, Massachusetts) until his death in 1973. Some of us who had worked with Jaime decided to hold a conference in his memory, a conference whose guiding principle would be that Jaime would have enjoyed it. This book is the result of that conference.

Jaime Carbonell's important contribution to cognitive science is best summarized in the title of one of his publications: *AI in CAI*. Jaime wanted to put principles of Aritificial Intelligence into Computer-Assisted Instruction (CAI) systems. He dreamed of a system which had a data base of knowledge about a topic matter and general information about language and the principles of tutorial instruction. The system could then pursue a natural tutorial dialog with a student, sometimes following the student's initiative, sometimes taking its own intiative, but always generating its statements and responses in a natural way from its general knowledge. This system contrasts sharply with existing systems for Computer-Assisted Instruction in which a relatively fixed sequence of questions and possible reponses have to be determined for each topic. Jaime did construct working versions of his dream--in a system which he called SCHOLAR. But he died before SCHOLAR reached the full realization of the dream.

It was a pleasure to work with Jaime. His kindness and his enthusiasm were infectious, and the discussions we had with him over the years were a great stimulus to our own thinking. Both as a friend and a colleague we miss him greatly.

Cognitive Science. This book contains studies in a new field we call *cognitive science.* Cognitive science includes elements of psychology, computer science, linguistics, philosophy, and education, but it is more than the intersection of these disciplines. Their integration has produced a new set of tools for dealing with a broad range

of questions. In recent years, the interactions among the workers in these fields has led to exciting new developments in our understanding of intelligent systems and the development of a science of cognition. The group of workers has pursued problems that did not appear to be solvable from within any single discipline. It is too early to predict the future course of this new interaction, but the work to date has been stimulating and inspiring. It is our hope that this book can serve as an illustration of the type of problems that can be approached through interdisciplinary cooperation. The participants in this book (and at the conference) represent the fields of Artificial Intelligence, Linguistics, and Psychology, all of whom work on similar problems but with different viewpoints. The book focuses on the common problems, hopefully acting as a way of bringing these issues to the attention of all workers in those fields related to cognitive science.

Subject Matter. The book contains four sections. In the first section, **Theory of Representation**, general issues involved in building representations of knowledge are explored. Daniel G. Bobrow proposes that solutions to a set of design issues be used as dimensions for comparing different representations, and he examines different forms such solutions might take. William A. Woods explores problems in representing natural-language statements in semantic networks, illustrating difficult theoretical issues by examples. Joseph D. Becker is concerned with the representation one can infer for behavioral systems whose internal workings can not be observed directly, and he considers the interconnection of useful concepts such as hierarchical organization, system goals, and resource conflicts Robert J. Bobrow and John Seely Brown present a model for an expert understander which can take a collection of data describing some situation, synthesize a *contingent knowledge structure* which places the input data in the context of a larger structural organization, and which answers questions about the situation based only on the contingent knowledge structure.

Section two, **New Memory Models**, discusses the implications of the assumption that input information is always interpreted in terms of large structural units derived

from experience. Daniel G. Bobrow and Donald A. Norman postulate active *schemata* in memory which refer to each other through use of *context-dependent descriptions*, and which respond both to input data and to hypotheses about structure. Benjamin J. Kuipers describes the concept of a *frame* as a structural organizing unit for data elements, and he discusses the use of these units in the context of a recognition system. Terry Winograd explores issues involved in the controversy on representing knowledge in declarative versus procedural form. Winograd uses the concept of a frame as a basis for the synthesis of the declarative and procedural approaches. The frame provides an organizing structure on which to attach both declarative and procedural information.

The third section, **Higher Level Structures**, focuses on the representation of plans, episodes, and stories within memory. David E. Rumelhart proposes a grammar for well-formed stories. His summarization rules for stories based on this grammar seem to provide reasonable predictions of human behavior. Roger C. Schank postulates that in understanding paragraphs, the reader fills in causal connections between propositions, and that such causally linked chains are the basis for most human memory organization. Robert P Abelson defines a notation in which to describe the intended effects of plans, and to express the conditions necessary for achieving desired states.

The fourth section, **Semantic Knowledge in Understander Systems**, describes how knowledge has been used in existing systems. John Seely Brown and Richard R. Burton describe a system which uses multiple representations to achieve expertise in teaching a student about debugging electronic circuits. Bonnie Nash-Webber describes the role played by semantics in the understanding of continuous speech in a limited domain of discourse. Allan Collins, Eleanor H. Warnock, Nelleke Aiello, and Mark L. Miller describe a continuation of work on Jaime Carbonell's SCHOLAR system. They examine how humans use strategies to find reasonable answers to questions for which they do not have the knowledge to answer with certainty, and how people can be taught to reason this way.

Acknowledgments. We are grateful for the help of a large number of people who made the conference and this book possible. The conference participants, not all of whom are represented in this book, created an atmosphere in which interdisciplinary exploration became a joy. The people attending were:

From Bolt Beranek and Newman--Joe Becker, Rusty Bobrow, John Brown, Allan Collins, Bill Merriam, Bonnie Nash-Webber, Eleanor Warnock, and Bill Woods.

From Xerox Palo Alto Research Center--Dan Bobrow, Ron Kaplan, Sharon Kaufman, Julie Lustig, and Terry Winograd (also from Stanford University).

