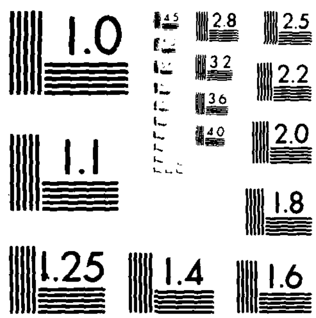
AD-A111 905 YALE UNIV NEW HAVEN CT DEPT OF COMPUTER SCIENCE F/G 5/7
MODELING MEMORY FOR LANGUAGE UNDERSTANDING.(U)
FEB 82 R C SCHANK, M BURSTEIN N00014-75-C-1111
UNCLASSIFIED RR-220 NL

1 of 1
AD
A111905

END
DATE
FILMED
4-82
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

(6)

LUX ET VERITAS

MODELING MEMORY FOR LANGUAGE UNDERSTANDING

February 1982

Research Report #220

Roger C. Schank and Mark Burstein

YALE UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

MODELING MEMORY FOR LANGUAGE UNDERSTANDING

by

Roger Schank and Mark Burstein

Abstract

Research on natural language understanding by computer has shown that the nature and organization of memory plays a central role in the understanding process. This chapter describes a series of computer models displaying increasingly integrated functioning of language and memory processes. The organization of a dynamically changing episodic memory is shown to be important both for its effects on the understanding process, and its ability to make generalizations useful for future understanding.

This work was supported in part by the Advanced Research Projects Agency of the Department of Defense, monitored by the Office of Naval Research under contract N00014-75-C-1111, and in part by the National Science Foundation under contract IST7918463.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER #220	2. GOVT ACCESSION NO. AD-A111905	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Modeling Memory for Language Understanding		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Roger C. Schank and Mark Burstein		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-1111
9. PERFORMING ORGANIZATION NAME AND ADDRESS Yale University Computer Science Department New Haven, Connecticut 06520		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Project Agency 1400 Wilson Boulevard Arlington, VA 22209		12. REPORT DATE February 1982
		13. NUMBER OF PAGES 33
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Office of Naval Research Information Systems Program Arlington, VA 22217		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this report is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Natural Language Analysis
Episodic Memory Organization
Artificial Intelligence
Argumentation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Research on natural language understanding by computer has shown that the nature and organization of memory plays a central role in the understanding process. This chapter describes a series of computer models displaying increasingly integrated functioning of language and memory processes. The organization of a dynamically changing episodic memory is shown to be important both for its effects on the understanding process, and its ability to make generalizations useful for future understanding.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

-- OFFICIAL DISTIRUBTION LIST --

Defense Documentation Center Cameron Station Alexandria, Virginia 22314	12 copies
Office of Naval Research Information Systems Program Code 437 Arlington, Virginia 22217	2 copies
Dr. Judith Daly Advanced Research Projects Agency Cybernetics Technology Office 1400 Wilson Boulevard Arlington, Virginia 22209	3 copies
Office of Naval Research Branch Office - Boston 495 Summer Street Boston, Massachusetts 02210	1 copy
Office of Naval Research Branch Office - Chicago 536 South Clark Street Chicago, Illinois 60615	1 copy
Office of Naval Research Branch Office - Pasadena 1030 East Green Street Pasadena, California 91106	1 copy
Mr. Steven Wong New York Area Office 715 Broadway - 5th Floor New York, New York 10003	1 copy
Naval Research Laboratory Technical Information Division Code 2627 Washington, D.C. 20375	6 copies
Dr. A.L. Slafkosky Commandant of the Marine Corps Code RD-1 Washington, D.C. 20380	1 copy
Office of Naval Research Code 455 Arlington, Virginia 22217	1 copy
Office of Naval Research Code 458 Arlington, Virginia 22217	1 copy

Naval Electronics Laboratory Center — 1 copy
Advanced Software Technology Division
Code 5200
San Diego, California 92152

Mr. E.H. Gleissner — 1 copy
Naval Ship Research and Development
Computation and Mathematics Department
Bethesda, Maryland 20084

Captain Grace M. Hopper, USNR — 1 copy
Naval Data Automation Command, Code OOH
Washington Navy Yard
Washington, D.C. 20374

Dr. Robert Engelmore — 2 copies
Advanced Research Project Agency
Information Processing Techniques
1400 Wilson Boulevard
Arlington, Virginia 22209

Professor Omar Wing — 1 copy
Columbia University in the City of New York
Department of Electrical Engineering and
Computer Science
New York, New York 10027

Office of Naval Research — 1 copy
Assistant Chief for Technology
Code 200
Arlington, Virginia 22217

Captain Richard L. Martin, USN — 1 copy
Commanding Officer
USS Francis Marion (LPA-249)
FPO New York 09501

Major J.P. Pennell — 1 copy
Headquarters, Marine Corp.
(Attn: Code CCA-40)
Washington, D.C. 20380

Computer Systems Management, Inc. — 5 copies
1300 Wilson Boulevard, Suite 102
Arlington, Virginia 22209

Ms. Robin Dillard — 1 copy
Naval Ocean Systems Center
C2 Information Processing Branch (Code 8242)
271 Catalina Boulevard
San Diego, California 92152

Dr. William Woods 1 copy
BBN
50 Moulton Street
Cambridge, MA 02138

Professor Van Dam 1 copy
Dept. of Computer Science
Brown University
Providence, RI 02912

Professor Eugene Charniak 1 copy
Dept. of Computer Science
Brown University
Providence, RI 02912

Professor Robert Wilensky 1 copy
Univ. of California
Elec. Engr. and Computer Science
Berkeley, CA 94707

Professor Allen Newell 1 copy
Dept. of Computer Science
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Professor David Waltz 1 copy
Univ. of Ill at Urbana-Champaign
Coordinated Science Lab
Urbana, IL 61801

Professor Patrick Winston 1 copy
MIT
545 Technology Square
Cambridge, MA 02139

Professor Marvin Minsky 1 copy
MIT
545 Technology Square
Cambridge, MA 02139

Professor Negroponte 1 copy
MIT
545 Technology Square
Cambridge, MA 02139

Professor Jerome Feldman — 1 copy
Univ. of Rochester
Dept. of Computer Science
Rochester, NY 14627

Professor Nils Nilsson — 1 copy
Stanford Research Institute
Menlo Park, CA 94025

Dr. Alan Meyrowitz — 1 copy
Office of Naval Research
Code 437
800 N. Quincy Street
Arlington, VA 22217

