

## READING WITH A PURPOSE

Michael Lebowitz  
Department of Computer Science, Yale University

### 1. INTRODUCTION

A newspaper story about terrorism, war, politics or football is not likely to be read in the same way as a gothic novel, college catalog or physics textbook. Similarly, the process used to understand a casual conversation is unlikely to be the same as the process of understanding a biology lecture or TV situation comedy. One of the primary differences amongst these various types of comprehension is that the reader or listener will have different goals in each case. The reasons a person has for reading, or the goals he has when engaging in conversation will have a strong affect on what he pays attention to, how deeply the input is processed, and what information is incorporated into memory. The computer model of understanding described here addresses the problem of using a reader's purpose to assist in natural language understanding. This program, the Integrated Partial Parser (IPP) is designed to model the way people read newspaper stories in a robust, comprehensive, manner. IPP has a set of interests, much as a human reader does. At the moment it concentrates on stories about international violence and terrorism.

IPP contrasts sharply with many other techniques which have been used in parsing. Most models of language processing have had no purpose in reading. They pursue all inputs with the same diligence and create the same type of representation for all stories. The key difference in IPP is that it maps lexical input into as high a level representation as possible, thereby performing the complete understanding process. Other approaches have invariably first tried to create a preliminary representation, often a strictly syntactic parse tree, in preparation for real understanding. Since high-level, semantic representations are ultimately necessary for understanding, there is no obvious need for creating a preliminary syntactic representation, which can be a very difficult task. The isolation of the lexical level processing from more complete understanding processes makes it very difficult for high level predictions to influence low-level processing, which is crucial in IPP.

One very popular technique for creating a low-level representation of sentences has been the Augmented Transition Network (ATN). Parsers of this sort have been discussed by Woods [11] and Kaplan [3]. An ATN-like parser was developed by Winograd [10]. Most ATN parsers have dealt primarily with syntax, occasionally checking a few simple semantic properties of words. A more recent parser which does an isolated syntactic parse was created by Marcus [4]. The important thing to note about all of these parsers is that they view syntactic parsing as a process to be done prior to real understanding. Even though systems of this sort at times make use of semantic information, they are driven by syntax. Their goal of developing a syntactic parse tree is not an explicit part of the purpose of human understanding.

The type of understanding done by IPP is in some sense a compromise between the very detailed understanding of

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

SAM [1] and PAM [9], both of which operated in conjunction with ELI, Riesbeck's parser [5], and the skimming, highly top-down, style of FRUMP [2]. ELI was a semantically driven parser which maps English language sentences into the Conceptual Dependency [6] representations of their meanings. It made extensive use of the semantic properties of the words being processed, but interacted only slightly with the rest of the understanding processes it was a part of. It would pass off a completed Conceptual Dependency representation of each sentence to SAM or PAM which would try to incorporate it into an overall story representation. Both these programs attempted to understand each sentence fully, SAM in terms of scripts, PAM in terms of plans and goals, before going onto the next sentence. (In [8] Schank and Abelson describe scripts, plans and goals.) SAM and PAM model the way people might read a story if they were expecting a detailed test on it, or the way a textbook might be read. Each program's purpose was to get out of a story every piece of information possible. They treated each piece of every story as being equally important, and requiring total understanding. Both of these programs are relatively fragile, requiring complex dictionary entries for every word they might encounter, as well as extensive knowledge of the appropriate scripts and plans.

FRUMP, in contrast to SAM and PAM, is a robust system which attempts to extract the amount of information from a newspaper story which a person gets when he skims rapidly. It does this by selecting a script to represent the story and then trying to fill in the various slots which are important to understand the story. Its purpose is simply to obtain enough information from a story to produce a meaningful summary. FRUMP is strongly top-down, and worries about incoming information from the story only insofar as it helps fill in the details of the script which it selected. So while FRUMP is robust, simply skipping over words it doesn't know, it does miss interesting sections of stories which are not explained by its initial selection of a script.

IPP attempts to model the way people normally read a newspaper story. Unlike SAM and PAM, it does not care if it gets every last piece of information out of a story. Dull, mundane information is gladly ignored. But, in contrast with FRUMP, it does not want to miss interesting parts of stories simply because they do not mesh with initial expectations. It tries to create a representation which captures the important aspects of each story, but also tries to minimize extensive, unnecessary processing which does not contribute to the understanding of the story.