From the University of California, San Diego--Don Norman and Dave Rumelhart. From the University of Texas--Bob Simmons. From Yale University--Bob Abelson. From Uppsala University--Eric Sandewall.

Julie Lustig made all the arrangements for the conference at Pajaro Dunes, and was largely responsible for making it a comfortable atmosphere in which to discuss some very difficult technical issues. Carol Van Jepmond was responsible for typing, editing, and formatting the manuscripts to meet the specifications of the systems used in the production of this book. It is thanks to her skill and effort that the book looks as beautiful as it does. June Stein did the final copy editing, made general corrections, and gave many valuable suggestions on format and layout.

Photo-ready copy was produced with the aid of experimental formatting, illustration, and printing systems built at the Xerox Palo Alto Research Center. We would like to thank Matt Heiler, Ron Kaplan, Ben Kuipers, William Newman, Ron Rider, Bob Sproull, and Larry Tesler for their help in making photo-ready production of this book possible. We are grateful to the Computer Science Laboratory of the Xerox Palo Alto Research Center for making available the experimental facilities and for its continuing support.

Daniel G. Bobrow
Allan M. Collins
March 1975

ORGANIZATION AND INFERENCE IN A FRAME-LIKE SYSTEM OF COMMON SENSE KNOWLEDGE

Eugene Charniak
Institute for Semantic and Cognitive Studies
Castagnola, Switzerland

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 42-51.

Frames are static structures about one stereotyped topic Each frame has many statements about the topic, each expressed in a suitable semantic representation. The primary goal in understanding is to find instances of frame statements in the discourse Questions about a source statement can be answered by reference to the frame of which it is an instance.

Semantics - Discourse : Memory 56

COGNITIVE NETWORKS AND ABSTRACT TERMINOLOGY

David G. Hays
Department of Linguistics
State University of New York at Buffalo

Journal of Clinical Computing, 3, 110-118, 1973

By systematic application of a cognitive network or similar theory of knowledge the internal structure of a (medical) code can be improved and tools developed for different purposes. Hays's theory uses paradigmatic, syntagmatic, discursive, attitudinal, and metalingual (MTL) arcs. The MTL arcs shift level of abstraction; e.g., anemia is neither a fewness nor an erythrocyte but an abstract condition. An abstract definition can include several syntagmatic propositions, linked discursively. A medical term can be linked by MTL to definitions in different languages (clinical, Pathophysiological, etc.)

STRUCTURAL KNOWLEDGE IN A DOCUMENT INFORMATION CONSULTING SYSTEM

Ronald J. Brachman
Center for Research in Computing Technology
Harvard University
Cambridge, Mass 02138

Report No. TR 6-75.

A data structure schema for creating structured concept nodes in a semantic network is presented, with structuring techniques based on a set of primitive link types including: defined as attribute part, modality, role, structural/condition, value/restriction, subconcept and superconcept. This structure will store descriptions of bibliographic references in a way that will facilitate the important processes of inference, paraphrase and analogy.

Semantics - Discourse : Memory 57

THE TROUBLE WITH MEMORY DISTINCTIONS

Allan Collins
Bolt Beranek & Newman
Cambridge, Mass. 02138

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 52-54.

Tulving's episodic memory is seen as a record of experiences and their context. However, both episodic and semantic memories must have similar power of representation, so their structures are not distinguishable. Similarly, a lexical memory must have the power to represent propositional information about words. Thus, the fabric of knowledge is merely cut into different shapes.

A FORMALISM FOR RELATING LEXICAL AND PRAGMATIC INFORMATION: ITS RELEVANCE TO RECOGNITION AND GENERATION

Aravind K. Joshi and Stanley J. Rosenschein
The Moore School of Electrical Engineering
University of Pennsylvania
Philadelphia, 19174

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 79-83.

A uniform formal structure for the interpretation of events, initiation of actions, understanding language, and using language is sought. The components of the system are CONTROL --the procedural component; SCHEMATA --a lattice whose points are lexical decompositions; LEXICON --non-definitional information; BELIEFS -- a closed and consistent set of statements in a predicate calculus; and GOALS.

HOW EPISODIC IS SEMANTIC MEMORY?

Andrew Ortony
University of Illinois at Urbana-Champaign

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 55-60.

The distinction between semantic and episodic memory is not so much one between different kinds of memory, but one between different kinds of knowledge. The distinction has been rejected, because it is said that since we know everything from experience, there is no room for the distinction. The error lies in confusing knowledge from knowledge, and knowledge of knowledge. Semantic knowledge is knowledge that has been reorganized around concepts from knowledge originally encoded around events; it is stripped of personal experience. One question raised by the distinction is how does information get into semantic memory, and how and when does it get lost from episodic memory.

BAD-MOUTHING FRAMES

Jerry Feldman
University of Rochester
New York

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 92-93.

There is evidence that people use three-dimensional models and that they integrate several views into a single model. This is counter to the claim that we symbolically store a large number of separate views. Another problem is with the assumption of default values for slots in frames. In the extreme, this gives visual perception without vision. The evidence is that people can understand totally unexpected images presented for quite short periods. A third point concerns the relatively static nature of frames. A better model is to construct a goal oriented subsystem making use of context specific knowledge.

Semantics - Discourse : Memory 59

SOME THOUGHTS ON SCHEMATA

Wallace L. Chafe
University of California
Berkeley

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 89-91.