Dr. Edward Shortliffe — 1 copy
Stanford University
MYCIN Project TC-117
Stanford Univ. Medical Center
Stanford, CA 94305

Dr. Douglas Lenat — 1 copy
Stanford University
Computer Science Department
Stanford, CA 94305

Dr. M.C. Harrison — 1 copy
Courant Institute Mathematical Science
New York University
New York, NY 10012

Dr. Morgan — 1 copy
University of Pennsylvania
Dept. of Computer Science & Info. Sci.
Philadelphia, PA 19104

Table of Contents

1. Introduction	1
2. Background	3
3. Integrating levels of the understanding process	6
4. In-depth integrated understanding	9
5. Programs that skim text	13
6. Reading with a memory	19
7. Storage and Retrieval from Episodic Memory	22
8. Memory in Arguments	26
9. Conclusion	29

1. Introduction

Consider the following story which appeared recently in a newspaper:

> WASHINGTON -- An Air Florida jetliner taking off from National Airport in a snowstorm crashed into a crowded bridge this afternoon and broke as it plunged into the Potomac River, leaving at least 10 people dead and more than 40 missing, according to unofficial police estimates.

If you saw another story about the same event in a different newspaper, it is quite likely that you would not read this second story as thoroughly as you read the first. Rather, you might skim the story looking only for new information.

A few days after the above story appeared, the following event was reported in the newspapers:

> BOSTON -- A World Airways DC-10 plane carrying 190 passengers skidded as it was landing during a winter storm at Logan International Airport last night, ending up partially in Boston Harbor, authorities said. Passengers were being taken by ambulances to area hospitals. Authorities were uncertain how many were injured.

People who read both stories were likely to have been reminded of the first story while reading the second. Clearly, we must be reminded of the facts we learned from reading other stories about an event when reading new accounts of that event. Otherwise, we will see the new account as describing a completely new situation. But what is the value of being reminded of a story like the first one above when reading about a new situation? We cannot help being reminded in these instances. Our claim is that such reminding is due to the nature of our memory-based understanding mechanism. Further we claim that such reminding is at the root of how we learn.

Issues such as these have played an important part in shaping the direction of our research in natural language processing over the last several years. Prior to that time, we worked primarily on developing representations for aspects of meaning that could be used by a computer in parsing and plausible inference processes designed to build representations of the meaning of texts. The attempt to construct meaning representations for stories, as opposed to single sentences, made it clear that inferring causal connections between events was an extremely important part of the understanding process. Since it was often impossible to determine from just the descriptions of two states or actions what the causal connections between them were, we were led to postulate the existence of a variety of complex memory structures, including scripts, plans, and goals, that could be used to represent the world knowledge needed to infer causal connections in those cases.

What the stories shown above demonstrate is the fact that language processing is dependent not only on general semantic and pragmatic knowledge of the world, but also on the embodiment of that knowledge in a memory system that organizes memories of past experiences. This is crucial if learning and generalization are to take place. It is just as important for the language understanding process.

Language understanding is essentially a memory-based process. We interpret new stories we read not just in terms of our general world knowledge but also in terms of our own related personal experiences. Our interest in a subject, and our knowledge of that subject, are both increased and refined by experience.

This relationship between language and memory is important for verbal discourse as well as for reading. Arguments and everyday conversations appear coherent in part because the memory-based process of interpreting what the other conversant has just said can remind you of something "related", from which a response can be formed. Most importantly, a usefully organized episodic memory can help us to underst[illegible] more fully by filling in details using situations that we have abstracted from our own experience.

2. Background

The first story understanding programs developed at Yale were designed to illustrate some of the different levels of semantic and contextual knowledge that people regularly employ to understand stories, and to show how that knowledge might be represented and used in the understanding process. Schank [Schank 75], described a representational system for simple actions, states, and the causal relations that connect them (conceptual dependency). Schank and Abelson [Schank and Abelson 77] greatly expanded the inferential capability of the system by adding representations for packages of stereotypical action sequences called scripts, various classes of intentional goals, sets of general plans for achieving those goals, and themes giving rise to basic human goals and classes of goals characteristic of different personality types. Computer programs implementing these constructs illustrated the role they play in the process of understanding a text, and at the same time pointed out problems and deficiencies in those representations.

When we read a story, we try to connect together the events described into a coherent sequence. Sometimes this can be done using just our common-sense knowledge of simple causality. Other times, we must resort to generalizations we have made from experience about what "usually happens" in a given situation. The SAM system [Cullingford 78, Cullingford 81] was based on the notion of a script, a knowledge structure representing a stereotypical sequence of events for some often-repeated situation. Scripts were used to represent knowledge about situations like taking a bus or going to a movie theatre. The prototypical example was going to a restaurant for dinner. SAM's representation of the restaurant script ($RESTAURANT) consisted of a set of simple actions described in Conceptual Dependency, together with the causal connections between those actions. The actions in a script were further organized into a sequence of scenes, which in the case of $RESTAURANT included entering the restaurant, ordering food, eating, paying, and leaving. Each script also had a set of roles and props, characterizing the people and objects appearing in that script.

In processing a story about eating in a restaurant, SAM first had to recognize that the restaurant script was the relevant context for interpreting the story. Once the script was chosen, SAM would try to interpret each new sentence it read as part of that script. It did this by matching the conceptual representation of the new sentence against the actions represented in the script. When it found a match, it incorporated the sentence meaning into its representation of the story. It also filled in the script actions preceding the one matched. By this process, SAM inferred actions that were implicit in the story it was reading. Missing agents were inferred in sentences like "The food was served" where the agent, a waiter or waitress, was characterized by his or her role in the script, and the action described was part of that role. Missing actions were automatically inferred from their description in the script. Thus, when SAM read the story:

> John went to the Fisherman's Grotto for dinner.
> He ordered lobster. The bill was outrageous.

it included as part of it's representation of the story that John actually ate his lobster, received a large bill, and paid it. From this representation, SAM could answer a number of questions about the story it had just read, including questions like "What did John eat?", and "Who paid the bill?".

PAM [Wilensky 78, Wilensky 81] dealt with situations which were not at all script-like. The levels of representation it addressed dealt with knowledge of human intentions. That is, PAM was concerned with analyzing stories in terms of the goals of the characters mentioned, and discovering and tracking the plans they used to achieve those goals. Below is a simple story processed by PAM, a summary it produced, and several questions it answered showing how general knowledge of goals and plans helped it make sense of the story.

(UNDERSTAND STORY1)

John needed money. He got a gun and walked into a liquor store.
He told the owner he wanted some money. The owner gave John the
money and John left.