Thus IPP's purpose is to decide what parts of a story, if any, are interesting (in IPP's case, that means related to terrorism), and incorporate the appropriate information into its memory. The concepts used to determine what is interesting are an extension of ideas presented by Schank [7].

### 2. HOW IPP WORKS

The ultimate purpose of reading a newspaper story is to incorporate new information into memory. In order to do this, a number of different kinds of knowledge are needed. The understander must know the meanings of words, linguistic rules about how words combine into sentences, the conventions used in writing newspaper

stories, and, crucially, have extensive knowledge about the "real world." It is impossible to properly understand a story without applying already existing knowledge about the functioning of the world. This means the use of long-term memory cannot be fruitfully separated from other aspects of the natural understanding problem. The management of all this information by an understander is a critical problem in comprehension, since the application of all potentially relevant knowledge all the time, would seriously degrade the understanding process, possibly to the point of halting it altogether. In our model of understanding, the role played by the interests of the understander is to allow detailed processing to occur only on the parts of the story which are important to overall understanding, thereby conserving processing resources.

Central to any understanding system is the type of knowledge structure used to represent stories. At the present time, IPP represents stories in terms of scripts similar to, although simpler than, those used by SAM and FRUMP. Most of the common events in IPP's area of interest, terrorism, such as hijackings, kidnappings, and ambushes, are reasonably stereotyped, although not necessarily with all the temporal sequencing present in the scripts SAM uses. IPP also represents some events directly in Conceptual Dependency. The representations in IPP consist of two types of structures. There are the event structures themselves, generally scripts such as \$KIDNAP and \$AMBUSH, which form the backbone of the story representations, and tokens which fill the roles in the event structures. These tokens are basically the Picture Producers of [6], and represent the concepts underlying words such as "airliner," "machine-gun" and "kidnapper." The final story representation can also include links between event structures indicating causal, temporal and script-scene relationships.

Due to IPP's limited repertoire of structures with which to represent events, it is currently unable to fully understand some stories which make sense only in terms of goals and plans, or other higher level representations. However, the understanding techniques used in IPP should be applicable to stories which require the use of such knowledge structures. This is a topic of current research.

It is worth noting that the form of a story's representation may depend on the purpose behind its being read. If the reader is only mildly interested in the subject of the story, scriptal representation may well be adequate. On the other hand, for an story of great interest to the reader, additional effort may be expended to allow the goals and plans of the actors in the story to be worked out. This is generally more complex than simply representing a story in terms of stereotypical knowledge, and will only be attempted in cases of great interest.

In order to achieve its purpose, IPP does extensive "top-down" processing. That is, it makes predictions about what it is likely to see. These predictions range from low-level, syntactic predictions ("the next noun phrase will be the person kidnapped," for instance) to quite high-level, global predictions ("expect to see demands made by the terrorist"). Significantly, the program only makes predictions about things it would like to know. It doesn't mind skipping over unimportant parts of the text.

The top-down predictions made by IPP are implemented in terms of requests, similar to those used by Riesbeck [5], which are basically just test-action pairs. While such an implementation in theory allows arbitrary computations to be performed, the actions used in IPP are in fact quite limited. IPP requests can build an event structure, link event structures together, use a token to fill a role in an event structure, activate new

requests or de-activate other active requests.

The tests in IPP requests are also limited in nature. They can look for certain types of events or tokens, check for words with a specified property in their dictionary entry, or even check for specific lexical items. The tests for lexical items are quite important in keeping IPP's processing efficient. One advantage is that very specific top-down predictions will often allow an otherwise very complex word disambiguation process to be bypassed. For example, in a story about a hijacking, IPP expects the word "carrying" to indicate that the passengers of the hijacked vehicle are to follow. So it never has to consider in any detail the meaning of "carrying." Many function words really have no meaning by themselves, and the type of predictive processing used by IPP is crucial in handling them efficiently.

Despite its top-down orientation, IPP does not ignore unexpected input. Rather, if the new information is interesting in itself the program will concentrate on it, making new predictions in addition to, or instead of, the original ones. The proper integration of top-down and bottom-up processing allows the program to be efficient, and yet not miss interesting, unexpected information.

The bottom-up processing of IPP is based around a classification of words that is done strictly on the basis of processing considerations. IPP is interested in the traditional syntactic classifications only when they help determine how words should be processed. IPP's criteria for classification involve the type of data structures words build, and when they should be processed.