Stories are broken down into schemata, e.g., plot plus moral. Questions about schemata are: what are the essential ingredients of a schema; are some more abstract than others; and how are they to be discovered--by imagination and intuition?

STEREOTYPES AS AN ACTOR APPROACH TOWARDS SOLVING THE PROBLEM OF PROCEDURAL ATTACHMENT IN FRAME THEORIES

Carl Hewitt

In: R.Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 94-103.

Stereotypes are actor versions of frames. A stereotype has the following parts: a collection of characteristic objects, characteristic relations for these objects and invocable plans for transforming the objects and relations.

Semantics - Discourse : Memory 60

MINSKY'S FRAME SYSTEM THEORY

Marvin Minsky
Artificial Intelligence Laboratory
M.I.T.
Cambridge, Mass

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 104-116.

Frames are data structures for representing stereotyped situations. Each frame contains information about how to use the frame, what to expect to happen next, and what to do if the expectations are not fulfilled. Lower levels of a frame have terminals that can be filled by specific instances from source statements. Frames are linked together into a frame system and the action to go from one to another indicated. Different frames can share the same terminals. Unfilled slots in instances of frames are filled by default options from the general frame.

USING KNOWLEDGE TO UNDERSTAND

Roger C. Schank
Yale University
New Haven, Connecticut

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 117-121.

A SCRIPT is a structure consisting of slots and requirements on what can fill the slots. It is defined as a predetermined causal chain of conceptualizations that describe the normal sequence of things in a familiar situation. A SCRIPT header defines the circumstances under which a SCRIPT is called into play.

GRAMMATICAL INFERENCE: INTRODUCTION AND SURVEY. PART 1

K. S. Fu and T. L. Booth
Purdue University Connecticut University

IEEE Transactions on Systems, Man and Cybernetics, SMC-5: 95-111, 1975

Potential engineering applications. Inference algorithms for finite-state and context-free grammars. Application of some of the algorithms to the inference of pattern grammars in syntactic pattern recognition illustrated by examples.

AN ANTHROPOLOGICAL LINGUISTIC VIEW OF TECHNICAL TERMINOLOGY

Paul L. Garvin
Department of Linguistics
State University of New York at Buffalo

Journal of Clinical Computing, 3, 103-109, 1973

The health-care community has a functional dialect, with subdialects for physicians, nurses, etc. Anthropological study of the naming behavior of the community is a suitable preliminary step in thesaurus building. It would determine what are terms to be entered, how they are related, and what theoretical differences require alternative definitions of the same term.

SYNTACTIC RECOGNITION OF IMPERFECTLY SPECIFIED PATTERNS

M. G. Thomason and R. C. Gonzalez
Tennessee University

IEEE Transactions on Computers, C-24: 93-95, 1975

Using for illustration a recognition system for chromosome structures, methods are developed which basically consist of applying error transformations to the productions of context-free grammars in order to generate new context-free grammars capable of describing not only the original error free patterns, but also patterns containing specific types of errors such as deleted, added, and interchanged symbols which often arise in the pattern-scanning process.

METHODOLOGY IN AI AND NATURAL LANGUAGE UNDERSTANDING

Yorick Wilks
Istituto per Gli Studi
Semantici e Cognitivi
Castagnola, Switzerland

In: R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 130-133.

Artificial Intelligence has had at least four benefits for the study of natural language: (a) emphasis on complex stored structures, (b) emphasis on the importance of real world knowledge, (c) emphasis on the communicative function of sentences in context, and (d) emphasis on the expression of rules, structure and information within the operational environment. The only test of a natural language system is its success on a task, any demand for more theory must bear this in mind. Neither can recent work in AI be regarded as theoretical; it is the semi-formal expression of intuition. AI is engineering, not a science, and as such there is no boundary to natural language; one counter example does not overthrow a rule system. Further, talk of theory distracts from heuristics.

FORMAL REASONING AND LANGUAGE UNDERSTANDING SYSTEMS

Raymond Reiter
Department of Computer Science
University of British Columbia

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 175-179.

There are two mechanisms for formal reasoning: (a) resolution principle, a competence model, by virtue of its completeness, and (b) natural deductive systems, which are attempts to define a performance model for logical reasoning. A system could be designed that interfaces the two systems, each doing what it does best. Natural deductive systems have not considered fuzzy kinds of reasoning. Future questions concern other quantifiers, contexts for representing wanting, needing, etc., and the balance between computation and deduction.

THE COMMONSENCE ALGORITHM AS A BASIS FOR COMPUTER MODELS OF HUMAN MEMORY, INFERENCE, BELIEF AND CONTEXTUAL LANGUAGE COMPREHENSION

Chuck Rieger
Department of Computer Science
University of Maryland

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 180-195.

Commonsense algorithms are basic structures for modeling human cognition. The structure is defined by specifying a set of links which build up large structures of nodes of five types: Wants, Actions, States, Statechanges and Tendencies. There are 25 primitive links, e.g., one-shot causality, action concurrency, inducement Various applications are active problem solving, basis for conceptual representation of language, basis of self model, etc.

UNDERSTANDING HUMAN ACTION

Charles F. Schmidt
Rutgers University
New Brunswick, New Jersey

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 196-200.