WHO SHOULD TELL THE STORY? John

I needed to get some dough. So I got a gun, and I walked
down to the liquor store. I told the shopkeeper that if he
didn't let me have the money then I would shoot him. So he
handed it over. Then I left.

Q: Why did John get a gun?
JOHN HAD A WEAPON WHICH ENABLED HIM TO ROB THE LIQUOR STORE.

Q: Why did John threaten the shopkeeper?
BECAUSE HE NEEDED TO GET SOME MONEY.

Q: What were the consequences of John threatening the shopkeeper?
THE SHOPKEEPER DIDN'T WANT TO GET HURT, SO HE GAVE JOHN
THE MONEY.

In general, PAM had to do much more work than SAM did to understand a new story, since it dealt with situations for which no script existed, so the process of tying together the events of the story required that many more separate inferences be made. However, by using knowledge of general plans for achieving goals in a wide variety of situations, PAM could construct a more complete explanation of what had occurred in a story, and why.

As it read the above story, PAM inferred that John was going to use a plan to treaten or overpower someone from its knowledge of his goal of getting money and his acquisition of a weapon, the gun. It could then "explain" that the shopkeeper decided to give John the money because his goal of preserving his own life took precedence over keeping the money. Recognizing internal goal conflicts, such as this, and competitions between story characters with similar goals is an extremely important part of the story understanding process. Only by making inferences about the plans and goals driving characters actions, can such conflicts even be noticed.

3. Integrating levels of the understanding process

Both SAM and PAM dealt primarily with levels of the understanding process beyond what is generally considered a linguistic level of analysis (syntax and simple sentential semantics). In fact, both PAM and SAM relied on a conceptual analyzer, ELI [Riesbeck and Schank 76], to produce semantic representations of the actual text, one sentence at a time. On the other end, summaries and answers to questions were produced from the concepts which SAM or PAM incorporated into the final story representation.

While this was a necessary simplification at the time (and one that many language researchers make), it is in some ways unrealistic as part of a model of human language processing. For one thing, words in a text often have special meanings in specific contexts (e.g., compare "John ordered the hamburger." with "The boy ordered the blocks by size."). If a script or some other specific context is known before a sentence is read, then the meaning relevant in that context will be much preferred from the outset. In addition, words can refer directly to complex activities, or to settings for complex events, which must be retrieved from memory before the sentences in which they appear can be correctly interpreted. For example, "John caught the bus," "Fred picked up the phone," and "The plane landed," all describe specific actions which we can understand only if we know the normal uses of the objects mentioned [Lehnert 79, Lehnert and Burstein 79].

Another problem was that SAM and PAM each dealt only with a single level of interpretation. PAM did not deal with script-like situations, and SAM never made inferences about characters' goals and higher level, more abstract plans. Clearly, there are goals associated with scripts. We usually go to restaurants when we are hungry. We go to doctor's offices when we are sick. In fact, knowing someone's goal is an important factor in determining whether a script is active or not. Consider the following stories:

> Freddie the fish merchant only had two more deliveries to make before he was done for the day. When he arrived at Sal's restaurant, he was told the deliveries should be made at the rear entrance.
>
> Freddie the fish merchant hadn't had a good steak in months. When he arrived at Sal's restaurant, he was told the deliveries should be made at the rear entrance.

Reading the second sentence of each story out of context, one would not be sure why the merchant was going to the restaurant. He could have several possible goals, one of which might be to get a good meal. Knowing his goal affects the degree to which we expect to see actions in the restaurant script. Clearly, then, these levels of interpretation are closely intertwined.

Other problems arose from psychological experiments on the nature of scripts in human memory. A number of experiments have confirmed the basic notion that stereotypical sequences were represented in memory [Abelson 80, Graesser, Gordon, Sawyer 79, Abbott and Black 80, Nelson 79]. One study in particular raised an interesting problem. A series of experiments reported by Bower, Black and Turner [Bower, Black and Turner 79] showed that the presence of an underlying script in a story had significant effects on subjects' recall for that story. Specifically, they found that recognition confusions could occur between stories for which the underlying scripts were similar. For example, a story about a visit to a doctor's office was easily confused with a story about a visit to a dentist, particulary in recognizing details of the waiting rooms described in each case.

Intuitively, this result was not surprising, but it posed problems for our conception of scripts. The recognition confusions indicated that some single structure in memory was being used to store information from each story, at least in the scenes where the confusions occurred. Yet, postulating a HEALTH-CARE-VISIT script, or an even more general OFFICE-VISIT script meant that the script was being represented at a level of abstraction far enough removed from its experiential base to rob it of much of its value as an organizer of specific, detailed inferences about a given kind of situation.

What resulted from consideration of these issues was the development of the theory of MOPs (Memory Organization Packets) [Schank 80a, Schank 82]. In general terms, MOPs organize a group of scenes directed towards the achievement of a low level goal. They are usually organized around a single scene that fulfills the purpose of the MOP.

The difference between this definition and the earlier definition of a script is mostly in the reformulation of the notion of a scene. Scenes, under the new definition, function as independent entities in memory. Many MOPs share some of the same generalized scenes, which serve a particular function in the plan implied by each MOP. However, they may also be "colored" by the MOP in which they appear with respect to props and other situation-specific information represented by that MOP. Scenes, then, are the principle organizers of specific memories of episodes. When there are many similar episodes organized around a scene, the result is a script describing the actions in that scene.

Another aspect of the MOP theory was what it implied about the process of understanding a story involving a script. As packagers of sets of scenes, MOPs provide the framework for dynamic creation of extended scripts, possibly involving several purposes. When understanding a scene in a story, several MOPs may be active at one time. If the scene has several purposes for it's characters, then the presence of several active MOP scenes provides several sources of expectations for the events described. Thus, scenes from different MOPs can simultaneously provide explanations for a single scene of a story. For example, a story about a a business meeting in a restaurant may result in simultaneous expectations for the serving of food and the signing of a contract.

4. In-depth integrated understanding

Knowledge source integration was the basis for a program called BORIS [Lehnert, Dyer, et al. 82, Dyer 82]. developed at Yale during the last several years. BORIS was designed as a successor to SAM and PAM. It reads several 300 word narratives in great detail, making inferences at many different levels of representation as it parses the words and sentences of a story. In addition to script and goal inferences, represented in terms of a simplified form of MOPs, BORIS keeps track of story scenes in terms of structures called scenarios. It makes use of a representation of affective states in combination with goal successes and failures to explain the emotional states of characters and to guide its summarization of what it reads. It also makes use of knowledge of interpersonal relationships (e.g., "friendship") to explain things like the granting of favors, the desire for repeated communication between friends, and affective empathy for a friend's successes and failures.