Words can build either of the main data structures used in IPP, events and tokens. The words building events are usually verbs, but many syntactic nouns, such as "kidnapping," "riot," and "demonstration" also indicate events, and are handled in just the same way as traditional verbs. Some words, such as most adjectives and adverbs, do not build structures but rather modify structures built by other words. These words are handled according to the type of structure they modify.

The second criteria for classifying words - when they should be processed - is crucial to IPP's operation. In order to model a rapid, normally paced reader, IPP attempts to avoid doing any processing which will not add to its overall understanding of a story. To do this, it classifies words into three groups - words which must be fully processed immediately, words which should be saved in short-term memory, and then processed later, if necessary, and words which should be skipped entirely.

Words which must be processed immediately include interesting words building either event structures or tokens. "Gunmen," "kidnapped" and "exploded" are typical examples. These words give us the overall framework of a story, indicate how much effort should be devoted to further analysis, and, most importantly, generate the predictions which allow later processing to proceed efficiently.

The save and process later words are those which may become significant later, but are not obviously important when they are read. This class is quite substantial, including many dull nouns and nearly all adjectives and adverbs. In a noun phrase such as "numerous Italian gunmen," there is no point in processing to any depth "numerous" or "Italian" until we know the word they modify is important enough to be included in the final representation. In the cases where further processing is necessary, IPP has the proper information to easily incorporate the saved words into the story representation, and in the many cases

where the word is not important, no effort above saving the word is required. The processing strategy for these words is a key to modeling normal reading.

The final class of words are those IPP skips altogether. This class includes very uninteresting words which neither contribute processing clues, nor add to the story representation. Many function words, adjectives and verbs irrelevant to the domain at hand, and most pronouns fall into this category. These words can still be significant in cases where they are predicted, but otherwise they are ignored by IPP and take no processing effort.

In addition to the processing techniques mentioned so far, IPP makes use of several very pragmatic heuristics. These are particularly important in processing noun groups properly. An example of the type of heuristic used is IPP's assumption that the first actor in a story tends to be important, and is worth extra processing effort. Other heuristics can be seen in the example in section 4. IPP's basic strategy is to make reasonable guesses about the appropriate representation as quickly as possible, facilitating later processing and fix things later if its guesses are prove to be wrong.

3. A DETAILED EXAMPLE

In order to illustrate how IPP operates, and how its purpose affects its processing, an annotated run of IPP on a typical story, one taken from the Boston Globe is shown below. The text between the rows of stars has been added to explain the operation of IPP. Items beginning with a dollar sign, such as \$TERRORISM, indicate scripts used by IPP to represent events.

[PHOTO: Initiated Sun 24-Jun-79 3:36PM]

#RUN IPP

\*(PARSE S1)

Input: S1 (3 14 79) IRELAND

(GUNMEN FIRING FROM AMBUSH SERIOUSLY WOUNDED AN 8-YEAR-OLD GIRL AS SHE WAS BEING TAKEN TO SCHOOL YESTERDAY AT STEWARTSTOWN COUNTY (YRONNE))

Processing:

GUNMEN : Interesting token - GUNMEN

Predictions - SHOOTING-WILL-OCCUR ROBBERY-SCRIPT  
TERRORISM-SCRIPT HIJACKING-SCRIPT

\*\*\*\*\*  
GUNMEN is marked in the dictionary as inherently interesting. In humans this presumably occurs after a reader has noted that stories involving gunmen tend to be interesting. Since it is interesting, IPP fully processes GUNMEN, knowing that it is important to its purpose of extracting the significant content of the story. It builds a token to represent the GUNMEN and makes several predictions to facilitate later processing. There is a strong possibility that some verb conceptually equivalent to "shoot" will appear. There are also a set of scripts, including \$ROBBERY, \$TERRORISM and \$HIJACK which are likely to appear, so IPP creates predictions looking for clues indicating that one of these scripts should be activated and used to represent the story.  
\*\*\*\*\*

FIRING : Word satisfies prediction

Prediction confirmed - SHOOTING-WILL-OCCUR

Instantiated \$SHOOT script

Predictions - \$SHOOT-ROLE-FINDER REASON-FOR-SHOOTING  
\$SHOOT-SCENES

\*\*\*\*\*  
FIRING satisfies the prediction for a "shoot" verb. Notice that the prediction immediately disambiguates FIRING. Other senses of the word, such as "terminate employment" are never considered. Once IPP has confirmed an event, it builds a structure to represent it, in this case the \$SHOOT script and the token for GUNMEN is filled in as the actor. Predictions are made trying to find the unknown roles of the script, VICTIM, in particular, the reason for the shooting, and any scenes of \$SHOOT which might be found.  
\*\*\*\*\*