A model of reasoning about human action must include (1) how people arrive at a plan, (2) what can count as a reason for choosing to perform the plan, and (3) discovering plans and motivations from observation or linguistic report of actions. A plan is the internal representation or set of beliefs about how a particular goal may be achieved. The belief by an observer that an actor performed one act to enable a second to be performed can follow neither from deductive nor inductive reasoning. An observer may have other propositions that are reasons for believing or not believing that a plan correctly characterizes the beliefs of the actor. An act name organizes a set of beliefs about how a move of this type might relate to other moves, and the cognitive and motivational states of the actors.

Programming

STRING AND LIST PROCESSING IN SNOBOL4: TECHNIQUES AND APPLICATIONS

Ralph E. Griswold

Prentice-Hall, Inc.
Englewood Cliffs
New Jersey
1975

Reviewed by Norman Badler
Department of Computer and Information Science
The Moore School of Electrical Engineering
University of Pennsylvania

Among popular computer programming languages, SNOBOL4 stands out as the only one offering complex pattern definition and matching capabilities. It also has a flexible function definition facility and programmer-defined data types. While not unique, these two features encourage problem-dependent extensions of the language. . All three aspects of SNOBOL4 form the basic tools in Griswold's new book

Intended as a text for the SNOBOL4 user (it is not an "introductory" text), it presents techniques for the representation and manipulation of data in string, list, or otherwise "structured" form. The text includes many programmed examples, problems with a wide range of difficulty, and answers to many of these problems. The first three chapters develop pattern matching, function definition, and data structures. The last four chapters examine particular application domains: mathematics, cryptography, document preparation, Plus a few more specialized problems. Although this may seem to ignore computational linguistics, the greatest immediate benefit for the programmer lies in the first three chapters anyway.

Within Chapter 1, the section on grammars and patterns can be used for the implementation of simple syntactic analysis. For example, there is a straightforward mapping of a BNF grammar into SNOBOL4 patterns, but there are pitfalls (as well as some more efficient representations in the balance) that the programmer ought to know. These are carefully explained.

A topic that I felt was inadequately covered in Chapter 1 was the definition of the pattern matching mechanism itself. The immediate presentation of examples using pattern matching (page 2) calls for a brief overview of pattern matching syntax and semantics. Surely a programmer would appreciate not having to refer back to his introductory text should some pattern function or construct be hazy in his memory. Even an appendix would be satisfactory. In addition, this would support the section on patterns as procedures by providing the underlying semantics for such "procedures." Further incentive for its inclusion is provided by the excellent review of programmer-defined data types in Chapter 3. Why leave pattern matching to the user's recollection?

The function definition facility discussed in Chapter 2 enables the constructic of generic functions. Since there are no data type declarations for function arguments or parameters, often only one function is required for the execution of related operations on various data types. The proliferation of functions in a complex system might therefore be systematically reduced. The burden falls on the programmer, of course, to sort out the admissible combinations or appropriate actions. An addition function for real and complex numbers is discussed, where the former is a SNOBOL4 primitive and the latter is constructed from programmer-defined data types. Although not in the realm of computational linguistics, it does have a parallel, for example, in a function which inserts data into a semantic network and is expected to handle various chunks of network as well as atomic data. The data type might only be determined during program execution; using a generic function avoids distracting logic within the user's primary function.

The section on functions as generators is a little weak from the point of view of computational linguistic requirements for procedures which generate successive alternatives from a complex structure, for example, sentence parsing or

referent resolution. The use of simple global variables is too limited in these contexts; one often needs to become involved with saving the values of several local variables in special data blocks or stacking the decision points associated with alternatives. The first is a well-known compiler-design technique, while the second involves a backtracking control structure. In fact, an excellent illustration of these ideas would be an implementation of the SNOBOL4 pattern matching system in SNOBOL4.

Chapter 3 is the most useful because it describes how programmer-defined data types can be used to build "structures": stacks, queues, linked lists, binary trees, and trees. The skillful user of such representations will find a reduced role for complicated pattern matching expressions because the implicit structure encoded into a string becomes manifest in the explicit links of the structure. Not only is there often an economic advantage, but the semantics of SNOBOL4 are easier to use than the implicit backtracking semantics of pattern matching. (Griswold himself points this out in the section on patterns as procedures.) The programmer is encouraged to consider economic trade-offs in the implementation of structures. Often overlooked questions are addressed: for example, the relative merits of implementing stacks using strings, arrays, tables, or defined data types. Programs for the use or traversal of structures are also provided.

Although exercise 3.40 requests a representation for directed graphs, neither hint nor answer is provided. The computational linguist having an interest in semantic networks or similar associative structures is thus left to his own expertise. The basic tree representation must be significantly modified to incorporate labelled edges, a means of traversal (search) through the edge set, and, of course, non-tree structures. Griswold apologizes for not covering every application,

but the generality and current popularity of networks for the representation of knowledge calls for expanded treatment of the topic.

Among the applications covered in detail, the ones most relevant to computational linguistics include a random sentence generator (from a grammar), a macro processor, and (perhaps) a context editor. The input and output of textual material is covered in depth under document preparation (Chapter 6). Since the text does not delve into computational linguistics per se, the reader (or instructor) will often be called upon to map techniques described in the text onto his own problem. I think that a good programmer would be able to perform this transformation since solutions are provided for many of the basic problems in handling input text, setting up data structures, and traversing these structures.