Unlike PAM and SAM, BORIS is largely "bottom-up". That is, its inferential memory is no longer represented as a module or set of active modules looking at the output of a conceptual analyzer (although the first version of BORIS was structured in that way, rather unsuccessfully). Instead, the action of the parser results in almost continuous search of the system's memory for connections between possible meanings being constructed for the sentence it is currently reading, and what it knows about the story so far.

PROCESSING PARAGRAPH 1

> Richard hadn't heard from his college roommate Paul for years. Richard had borrowed money from Paul which was never paid back. but now he had no idea where to find his old friend. When a letter finally arrived from San Francisco, Richard was anxious to find out how Paul was.

Who is Paul?
RICHARD'S FRIEND.

Did Richard want to see Paul?
YES, RICHARD WANTED TO KNOW HOW PAUL WAS.

Had Paul helped Richard?
YES, PAUL LENT MONEY TO RICHARD.

Why didn't Richard pay Paul back?
RICHARD DID NOT KNOW WHERE PAUL WAS.

How did Richard feel when the letter appeared?
RICHARD FELT GLAD BECAUSE PAUL AND HE WERE FRIENDS.

Since the parser controls memory search, BORIS can avoid the problem which ELI faced of generating erroneous intermediate representations for sentences because memory was not available to provide a context-based interpretation. For example, the phrase "pay back", in the context of someone having borrowed money refers specifically to the act of returning the money borrowed. In the context of a vicious prank, or, more generally, causing someone harm, "pay back" refers to the more abstract concept of revenge. By searching episodic memory for a recently mentioned act of borrowing before deciding on an interpretation, BORIS can construct a representation of the phrase which references a part of the active memory structure directly. As with SAM and PAM, this process automatically resolves many pronominal references in the English text of the phrase, as well.

In BORIS, the parser that reads the story is the same as the parser which answers questions about what it has read. While this seems like an obvious thing to have done, in fact, the parser used was originally designed only to answer questions about the story after it had been read by another, more modular program. The question parser was designed to actively search the representation of the story as parsing proceeded, with the result that the event referred to in the question was often found in memory before the parsing of the question was completed, or with very little effort afterward [Dyer and

Lehnert 80]. Once developed, this parser was modified to read the story as well as questions about it, simply by having it build a memory structure when the text referred to one which could not be found by searching the memory structures that had been built previously.

BORIS makes extensive use of the structure of MOPs in it's design, interpreting each scene on many different levels. For example, later in the story shown above, Richard, who turns out to be a lawyer, meets Paul in a restaurant to discuss Paul's impending divorce. This scene is interpreted in terms of a scene in the restaurant script, a lawyer-client meeting, and as the reunion of two old friends. In a variation of this story in which a waitress spills coffee on Paul, it is interpreted both in terms of the action she was supposedly performing of delivering the coffee, and as a violation of the more general DO-SERVICE structure, which explains why Paul refused to leave a tip.

Goal failures and plan violations are extremely important landmarks in a story, both because they represent situations occasioning strong affective responses, and because they mark points of deviation from expectations. In Dynamic Memory [Schank 82], Schank describes a system of TOP's (Thematic Organization Points) for organizing memories of such failures. We claim these structures are needed to explain many kinds of cross-contextual remindings of similar failures and remindings of classic stories and fables depicting similar problem situations. Lehnert has described a system of affect-based plot units also based partially on points of goal failure and achievement, which proved to be a useful predictor of human narrative summarization behavior [Lehnert 82, Lehnert, Black, Reiser 81]. Aspects of both of these systems have been combined by Dyer in his description of TAU's or Thematic Affect Units [Dyer 81], used in the BORIS system to infer characters' affective states, and to generate summaries of the stories it has read. As with all other processing in BORIS, thematic and affective inferences are made during the parsing process. That is, words suggestive of affective states like anxious, relieved, grateful, cheerful, etc. that appear in a story will be related via TAUs to a character's goal failures and achievements.

5. Programs that skim text

The interrelationship between parsing and memory can be observed from many different angles. Another argument against the separation of parsing and memory retrieval is based on the observation that people often don't even read every word of every sentence they see. They can, often without significant loss of meaning, skip words or phrases, fail to finish sentences, or even skip whole paragraphs because of strong contextual predictions which make that extra work seem unnecessary, or because they lack interest in what they expect to see. While this can lead to misinterpretations, the fact that people can read this way runs counter to any strict level-by-level model of language processing.

Skimming is an extreme case of reading from context. Several programs have been written at Yale to simulate forms of this process. The first was called FRUMP (Fast Reading, Understanding and Memory Program) [DeJong 79a, DeJong 79b]. FRUMP skimmed stories taken directly from news wire services. It used a simplified form of script called a "sketchy script," which contained only representations of the concepts which FRUMP needed to extract from a story in order to produce a summary. In a sense, it only read the parts of a story that it was interested in.

FRUMP almost never read an entire sentence of a news story. Instead, it searched the story for words or phrases which could be interpreted as corresponding to an important concept in one of its scripts. Once it was reasonably certain which script best characterized the story, it searched only for phrases which would allow it to fill in the details of the concepts in that script, using syntactic cues only when necessary to help identify an item it was looking for. The remainder of the story was ignored completely. Once it had filled in the required information in the chosen script, it would generate a summary of the story from the concepts in that script. Because the summary was based on FRUMP's internal conceptual representation of the story, FRUMP could produce summaries in many different languages [Carbonell, Cullingford and Gershman 78].

FRUMP had many of the same limitations that SAM had, since it could only "understand" stories that corresponded to one of its scripts. In addition, only about 50% of the stories which come over the news wire are "scripty" in the sense that they contain a fairly standard set of facts to be conveyed. Because it neither had all of the necessary scripts, nor a complete dictionary, FRUMP could read correctly only about 20% of the stories it saw.