Instantiated \$ATTACK-PERSON script  
Predictions - \$ATTACK-PERSON-ROLE-FINDER  
\$ATTACK-PERSON-SCENES

\*\*\*\*\*  
IPP does not consider the \$SHOOT script to be a total explanation of a snooting event. It requires a representation which indicates the purpose of the various actors. In the absence of any other information, IPP assumes people who shoot are deliberately attacking someone. So the \$ATTACK-PERSON script is inferred, and \$SHOOT attached to it as a scene. The \$ATTACK-PERSON representation allows IPP to make inferences which are relevant to any case of a person being attacked, not just snootings. IPP is still not able to instantiate any of the high level scripts predicted by GUNMEN, since the \$ATTACK-PERSON script is associated with several of them.  
\*\*\*\*\*

FROM : Function word

Predictions - FILL-FROM-SLOT

\*\*\*\*\*  
FROM in a context such as this normally indicates the location from which the attack was made is to follow, so IPP makes a prediction to that effect. However, since a word building a token does not follow, the prediction is deactivated. The fact that AMBUSH is syntactically a noun is not relevant, since IPP's prediction looks for a word which identifies a place.  
\*\*\*\*\*

AMBUSH : Scene word

Predictions - \$AMBUSH-ROLE-FINDER \$AMBUSH-SCENES

Prediction confirmed - TERRORISM-SCRIPT  
Instantiated \$TERRORISM script  
Predictions - TERRORIST-DEMANDS \$TERRORISM-ROLE-FINDER  
\$TERRORISM-SCENES COUNTER-MEASURES

\*\*\*\*\*  
IPP knows the word AMBUSH to indicate an instance of the \$AMBUSH script, and that \$AMBUSH can be a scene of \$TERRORISM (i.e. it is an activity which can be construed as a terrorist act). This causes the prediction made by GUNMEN that \$TERRORISM was a possible script to be triggered. Even if AMBUSH had other meanings, or could be associated with other higher level scripts, the prediction would enable quick, accurate identification and incorporation of the word's meaning into the story representation. IPP's purpose of associating the shooting with a high level knowledge structure which helps to explain it, has been achieved. At this point in the processing an instance of \$TERRORISM is constructed to serve as the top level representation of the story. The \$AMBUSH and \$ATTACK-PERSON scripts are attached as scenes of \$TERRORISM.  
\*\*\*\*\*

SERIOUSLY : Skip and save  
WOUNDED : Word satisfies prediction

Prediction confirmed - \$WOUND-SCENE  
Predictions - \$WOUND-ROLE-FINDER \$WOUND-SCENES

\*\*\*\*\*  
\$WOUND is a known scene of \$ATTACK-PERSON, representing a common outcome of an attack. It is instantiated and attached to \$ATTACK-PERSON. IPP infers that the actor of \$WOUND is probably the same as for \$ATTACK-PERSON, i.e. the GUNMEN.  
\*\*\*\*\*

AN : Skip and save  
8-YEAR-OLD : Skip and save  
GIRL : Normal token - GIRL

Prediction confirmed - \$WOUND-ROLE-FINDER-VICTIM

\*\*\*\*\*  
GIRL builds a token which fills the VICTIM role of the \$WOUND script. Since IPP has inferred that the VICTIM of the \$ATTACK-PERSON and \$SHOOT scripts are the same as the VICTIM of \$WOUND, it also fills in those roles. Identifying these roles is integral to IPP's purpose of understanding the story, since an attack on a person can only be properly understood if the victim is known. As this person is important to the understanding of the story, IPP wants to acquire as much information as possible about her. Therefore, it looks back at the modifiers temporarily saved in short-term memory, 8-YEAR-OLD in this case, and uses them to modify the token built for GIRL. The age of the girl is noted as eight years. This information could easily be crucial to appreciating the interesting nature of the story.  
\*\*\*\*\*

AS : Skip  
SHE : Skip  
WAS : Skip and save  
BEING : Dull verb - skipped  
TAKEN : Skip  
TO : Function word  
SCHOOL : Normal token - SCHOOL  
YESTERDAY : Normal token - YESTERDAY