Before you begin programming your next computational linguistics project, a glance through this book may save you considerable programming time and reward you with usable and flexible data structures. Even if you do not program in SNOBOL4, the techniques presented here might guide you to more efficient usage of other languages. On the other hand, it might convince you to try SNOBOL4.

FORTRAN TECHNIQUES

WITH SPECIAL REFERENCE TO NON-NUMERICAL APPLICATIONS

A. Colin Day

Cambridge University Press
New York
1972

Reviewed by Richard J. Miller
St. Olaf College

A practical guide for the occasional Fortran IV programmer to the basic "tricks" and vocabulary used by the systems programmers. This book ranges over topics from plotting on a line printer to hashing and basic storage structures (stacks, queues, etc.) using a concise, to-the-point writing style. This style reinforces the stated intention of the book, which is to help a programmer with a problem by providing descriptions of non-mathematical techniques. The style and intention do limit the usefulness of this book, as some of the topics would be well known to advanced programmers and are not covered in sufficient depth for such a person. It is then the area between these two extremes to which this book is aimed, and there it can be of great service.

The only important assumption made of the reader is that he know the variable types of Fortran (integer, real, Hollerith, etc.) and their attendant format specifications. A good knowledge of character formats is especially useful, although the major use for them is in output statements used in the examples given in the book. It is also assumed that the reader knows the basic Fortran statements, but this

is simple matter as opposed to the format and variable type problems which confront a Fortran programmer.

The book also includes several exercises at the end of each chapter (answers not supplied unfortunately) and a short but very complete bibliography which includes several sources for each chapter. The book's primary value is as a source for hints to problems encountered during programming, providing an introduction to the more sophisticated literature which can be found by starting with the bibliography. This book is therefore a starting point for picking up a basic vocabulary, techniques, and references for someone who has just completed a programming course or who needs a quick introduction to some technique which he may want to look at later in more detail.

INFORMATION SYSTEMS

VOLUME 1 NUMBER 2

APRIL 1975

Hans-Jochen Schneider, Editor-in-Chief
Institute für Informatik
Universitdt Stuttgart
Herdweg 51
D-7000 Stuttgart 1/Fedrep. Germany

Pergamon Press
Oxford, England
1975

TABLE OF CONTENTS

INFORMATION ABOUT COMPUTER-ASSISTED INFORMATION SYSTEMS

ON RETRIEVING INFORMATION FROM VISUAL IMAGES

Stephen Michael Kosslyn
The Johns Hopkins University
Baltimore, MD

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 146-150.

A computer graphics metaphor is useful for human visual imagery. Analogous oroperties are found: as objects become smaller their constituent parts become more difficult to discern perceptually; as more parts are added to an image it becomes more degraded due to capacity limitations; images displaying more identifiable details take longer to construct; images cannot be indefinitely expanded before overflowing; and the existence of decay time for an image which affects the time taken to construct a new image.

Computation : Pictorial systems 72

REPRESENTATION OF KNOWLEDGE: NON-LINGUISTIC FORMS DO WE NEED IMAGES AND ANALOGUES?

Zenon W. Pylyshyn
Department of Psychology
University of Western Ontario
London, Canada

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 160-163.

Semantic structure is relative to the process that constructs and uses the representation. By positing analogue representations it is suggested that a process does not need to know the rules of transformation, e.g., rotation, but this is impossible unless the analogical modelling medium intrinsically follows the laws of physics, i.e., ascribing these laws to brain tissue.

THE NATURE OF PERCEPTUAL REPRESENTATION: AN EXAMINATION OF THE ANALOG/PROPOSITIONAL CONTROVERSY

Stephen E. Palmer
Department of Psychology
University of California
Berkeley

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 151-159.

Sensory data is considered as having several levels of interpretation. At the sensory end, the representation is analog, and propositional at the cognitive end. Analog images are incorrectly seen as having all details of the stimulus whereas quasi-linguistic representations are only partial. The important issue is not the partiality but the selection, possibly information that discriminates the object in context. For structural information there needs to be a mechanism for both parts and wholes. Parametric information can be coded componentially and explicitly, but some seems to function integrally. It is claimed that structural perception is qualitative whereas parametric perception is quantitative, but structural elements may have quantitative aspects--its strength of association with different groups. Although both structure and parameters are encoded relative to other information, there is evidence of preferred orientation and perspectives for parameters

Computation : Pictorial systems 73

AFTERTHOUGHTS ON ANALOGICAL REPRESENTATIONS

Aaron Sloman
Cognitive Studies Programme
School of Social Sciences
University of Sussex
Brighton, England

In: R. Schank and B.L. Nash-Webber, Eds., Theoretical Issues in Natural Language Processing, 1975, 164-168.

The distinction between Fregean (symbolic) and analogical representations is that in the latter both representation and thing must be complex and there must be correspondence between the structures, whereas in the former case there is no need for a correspondence. Attempts to subsume either representation under the other have not succeeded. There is a mistaken belief that only proofs in Fregean symbolism are rigorous. Although analogical representations can sometimes be implemented using Fregean ones, this does not imply that they are not used.