The combination of incomplete reading and lack of knowledge created a problem for FRUMP. Since FRUMP did not always have the script which best summarized a story, it would occasionally "guess" the wrong interpretation for a story by choosing another script that matched part of the story.

```
    JERUSALEM (UPI) - A BOMB EXPLOED IN THE WALLED OLD CITY
OF JERUSALEM MONDAY JUST ABOUT THE SAME TIME THE LEADERS OF
EGYPT AND ISRAEL SIGNED A PEACE TREATY IN WASHINGTON.

    A POLICE SPOKESWOMAN SAID THERE WERE "SOME" CASUALTIES BUT
NO FURTHER DETAILS WERE IMMEDIATELY AVAILABLE.

    THE BOMBING CAME DESPITE INCREASED SECURITY PRECAUTIONS BY
ISRAELI ARMY TROOPS AND BORDER GUARDS AGAINST POSSIBLE ATTACKS
BY ARABS OPPOSED TO THE TREATY.

...

Summary without a script for terrorist bombings:
ENGLISH SUMMARY:
        EGYPT AND ISRAEL HAVE AGREED TO A TREATY.

Summary produced by adding that script:
        AN EXPLOSION IN JERUSALEM HAS INJURED SEVERAL PEOPLE.
```

Clearly, in principle FRUMP had the knowledge necessary to see several aspects of this story. However, it could only deal with one script at a time. It couldn't make connections between concepts or scripts if those connections were not already part of one of those scripts. It was also completely incapable of handling the third paragraph, which involved the Israelis' goal of preventing such attacks.

Another "skimming" program, IPP, tried to deal with some of these issues. IPP (Integrated Partial Parser) [Schank 80b, Lebowitz 80]was based on several important observations about the way people read. The first was that in reading with any speed, people don′t have enough time to process every word "completely" and still finish interpreting each sentence within a short time after seeing the last word of that sentence. That is, they can′t consider every possible meaning of every word, or make the inferences necessary to find every possible interpretation of each phrase before going on to the next word or phrase. Therefore, they must limit the amount of time devoted to processing each word and choose to pay more attention to some words than others. The second observation was that what one gets out of a news story is largely governed by what one is interested in [Schank 79]. A reasonable conclusion from these points is that more processing must be devoted to the interesting parts of a story. These hypotheses motivated a kind of "wait and see" strategy for language processing that combines skimming with some careful reading.

The assumption that a relatively small number of words in the text convey most of the content suggests that a good strategy for reading quickly is to do very little processing of any words until a word or phrase referring to some large or interesting concept is seen. After such a concept is found, one can then justify further processing of modifiers of that word, using short-term lexical memory, or even going back in the text if necessary. These concepts also establish a focus of interest for further processing, since they bring with them expectations for details. By continuing to be alert for other independently interesting concepts, in addition to concepts which are interesting because of some relation to an earlier part of the story, one can avoid the problem of being too "top-down." This strategy causes temporary increases in the attention paid to phrases that relay signifigant information about a concept of interest, while allowing other phrases to be ignored altogether. The result is a system which is more flexible and sensitive to the actual text than FRUMP was, and one which suggests an important reason for variability in reading speed.

IPP was an implementation of this basic strategy for news stories in the domain of international terrorism. Below is a story from the New York Times processed by IPP:

> An Arabic-speaking gunman shot his way into the Iraqi Embassy here (Paris) yesterday morning, held hostages through most of the day before surrendering to French policeman and then was shot by Iraqi security officials as he was led away by the French officers.

To give some idea of how IPP works, we will quickly run through its processing of this story. Using a default rule about adjectives, IPP quickly skipped along until it saw the word 'gunman' which it considered *a priori* to be interesting because of its relationship to the concept of death. This led IPP to collect more information about the actor, by looking back at the modifiers it had skipped. Since the action a gunman is most likely to perform is shooting someone, IPP then assumed it might see a verb meaning shot and proceeded to set up expectations for who was shot. When this assumption was quickly confirmed by the verb, it began looking for reasons for that action, by preparing to see indications that the gunman was engaged in a robbery, some terrorist activity, or a kidnapping.

When it saw that the shooting took place in an embassy, a politically significant location, it inferred that terrorism was the reason for the action. Once activated, this intentional level structure replaced the shooting as the principle point of the story, and provided a higher level framework into which pieces of the story fit.

The inference that this story described a terrorist attack also caused IPP to prepare for a number of possible contingencies. It was thus prepared for such scenes as the taking of hostages, making demands, and certain kinds of countermeasures by police. The high interest word ´hostages´ activated the hostage scene and confirmed one of those expectations. The verb preceding it, ´held´ was largely ignored, since it added nothing to the meaning.

The word ´surrendering´ confirmed another expectation. IPP was interested in the surrender scene, so it paid attention to the word ´to´ following ´surrendering´ since that indicated that a description of the captors, in this case some French policemen, would follow.

If the processing of this story had been done in a strictly top-down fashion, the remaining and most interesting part might well have been ignored, since there were no outstanding expectations. However, IPP continued to process the rest of the sentence: "and then was shot by Iraqi security officials." It thus added to its final representation of the story an interesting, though unexpected, subsequent action. What follows is a pictorial version of the final story representation.

```
** MAIN EVENT **

EV1 =
 SCRIPT      I-TERRORISM
 ACTOR       ARAB GUNMAN
 PLACE       IRAQI EMBASSY
 INTEREST    9
 CITY        PARIS
 TIME        MORNING
 SCENES
    EV2 =
     SCENE       $HOSTAGES
     PLACE       IRAQI EMBASSY
     ACTOR       ARAB GUNMAN
     INTEREST    7.
     TIME        DAY
    EV3 =
     SCENE       $CAPTURE
     PLACE       IRAQI EMBASSY
     OBJECT      ARAB GUNMAN
     ACTOR       FRENCH POLICEMEN
     INTEREST    6.
     AFTER       EV2
```

```
** UNEXPECTED EVENTS **

EV4 =
 ACTION      PROPEL
 ACTOR       IRAQI OFFICIALS
 OBJECT      ARAB GUNMAN
 ITEM        *BULLETS*
 DIR-FROM    GUN
 INTEREST    5.
 AFTER       EV3
 RESULT
   EV6 =
     STATE     DEAD
     ACTOR     ARAB GUNMAN
     INTEREST  4.
```

IPP has successfully processed over 300 stories in the domain of international terrorism. Unlike previous programs, however, it did not simply summarize each story in isolation and then "forget" about it. IPP was also a model of how people classify and store such stories in memory, how generalizations are made from similar stories, and how those generalizations can affect the reading of new stories.

6. Reading with a memory

What do we remember of all of the news that we see and hear throughout our lives? We eventually forget the details of most stories, even those that we read carefully at the time. Yet we do get something useful out of what we've read. We develop a characterization of our world from the generalizations we make as we try to understand stories we read.