\*\*\*\*\*  
Nothing in this phrase is either inherently interesting or fulfills expectations made earlier in the processing of the story. So it is all processed very superficially, adding nothing to the final representation. It is important that IPP makes no attempt to disambiguate words such as TAKEN, an extremely complex process, since it knows none of the possible meanings will add significantly to its understanding.  
\*\*\*\*\*

AT : Function word  
STEWARTSTOWN : Skip and save  
COUNTY : Skip and save  
TYRONNE : Normal token - TYRONNE

Prediction confirmed - \$TERRORISM-ROLE-FINDER-PLACE

\*\*\*\*\*  
STEWARTSTOWN COUNTY TYRONNE satisfies the prediction for the place where the terrorism took place. IPP has inferred that all the scenes of the event took place at the same location. IPP expends effort in identifying this role, as location is crucial to the understanding of most stories. It is also important in the organization of memories about stories. A incidence of terrorism in Northern Ireland is understood differently from one in New York or Geneva.  
\*\*\*\*\*

Story Representation:

```
** MAIN EVENT **
SCRIPT $TERRORISM
ACTOR GUNMEN
PLACE STEWARTSTOWN COUNTY TYRONNE
TIME YESTERDAY
SCENES
SCRIPT $AMBUSH
ACTOR GUNMEN

SCRIPT $ATTACK-PERSON
ACTOR GUNMEN
VICTIM 8 YEAR OLD GIRL
SCENES
SCRIPT $SHOOT
ACTOR GUNMEN
VICTIM 8 YEAR OLD GIRL

SCRIPT $WOUND
ACTOR GUNMEN
VICTIM 8 YEAR OLD GIRL
EXTENT GREATERTHAN-#NORM*
```

\*\*\*\*\*  
IPP's final representation indicates that it has fulfilled its purpose in reading the story. It has extracted roughly the same information as a person reading the story quickly. IPP has recognized an instance of terrorism consisting of an ambush in which an eight year-old girl was wounded. That seems to be about all a person would normally remember from such a story.  
\*\*\*\*\*

[PHOTO: Terminated Sun 24-Jun-79 3:38PM]

As it processes a story such as this one, IPP keeps track of how interesting it feels the story is. Novelty and relevance tend to increase interestingness, while redundancy and irrelevance decrease it. For example, in the story shown above, the fact that the victim of the shooting was an 8 year-old increases the interest of the story, and the the incident taking place in Northern Ireland as opposed to a more unusual site for terrorism decreases the interest. The story's interest is used to determine how much effort should be expended in trying to fill in more details of the story. If the level of interestingness decreases far enough, the program can stop processing the story, and look for a more interesting one, in the same way a person does when reading through a newspaper.

#### 4. ANOTHER EXAMPLE

The following example further illustrates the capabilities of IPP. In this example only IPP's final story representation is shown. This story was also taken from the Boston Globe.

[PHOTO: Initiated Wed 27-Jun-79 1:00PM]

@RUN IPP

\*(PARSE S2)

Input: S2 (6 3 79) GUATEMALA

(THE SON OF FORMER PRESIDENT EUGENIO KJELL LAUGERUD WAS SHOT DEAD BY UNIDENTIFIED ASSAILANTS LAST WEEK AND A BOMB EXPLODED AT THE HOME OF A GOVERNMENT OFFICIAL POLICE SAID)

## Story Representation:

### \*\* MAIN EVENT \*\*

```
SCRIPT $TERRORISM
ACTOR UNKNOWN ASSAILANTS
SCENES
  SCRIPT $ATTACK-PERSON
  ACTOR UNKNOWN ASSAILANTS
  VICTIM SON OF PREVIOUS PRESIDENT
  EUGENIO KJELL LAUGERUD
SCENES
  SCRIPT $SHOOT
  ACTOR UNKNOWN ASSAILANTS
  VICTIM SON OF PREVIOUS PRESIDENT
  EUGENIO KJELL LAUGERUD
  SCRIPT $SKILL
  ACTOR UNKNOWN ASSAILANTS
  VICTIM SON OF PREVIOUS PRESIDENT
  EUGENIO KJELL LAUGERUD
  SCRIPT $ATTACK-PLACE
  ACTOR UNKNOWN ASSAILANTS
  PLACE HOME OF GOVERNMENT OFFICIAL
SCENES
  SCRIPT $BOMB
  ACTOR UNKNOWN ASSAILANTS
  PLACE HOME OF GOVERNMENT OFFICIAL
```

[PHOTO: Terminated - wed 27-Jun-79 1:09PM]