MEDICAL VOCABULARY

PROCEEDINGS OF THE FIFTH BUFFALO CONFERENCE ON COMPUTERS IN MEDICINE

October 29-31, 1973

Published as the Journal of Clinical Computing
Volume 3, Number 2, September 1973

Editor-In-Chief:
E. R. Gabrieli

TABLE OF CONTENTS

PROCEEDINGS OF THE FIFTH BUFFALO CONFERENCE ON COMPUTERS IN MEDICINE (CONT'D).

INFORMATION STORAGE AND RETRIEVAL

Gerard Salton, Project Director
Department of Computer Science
Cornell University
Ithaca, New York 14853

Scientific Report No. ISR-22
to
The National Science Foundation

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd.)

A THEORY OF TERM IMPORTANCE IN AUTOMATIC TEXT ANALYSIS

G. Salton, C. S. Yang and C. T. Yu
Department of Computer Science, Cornell University, Ithaca, NY 14853
Dept. of Computer Science, University of Alberta, Edmonton, Ata, Canada

In: Information Storage and Retrieval, Gerard Salton, Editor Report No. ISR-22 November 1974

Discrimination value analysis ranks text words in accordance with how well they are able to discriminate the documents of a collection from each other. The value of a term depends on how much the average separation between individual documents changes when the given term is assigned for content identification. The best words are those which achieve the greatest separation. Effective criteria are given for assigning each term to either single word, phrase or word group categories and for constructing optimal indexing vocabularies. The theory is validated by citing experimental results.

MILITARY APPLICATIONS OF SPEECH UNDERSTANDING SYSTEMS

R. Turn, A. S. Hoffman, T. F. Lippiatt
Rand Corporation
Santa Monica, California

Report No. R-1434-ARPA, June 1974

The general military environment. Possible uses: avionics equipment control, field data entry, tactical command systems, and data base management in tactical and administrative systems. New operational capabilities may arise from spoken language translation, biomedical monitoring, and speech-operated writing machines. Applications areas for further research. Methodology for transferring this technology into operational systems.

AUTOMATED REVIEW OF PROFESSIONAL SERVICES AND THE PROBLEMS OF MEDICAL RECORDS

Paul Y. Ertel
Ohio State University
Columbus

Journal of Clinical Computing, 3, 177-184, 1973

The Medical Advances Institute developed a system to keep records, select cases for review, and information about individuals and categories, for the quality control system now established by law. Over 200 quality criteria packages have been developed. They concern the process and result of medical care. The system screens each case within a day of hospital admission and frequently thereafter. It provides a review of use of facilities and conformity to standards of care.

Documentation : Retrieval 79

CONTENT ANALYSIS AND RELEVANCE FEEDBACK

A. Wong, R. Peck, and A. van der Meulen
Cornell University
Ithaca, New York

In: Information Storage and Retrieval, Gerard Salton, Editor. Report No. ISR-22, November 1974

Experimental results indicate that final retrieval system performance, after user feedback is applied using Rocchio's algorithm, is highly dependent on the system performance of the initial indexing process. Therefore every tool which improves the indexing performance as an outcome of the content analysis of natural language is beneficial because initial differences in a system performance are retained after user feedback is applied.

THE BUREAU OF DRUGS, FOOD AND DRUG ADMINISTRATION SCIENTIFIC INFORMATION SYSTEMS

Alan Gelberg
Bureau of Drugs
Food and Drug Administration
Rockville, Maryland

Journal of Clinical Computing, 3, 155-162, 1973

ASTRO-4 is a file of new drug applications. The Ingredients File lists 36,000 chemicals believed to have biological effects. The National Drug Code is a list of manufacturers and products. A file of Clinical Investigators and a file of Facilities are kept. Also Drug Experience and Adverse Drug Reaction, Poison Control Center file of incidents and a Drug Product Defect file. A dictionary of adverse reaction terms is in progress. Sophisticated hardware, software, and terminological controls are in use or development.

PROPOSED USE IN CANADA OF SNOMED IN A MEDICAL INFORMATION MANAGEMENT SYSTEM

Roger A. Cote
University of Sherbrooke Faculty of Medicine
Department of Pathology
Sherbrooke, Quebec

Journal of Clinical Computing, 3, 142-152, 1973

SNOP, published in 1965, is not rich enough to code problems, signs, symptoms, disease entities, administrative, diagnostic, and therapeutic procedures. SNOMed is to cover the whole. The code is hierarchical: Topography is organized by system or tract, Morphology by such categories as traumatic, neoplasm, etc., Etiology by categories of organisms and chemicals, Normal function by metabolism, enzyme, etc., Abnormal function correspondingly, and Procedure by medical discipline. Qualifiers such as history of, laboratory diagnosis, etc., are included, and terms can be linked.

CODING DIAGNOSES OF MEDICAL RECORDS: A CHALLENGE

J. van Egmond and R. Wieme
Medische Informatica Gent
Gent, Belgium

Journal of Clinical Computing, 3, 130-135, 1973

The authors' codification system splits compound diagnoses into units, interrelated if relevant by the grammatical operator "complication of". The content of a unit is described with three sets of codes: disturbance, localization, and etiology. Representation is mnemonic for the coder, numeric for the processor.

RETRIEVAL-ORIENTED STORAGE OF MEDICAL DATA: OPERATIONAL ASPECTS

Charles W. Conaway and Edward T. O'Neill
School of Information and Library Studies
State University of New York at Buffalo

Journal of Clinical Computing, 3, 136-141, 1973

Records are encoded by a clerk. The system is to give a physician at a terminal the current synopsis of a patient record, the complete record (delay of a few minutes), any facts selected for periodic determination in the pool of Clinical Experience; input is to be interactive, with verification of single statements.