IPP also forms generalizations when it reads stories which have a lot in common [Lebowitz 81]. Here are two stories that IPP read that led it to make a simple generalization.

New Haven Register, 25 July 79, Israel

Nine people were wounded today by a bomb believed to be set by Palestinian terrorists near a snack bar, police said.

UPI, 10 February 80, Israel

A bomb apparently set by Palestinian Guerrillas exploded in Petah Tikva today, and wounded at least 10 persons, the national radio said.

From these two stories, IPP was able to make the generalization that bombings in Israel are often carried out by Palestinians.

It is a fairly simple process to draw such conclusions when presented with two stories one right after the other. What makes IPP so interesting is the fact that it can make the same generalization no matter how many other stories it read in between its reading of these two. Once IPP has made a generalization, it will use it as a source of inferences when reading new stories. Both of these effects are a result IPP's use of its memory as an integral part of the understanding process. They also depend on the nature of its memory organization system, which models an important aspect of human memory behavior. It gets <u>reminded</u>.

In understanding a story, as in our own experiences, we try to apply the knowledge structures in memory which best match, and hopefully best explain, what we see. These structures are an important part of what we know and will remember about the situation. It is reasonable, then, that we store episodic memories _in terms of those structures_. That is, we remember episodes as variants of previously formed generalizations, distinguishing them from the general structures on the basis of noticed differences from those structures and problems encountered in using them as sources of inferences and explanations for what we see. Thus, the structures used in understanding a new story are the principle organizers of our memories of that story.

What this means for the understanding process is extremely important. In the course of understanding a new story, we will naturally access structures we have used to store similar previous episodes. The very fact we do this can cause us to be reminded of previous episodes which either occurred not long before, shared important features beyond those captured in the general structure, or, most importantly, caused expectation failures similar those that occurred when interpreting the earlier episode [Schank 81].

When we are reminded of an old situation, the two are naturally compared. Storing the new situation "on top of" the old one results in a new generalization in terms of which both experiences are remembered. Stories read after the formation of such a generalization, and which have enough in common with the earlier ones will automatically be interpreted in terms of that generalization, since it provides the most specific plausible inferences that can be made from what is known.

This is precisely why IPP's representation of a third story it read depended on its having seen the two stories above first.

UPI, 2 May 80, Israel

A bomb ripped through a former hospital in the West Bank city of Hebron Friday, killing and injuring several people, the state-run television said.

Without the generalization made in the course of reading the earlier stories, IPP's summary of the third story was "A bombing of a former hospital in Hebron killed and wounded several people." After reading the two earlier stories, however, its summary also included the implicit assumption that the attack was made by Palestinian terrorists. IPP thus represents not only a model of an interested reader, but also in a limited sense a model of a reader whose depth of understanding changes as he reads.

7. Storage and Retrieval from Episodic Memory

Research on episodic memory organization also resulted in the development of the CYRUS system [Schank and Kolodner 79, Kolodner 80, Kolodner 81]. CYRUS was a computer model of intelligent, heuristic-based reconstructive recall from long term memory. The CYRUS program answered questions posed in English about facts which could be derived from events stored in its memory. CYRUS' memory organized information about events in the lives of former Secretaries of State Cyrus Vance and Edmund Muskie. A large percentage of the events it had information about were read directly from a news wire service by the FRUMP program. The rest of the data was added manually, and consisted primarily of background biographical data. Here is a sample dialog with the CYRUS system.

```
ARE YOU INTERESTED IN VANCE OR MUSKIE?  VANCE
BEGINNING DIALOG:

Q1> When was the last time you were in Egypt?
        inferring a diplomatic trip
        answering question using time context
 ON DEC 10, 1978

Q2> Why did you go there?
        answering question using previous context
 TO NEGOTIATE THE CAMP DAVID ACCORDS.

Q3> Who did you talk to there?
        inferring undifferentiated political meeting
 WITH ANWAR SADAT

Q4> Has your wife ever met Mrs. Begin?
        inferring a social political occasion
 YES, MOST RECENTLY AT A STATE DINNER IN ISRAEL IN JAN, 1980.
```

```
Q5> Who have you discussed SALT with?
        inferring a political meeting -- sM-MEETING
     searching memory for question concept
       additional information needed
        meeting could have occurred in USSR or USA
        searching for meeting in USSR
        searching for meeting in USA
     searching memory for episodes meeting could have occurred in
         searching I-NEGOTIATE for meeting
         searching for conferences about SALT
         searching for diplomatic trips to USSR
     searching memory for standard types of meetings
         searching for diplomatic meetings about SALT
         searching for briefings about SALT
         searching for public relations meetings

 CARTER, BREZNEV, GROMYKO, OTHER AMERICAN AND RUSSIAN DIPLOMATS,
 AND MUSTAFA KHALIL.
```

CYRUS indexed events in memory, much as IPP did, in terms of their differences from general event categories called E-MOPs, based on Schank's MOPs. CYRUS dealt with a much wider range of events than IPP, which only read stories about terrorism, and was able to answer questions of varying degrees of specificity.

The model was based on the assumption that episodes are not generally accessible in memory as lists. This assumption is intuitively understandable if you've ever tried to perform any recall task like naming all of the museums you've ever visited. This constraint forced the CYRUS model to cross-index each episode at many levels of description, using structures representing several perspectives to ensure that they would be retrievable later.

As with IPP, storing two similar episodes using some set of indices led to the formation of a new generalization. In addition, if, after a reasonable number of episodes had been stored under an E-MOP, CYRUS noticed that most of them shared additional features, it would try to collapse both generalizations together, in effect reorganizing it's memory to be useful in more situations. When such a merger takes place, fewer features are required to retrieve the generalization and more information is made available when it is accessed later.

When CYRUS was asked a question, it first tried to find those events in memory which contained the necessary information for it to build an appropriate response. Memory search was initiated by building a description of a target event based on the question that was asked. CYRUS then applied a memory search process called traversal to access episodes stored in terms of features found in the target event. In the traversal process, CYRUS used many different characterizations of the people and places mentioned in the question. This effectively varied the search pattern it was using to correspond to a number of the possible generalizations in terms of which episodes matching the target might have been stored originally.

Because CYRUS was designed to answer questions, rather than read stories, it always tried to retrieve specific episodes from its memory. However, the questions it was asked often didn't provide enough information for it to retrieve unique episodes, since it intentionally had no mechanism for linearly searching parts of its memory. If insufficient information was provided to CYRUS for it to access an episode directly by a series of discriminations, then its traversal process failed. Thus while question Q6 provided the traversal process with enough information for it to find an answer, that process could not be used to find answers to questions supplying less information like Q7.