This example makes several interesting points about the way IPP operates. Notice that IPP has jumped to a conclusion about the story, which, while plausible, could easily be wrong. It assumes that the actor of the \$BOMB and \$ATTACK-PLACE scripts is the same as the actor of the \$TERRORISM script, which was in turn inferred from the actor of the shooting incident. This is plausible, as normally news stories are about a coherent set of events with logical relations amongst them. So it is reasonable for a story to be about a series of related acts of terrorism, committed by the same person or group, and that is what IPP assumes here even though that may not be correct. But this kind of inference is exactly the kind which IPP must make in order to do efficient top-down processing, despite the possibility of errors.

The other interesting point about this example is the way some of IPP's quite pragmatic heuristics for processing give positive results. For instance, as mentioned earlier, the first actor mentioned has a strong tendency to be important to the understanding of a story. In this story that means that the modifying prepositional phrase "of former President Eugenio Kjell Laugerud" is analyzed and attached to the token built for "son," usually not an interesting word. Heuristics of this sort give IPP its power and robustness, rather than any single rule about language understanding.

## 5. CONCLUSION

IPP has been implemented on a DECsystem 20/50 at Yale. It currently has a vocabulary of more than 1400 words which is being continually increased in an attempt to make the program an expert understander of newspaper stories about terrorism. It is also planned to add information about higher level knowledge structures such as goals and plans and expand IPP's domain of interest. To date, IPP has successfully processed over 50 stories taken directly from various newspapers, many sight unseen.

The difference between the powers of IPP and the syntactically driven parsers mentioned earlier can best be seen by the kinds of sentences they handle. Syntax-based parsers generally deal with relatively simple, syntactically well-formed sentences. IPP

handles such sentences, but also accurately processes stories taken directly from newspapers, which often involve extremely convoluted syntax, and in many cases are not grammatical at all. Sentences of this type are difficult, if not impossible for parsers relying on syntax. IPP is able to process news stories quickly, on the order of 2 CPU seconds, and when done, it has achieved a complete understanding of the story, not just a syntactic parse.

As shown in the examples above, interest can provide a purpose for reading newspaper stories. In other situations, other factors might provide the purpose. But the purpose is never simply to create a representation - especially a representation with no semantic content, such as a syntax tree. This is not to say syntax is not important, obviously in many circumstances it provides crucial information, but it should not drive the understanding process. Preliminary representations are needed only if they assist in the reader's ultimate purpose - building an appropriate, high-level representation which can be incorporated with already existing knowledge. The results achieved by IPP indicate that parsing directly into high-level knowledge structures is possible, and in many situations may well be more practical than first doing a low-level parse. Its integrated approach allows IPP to make use of all the various kinds of knowledge which people use when understanding a story.

## References

- [1] Cullingford, R. (1978) Script application: Computer understanding of newspaper stories. Research Report 116, Department of Computer Science, Yale University.
- [2] DeJong, G. F. (1979) Skimming stories in real time: An experiment in integrated understanding. Research Report 158, Department of Computer Science, Yale University.
- [3] Kaplan, R. M. (1975) On process models for sentence analysis. In D. A. Norman and D. E. Rumelhart, eds., Explorations in Cognition. W. H. Freeman and Company, San Francisco.
- [4] Marcus, M. P. (1979) A Theory of Syntactic Recognition for Natural Language, in P. H. Winston and R. H. Brown (eds.), Artificial Intelligence: an MIT Perspective. MIT Press, Cambridge, Massachusetts.
- [5] Riesbeck, C. K. (1975) Conceptual analysis. In R. C. Schank (ed.), Conceptual Information Processing. North Holland, Amsterdam.
- [6] Schank, R. C. (1975) Conceptual Information Processing. North Holland, Amsterdam.
- [7] Schank, R. C. (1978) Interestingness: Controlling inferences. Research Report 145, Department of Computer Science, Yale University.
- [8] Schank, R. C. and Abelson, R. P. (1977) Scripts, Plans, Goals and Understanding. Lawrence Erlbaum Associates, Hillsdale, New Jersey.
- [9] Wilensky, R. (1978) Understanding goal-based stories. Research Report 140, Department of Computer Science, Yale University.
- [10] Winograd, T. (1972) Understanding Natural Language. Academic Press, New York.
- [11] Woods, W. A. (1970) Transition network grammars for natural language analysis. Communications of the ACM. Vol. 13, p 591.