DATA MANAGEMENT SYSTEMS AT THE SOCIAL AND REHABILITATION SERVICES

Webster A. Rogers
Division of Management Systems
Social and Rehabilitation Services
Department of Health, Education and Welfare
Washington, D.C.

Journal of Clinical Computing, 3, 164-171, 1973

To improve the management of Medicaid, which spends (predicted) $9 billion for 27 million persons in 1974, an information system was designed and installed in a pilot state. It maintains data about eligibility of persons, qualification (administrative) of providers, claims, background (e.g. normal prices); it delivers statistical summaries and exception reports for managers in addition to processing claims.

THE CLOWNS MICROWORLD

Robert F. Simmons
Department of Computer Science
University of Texas
Austin

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 17-19.

Sentences describing scenes centred around a clown who can balance and move are analyzed by an ATN parser. The parser produces property list semantic structures which are adequate to transmit data to a package that generates the scene on a display screen.

AN HISTORICAL NOTE ON THE USE OF WORD-FREQUENCY CONTIGUITIES IN CONTENT ANALYSIS

H. P. Iker
Rochester University

Computers and the Humanities, 8: 93-98, 1974

Discusses the development of this form of content analysis in information retrieval, the social sciences, and literary analysis.

Social-Behavioral Science 83

BIBLIOGRAPHY ON SOCIAL SCIENCE COMPUTING

R. E. Anderson
Minnesota University
Minneapolis

Computer Reviews, 15; 247-261, 1974

Contains 591 references in the period 1960 to 1973 covering statistical analysis, simulation, text processing, and laboratory automation.

Literature

ASSOCIATION FOR
LITERARY AND
LINGUISTIC
COMPUTING

BULLETIN

Volume 3 Number 1
Lent Term 1975

Editor
Joan M. Smith
6 Sevenoaks Ave
Heaton Moor Stockport
Cheshire SK4 4AW ENGLAND

CONTENTS

Humanities : Concordance

INDEX THOMISTICUS

SANCTI THOMAE AQUINATIS OPERUM OMNIUM INDICES ET CONCORDANTIAE

Roberto Busa, S.J.

Friedrich Fromann Verlag / Günther Holzboog KG, Stuttgart, 1974 -
DM 370 per volume, half-leather

A review of volumes 1-10 of the 26-volume Concordantia Prima

Ford Lewis Battles
Pittsburgh Theological Seminary

It is appropriate indeed that, on the seven hundredth anniversary of the death of St. Thomas Aquinas, Father Roberto Busa with his co-workers of the Faculty of Philosophy at the Aloisianum, Gallarate, Italy has begun to publish the long awaited massive concordance to his writings and to texts by other authors long associated with his circle. For many of us who have done lesser work in computerized humanistic studies, rumors and reports of Busa's enterprise aroused our curiosity and, in some cases, led us also to grapple with the manifold problems of producing a concordance by computer.

In studying the specifications and sampling the first ten volumes of *Index Thomisticus* (Sectio II, Concordantia Prima (A-Initor)), this reviewer has been reminded of his own struggle to produce a concordance to the *Institutes of John Calvin* (Pittsburgh, 1972). The *I.T.* provides a hierarchically organized concordance to a literary corpus of 10,600,000 words of Latin Texts; by comparison, the Calvin concordance contains 405,338 words of Latin text in a single sequence.

Thus, the vastly greater literary task of Busa called for a series of basic literary and philological and logical decisions not only to make the enormous work of processing possible, but also to produce a final instrument for the use of scholars that would rationally encompass the vast corpus.

At the outset the character of the Latin language and especially its morphological peculiarities had to be translated into computerizable routines, so that something other than a sea of raw alphabetical sorting would result. Lemmatization by hand sorting after the basic concordancing (feasible for a small corpus), preparation of an interlined ("glossed") lemmatized machine readable text (also suitable for smaller texts), even the elaborated encoding of the text developed by De Latte at the Liege Centre - none of these methods was chosen by Busa and his associates. They turned rather to Forcellini's *Lexicon totius Latinitatis* and encarded the 90,000 Forcellini lemmata (in all possible forms) plus additional ones in the Thomistic corpus to a total of 10,000,000 codes, put this on magnetic tape, and worked out procedures to apply this Latin Machine Dictionary (LEL) to the machine-readable text. This instrument is now available for the use of others working on Latin texts. To anyone knowing the homographs of Latin, the limitations of any mechanical routine are apparent: the *T.*, however, handles these problems in a clear and workable manner.

The size of the literary corpus also called for basic decisions by successively sequestering different fractions of the corpus (in a way that would have doubtless intrigued Thomas himself!), the compilers reduced the mass to manageable proportions. Their decisions

may be set down serially.

(1) LITERARY. First, divide the authentic works of Thomas (100 + 18?) from those of other authorship (61): treat each in separate series. Secondly, extract literal quotations, citations in references to other authors, and cross-citations to other Aquinian treatises; treat these separately. This leaves distinct layers of material for concordancing.