Q6> Have you ever attended a diplomatic meeting about the
     Camp David Accords with Dayan?

Q7> Have you ever attended a meeting in Jerusalem?

The target concept for question Q7 does not describe any situation explicitly enough for CYRUS to retrieve an episode from its memory, even though CYRUS had seen descriptions of many events that had occurred in Jerusalem. What CYRUS did in such cases was apply an inferential elaboration process to facilitate retrieval by generating plausible additional features for its description of the target event.

A number of heuristics were used to generate additional features and alternative contexts for search. Likely participants could often be inferred from the target event's location, or the name of a participating an organization, and vice versa. A topic of discussion at a meeting could provide an important clue as to where the meeting took place or who was involved, as in question Q5 above. Search could also be expanded by inferring larger and related contexts under which an episode might originally have been stored. A diplomatic meeting in Jerusalem, for example, might be found under "meetings in Israel," or as part of a longer sequence of negotiations in many locations on the topic of the Middle East.

The work on CYRUS was an important step in the examination of memory organization and retrieval heuristics for getting information from memory. It provides a framework in terms of which we can examine other, more goal directed memory retrieval problems.

8. Memory in Arguments

Episodic memory can also be viewed as an active part of the language understanding and generation processes we use in conversations and arguments. Forming a response in an argument, in particular, is highly dependent on one's ability to use memory effectively. Responding to a point requires that you have understood your opponent's statement, and followed its implications until you recognized something with which you disagreed. If episodes are stored in memory in terms of the knowledge structures used for understanding and inference, then finding counterexamples should often be a natural consequence of memory-based understanding. This is clearly a useful result, since rebuttal by counterexample is an important form of argument.

Consider the following fragment of a mock argument between an Arab and an Israeli, both sides of which are modeled by a computer program called ABDUL [Birnbaum, Flowers, and McGuire 80, Mcguire, Birnbaum, and Flowers 80].

[1] Arab: Israel is trying to take over the Middle East.

[2] Israeli: If that were our goal, we wouldn't have given back the Sinai to the Egyptians.

[3] Arab: But then why haven't you given back the West Bank to the Palestinians?

In order to understand [1], the Arab's claim must be related to some knowledge structure representing imperialism or military aggression for territorial gain (hereafter called TAKE-OVER), and, if possible, to specific episodes supporting or refuting that claim. Beginning with TAKE-OVER, the program begins to search its memory for evidence related to the Arab's assertion. Episodes involving aspects of the TAKE-OVER structure, including military build-up, attacks on other countries, and military occupation, are examined in this process. This may lead to the discovery of many related facts. Israel was, in fact, trying to increase its military strength, though that was justified in the Israeli's mind by the goal of self-defense. It had occupied territory, including the West Bank and Sinai. It had also agreed to return the Sinai to Egypt.

If [1] had been posed in the form of a question to an impartial observer, or a program like CYRUS, all of these facts might have been used in answering the question. In the context of an argument, however, an appropriate response must satisfy the arguer's goal in the argument. In this case that involves effectively rebutting a claim which is bad for your own position. This conversational goal imposes constraints on the selection of the facts to be used in response. Here, the intention to occupy territory _in general_, represented by the TAKE-OVER structure activated in memory, is contradicted by the action of returning the Sinai, a description of which was found in searching memory for relevant facts about Israel occupying territory. Since it contradicts the Arab's claim, it is a valid rebuttal, and thus the model is led to respond with [2].

This process illustrates the point that just to understand statement [1], most, if not all, of the work required to find an appropriate response in memory may be done. We can be reminded of the facts needed to form an appropriate response as a side-effect of our normal understanding process.

This also explains the Arab's response [3] supporting his claim in [1] that was attacked in [2]. The memory search which the Arab must go through to form that response looks much like the search responsible for [2]. However, since he was responding to an _attack_, his argument goal required that he find positive evidence of OCCUPY-TERRITORY, rather than negative.

Of course, not all rebuttals can be found so easily during the understanding process. Often, the focus of attack must be shifted to different points in the overall argument context. ABDUL represents that context in terms of an _argument graph_, describing evidential relations of support and attack that relate the statements made in the argument. Rules based on noticing patterns in the argument graph can be used to suggest new points to address when a particular point has been exhausted. This process is further described in [Flowers, McGuire, Birnbaum 82].

9. Conclusion

This chapter provides an overview of the ongoing work at Yale on problems in natural language analysis. In describing the progression of this research, we have tried to motivate our growing concern with the nature and organization of human memory. This interest in the functioning of memory is due not only to its effects on the language understanding process, but also to its implications for learning and the representation of meaning.

The programs described here all attest to the difficulty in maintaining modularity in describing of the language analysis process. They also lead to the conclusion that our understanding of language is far from static. Everything we read can affect our understanding because it affects the memory we will use to interpret what we read afterward. This in itself is not a surprising conclusion, but it is certainly an important point to remember as we study the reading process in detail.

The other aim of this chapter was to demonstrate how, through a succession of computer models, we can both discover and address interesting problems in the area of language analysis. The computer provides an excellent testbed for a number of ideas about language and its use, since it forces us to be specific in both our representations of meaning , and in our descriptions of the processes for manipulating and abstracting meaning from text. Examining issues relating to representation of knowledge in memory, the interactions between levels of languange processing and inferential memory processing necessitates the construction of extremely complex models. The computer is an extremely useful tool in this endeavor.

REFERENCES

[Abbott and Black 80]
Abbott, V. and Black, J.B.
The Representation of Scripts in Memory.
Cognitive Science Technical Report 5, Yale University Cognitive Science Program, August, 1980.

[Abelson 80] Abelson, R.P.
The Psychological Status of the Script Concept.
Cognitive Science Technical Report 2, Yale University Cognitive Science Program, March, 1980.

[Birnbaum, Flowers, and McGuire 80]
Birnbaum, L., Flowers, M., and McGuire, R.
Towards an AI Model of Argumentation.
In Proceedings of the First Annual National Conference on Artificial Intelligence, pages 313-315. AAAI, Stanford, CA, August, 1980.

[Bower, Black and Turner 79]
Bower, G. H., Black, J. B., and Turner, T. J.
Scripts in Memory for Text.
Cognitive Psychology 11:177-220, 1979.

[Carbonell, Cullingford and Gershman 78]
Carbonell, J. R., Cullingford, R. E., and Gershman, A.
Knowledge-Based Machine Translation..
Technical Report 146, Yale University. Department of Computer Science, 1978.