(2) PHILOLOGICAL. First, separate out indeclinables like prepositions, conjunctions, adverbs, forms of *esse*, helping verbs, etc., pronouns, numerals, etc., etc ; these will occupy a particular concordance. Second, put the remaining nouns, adjectives and verbs in a primary concordance: nouns and adjectives arranged alphabetically by termination; verbs by a standard order.

(3) LOGICAL. To reduce the bulk of the concordance and to make the vocabulary and context more rationally accessible: First, analyze out frequently used phrases (e.g., liberum arbitrium, acceptio personarum, caelum et terra - there are about 500 of these), concordancing them under one word only; for example, acceptio personarum would be listed after all other instances of acceptio. Secondly, distinguish words which are either proper names or so commonly found that only a brief context (1½ lines) need be quoted; the rest can then be set in a context of 2½ lines - and the whole interfiled in a single series.

These decisions have determined both the character and content of Sectio II, comprising two series (Aquinian and non-Aquinian works) of five concordances each. So much for the Concordance proper. There remains a further instrument for the use of scholars, Sectio I. In this are included indices of distribution, summaries of the lexicon, and indices of frequency. Through these lists linguistic and literary studies of all sorts can be made. By reverse alphabetical ordering of lemmata and forms additional kinds of analysis are facilitated. Since these volumes have not yet appeared, they can only be briefly mentioned here.

A massive concordance of this type must carry a concise yet precise location code for each item. The editors have determined the proper modern edition to be used, have set a precise order of works to be followed in the concordancing of each type under its appropriate lemma, and have summarized this on a separate 4-page insert, to which the user will doubtless make frequent reference as he learns to use this grand instrument of research.

A short review cannot do justice to the immense detail and the intelligence with which this detail has, with human and computer help, been marshalled in the *I.T.* Father Busa and his associates are to be commended not only for their achievement, but also for the example they have set for other laborers. Future Concordances to comparable literary corpora will obviously have their own special features; yet Busa's method of attacking problems - linguistic, philological, logical, quantitative - will suggest analogous modes of approach to others. While an index to St. Thomas by no means exhausts even the whole body of Christian Latinity, it provides a key to the heart of Roman theology during

its period of greatest fruitfulness and to classical and patristic thought that passed through the schoolmen's filter. Space also precludes discussion of the physical aspects of concordancing and printing, for which Father Busa had the assistance of IBM.

In a work of such vast proportions, the care of men cannot obviate error. Some 53 errors have been noted by the compilers in Concordantia Prima. But even the correction of at least one of these contains a minor error: B.012(QDV)23 13.ag8/8 should read B.012(QDV)22.13.ag8/8. See Sectio 2 vol. 1, p. 526, col.2. Also 12.Tabula Syntagmatum, the list of phrases concordanced under only one member of the phrase (pp.xiv-xvi) has at least two errors: bona esteriora should read bona exteriora (p. xiv): drosperitas terrena should read prosperitas terrena (p. xvi) The most useful pentaglot descriptive booklet of 46pp. is somewhat marred in the English version by misprints and verbal infelicities. But these small matters are quite eclipsed by the enormous accomplishment of Father Busa and his co-workers.

POTENTIALITIES OF MACHINE PROCESSING OF LATE MIDDLE HIGH GERMAN TEXTS. REPORT ON A RESEARCH PROJECT

[Möglichkeiten der maschinellen Verarbeitung spatmittelhochdeutscher Texte. Bericht über ein Forschungsunternehmen]

T. Baumgarten
Institute for Communication Research and Phonetics
Bonn University

Computers and the Humanities, 8: 85-91, 1974

Lemmatized and classifying index, verse concordance, rhyming index, reverse morphological index frequency list, computer-readable "MHG Working Dictionary" "Syntactical Rule System" making possible a mechanical text description and expandable to a "descriptive grammar".

ON UNDERSTANDING POETRY

D. L. Waltz
University of Illinois
Urbana

In R. Schank and B.L. Nash-Webber, eds., Theoretical Issues in Natural Language Processing, 1975, 20-23.

The processing of discourse is generally organized around verbs. However the structure at a topical or thematic level may not be so organized, bearing little resemblance to a deep structure. Analogies are used, not only in poetry, to transfer large amounts of information from one domain to another; to enable communications of the otherwise inexplicable; to make distinctions vivid; and to understand new concepts by analogy to old ones.

THOUGHT CLUSTERS IN EARLY GREEK ORAL POETRY

Cora Angier Sowa and John F. Sowa
21 Palmer Avenue
Croton-on-Hudson, New York 10520

Computers and the Humanities, 8, 3: 131-146, 1974

Analyzing associations in a literary text is analogous to the problem of computing term associations in document retrieval. This paper describes how the theory of clumps was used to find clusters of closely associated words in the *Homeric Hymns*. For each cluster, the program printed a mini-concordance to the lines of text containing each word in the cluster. The results showed two types of patterns in the poet's use of words: localized word plays extending over a few lines, and global interactions between a cluster of words and the overall thematic structure of the text.

COMPUTER AIDED INSTRUCTION
A BIBLIOGRAPHY WITH ABSTRACTS

E. J. Lehmann
National Technical Information Service
Springfield, Virginia

Report No. Com-74-11376/2, August 1974. Price $20.00

Contains 252 abstracts dating from 1970 to June 1974 covering use in education, computer system requirements, motivation, technical training, learning factors and human factors engineering. Retrieved using the National Technical Information Service on-line search system.