[Cullingford 78]
Cullingford, R.
Script Application: Computer Understanding of Newspaper Stories

PhD thesis, Yale University, 1978.

[Cullingford 81]
Cullingford, R. E.
SAM.
In R. C. Schank and C. K. Reisbeck (editors), Inside Computer Understanding: Five Programs Plus Miniatures, . Lawrence Erlbaum Associates, Hillsdale, NJ, 1981.

[DeJong 79a] DeJong, G.
Skimming Stories in Real Time: An Experiment in Integrated Understanding.
PhD thesis, Yale University, May, 1979.

[DeJong 79b] DeJong, G.F.
Prediction and substantiation: a new approach to natural Language processing.
Cognitive Science 3:251-273, 1979.

[Dyer 81] Dyer, Michael G.
The Role of TAUs in Narratives.
Proceedings of the Third Annual Conference of the Cognitive Science Society , August 1981.
Held at Berkeley, CA.

[Dyer 82] Dyer, Michael G.
In-Depth Understanding: A Computer Model of Integrated Processing and Memory for Narrative Comprehension..
PhD thesis, Computer Science Department, Yale University, 1982.

[Dyer and Lehnert 80]
Dyer, Michael G. and Wendy G. Lehnert.
Organization and Search Processes for Narratives.
Technical Report 175, Yale University. Department of Computer Science, 1980.

[Flowers, McGuire, Birnbaum 82]
Flowers, M., McGuire, R., and Birnbaum, L.
Adversary Arguments and the Logic of Personal Attacks.
In W. Lehnert and M. Ringle (editors), Strategies for Natural Language Processing, . Lawrence Erlbaum Associates, Hillsdale NJ, 1982.

[Graesser, Gordon, Sawyer 79]
Graesser, A.C., Gordon, S.E., and Sawyer, J.D.
Memory for typical and atypical actions in scripted activities: Test of a script pointer + tag hypothesis.
Journal of Verbal Learning and Verbal Behavior 18:319-332, 1979.

[Kolodner 80] Kolodner, Janet L.
Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model.
Technical Report 187, Yale University. Department of Computer Science, 1980.
Ph.D. Dissertation.

[Kolodner 81] Kolodner, Janet L.
Organization and Retrieval in a Conceptual Memory for Events or CON54, Where are you?
Proceedings of the 7th International Joint Conference on Artificial Intelligence , 1981.
Held at the University of British Columbia, Vancouver Canada.

[Lebowitz 80] Lebowitz, M.
Generalization and Memory in an Integrated Understanding System

PhD thesis, Yale University, October, 1980.

[Lebowitz 81] Lebowitz, Michael.
The Nature of Generalization in Understanding.
In _Proceedings of the Seventh International Joint Conference on Artificial Intelligence_, pages 58-60. IJCAI, Vancouver, British Columbia, August, 1981.

[Lehnert 79] Lehnert, Wendy G.
Representing Physical Objects in Memory.
In Martin D. Ringle (editor), _Philosophical Perspectives in Artificial Intelligence_, . Humanities Press Inc. Atlantic Highlands, NJ, 1979.

[Lehnert 82] Lehnert, Wendy G.
Plot Units and Narrative Summarization.
Cognitive Science , 1982.
(in press).

[Lehnert and Burstein 79]
Lehnert, Wendy G. and Burstein, Mark H.
The Role of Object Primitives in Natural Language Processing.
Proceedings of the 6th International Joint Conference on Artificial Intelligence , 1979.

[Lehnert, Black, Reiser 81]
Lehnert, W. G., Black, J. B., and B.J. Reiser.
Summarizing Narratives.
Proceedings of the 7th International Joint Conference on Artificial Intelligence , 1981.
Held in Vancouver, B.C.

[Lehnert, Dyer, et al. 82]
Lehnert, W. G., Dyer, M. G., Johnson, P. N., Yang, C. J. and S. Harley.
BORIS -- An In-Depth Understander of Narratives.
Artificial Intelligence , 1982.
(in press).

[Mcguire, Birnbaum, and Flowers 80]
McGuire, R., Birnbaum, L. and Flowers, M.
Opportunistic Processing in Arguments.
In _Proceedings of the Seventh International Joint Conference on Artificial Intelligence_, pages 58-60. IJCAI, Vancouver, British Columbia, August, 1980.

[Nelson 79] Nelson, K.
How Children Represent their world in and out of language.
In Seigler, R.S. (editor), _Children's thinking: What develops?_
Lawrence Erlbaum Associates, Hilsdale, N.J., 1979.

[Riesbeck and Schank 76]
Christopher K. Riesbeck and Roger C. Schank.
Comprehension by Computer: Expectation-Based Analysis of Sentences in Context.
Technical Report 78, Yale University. Department of Computer Science , 1976.

[Schank 75] Schank, R.C.
Fundamental Studies in Computer Science. Volume 3: *Conceptual Information Processing.*
North-Holland, Amsterdam, 1975.

[Schank 79] Schank, R. C.
Interestingness: Controlling Inferences.
Artificial Intelligence 12(3), 1979.

[Schank 80a] Schank, R.C.
Language and Memory.
Cognitive Science 4(3):243-284, 1980.

[Schank 80b] Schank, R.C., Lebowitz, M., and Birnbaum, L.
An integrated understander.
American Journal of Computational Linguistics 6(3):13-30, 1980.

[Schank 81] Schank, Roger C.
Failure-Driven Memory.
Cognition and Brain Theory 1, 1981.

[Schank 82] Schank, Roger C.
Dynamic Memory: A Theory of Reminding and Learning in Computers and People.
Cambridge University Press, Cambridge, 1982.

[Schank and Abelson 77]
Schank, R.C. and Abelson, R.
Scripts, Plans, Goals and Understanding.
Lawrence Erlbaum Associates, Hillsdale New Jersey, 1977.

[Schank and Kolodner 79]
Schank, R. and Kolodner, J.
Retrieving information from and episodic memory, or Why computers memories should be more like people's.
Technical Report 159, Yale University Department of Computer Science, 1979.

[Wilensky 78] Wilensky, R.
Understanding Goal-Based Stories.
PhD thesis, Yale University, 1978.

[Wilensky 81] Wilensky, Robert.
PAM.
In R. C. Schank and C. K. Reisbeck (editors), *Inside Computer Understanding: Five Programs Plus Miniatures*, . Lawrence Erlbaum Associates, Hillsdale, NJ, 1981.

FILMED

4–8