

Chuck Rieger and Steve Small
 Department of Computer Science
 University of Maryland
 College Park, Maryland 20742

An approach to natural language meaning-based parsing in which the unit of linguistic knowledge is the word rather than the first rewrite rule is described. In the Word Expert Parser, knowledge about language is distributed across a population of procedural experts, each representing a word of the language, and each an expert at diagnosing that word's intended usage in context. The parser is structured around a coroutine control environment in which the generator-like word experts ask questions and exchange information in coming to collective agreement on sentence meaning. The Word Expert theory is advanced as a better cognitive model of human language expertise than the traditional rule-based approach. The technical discussion is organized around examples taken from the prototype LISP system which implements parts of the theory.

1. Background

In building a natural language understanding system, one of the first major decisions to be made is that of how to express and organize linguistic and conceptual knowledge for the parser. Depending upon the level of analysis desired (e.g., from simple keyword and key phrase recognition to full conceptual understanding in context), different methods suggest themselves. For the most part, models of parsing to this point have made the assumption that systems of rewrite rules are good media for expressing knowledge about sentence-level parsing. Examples of relatively successful systems and theories based on this assumption can be taken from every nearly every level of analysis, from the simple keyword analysis of PARRY [1], the syntactic analysis of LUNAR, up to more semantic and conceptual parsers such as those by Riesbeck [2], [3], and Wilks [4], [5]. Although the implementation styles of these models differ markedly, they share the view that rules of language understanding are best captured by rules which span relatively large sentence constituents (e.g., a phrase, a "picture producing" clause, a verb case framework). This assumption preordains a certain view of language comprehension by imposing structure at the sentence level and treating the words of the language as tokens which simply participate in comprehension by virtue of their inclusion in sentence and concept level rules.

While it is highly likely that any knowledge whatsoever about language can be expressed in the form of a system of rewrite rules, this form of representation defines a particular point of view in which individual words of the language mean nothing outside the system of rules which make reference to them. This seems peculiar, since we intuitively know that each word of our language is an object with an often very rich knowledge structure attached to it. We know many contextual uses of each word (sense), we know its morphology, perhaps its history in some usage, perhaps when we first learned it, its idiosyncratic uses, perhaps even idiosyncratic uses of it with respect to some particular speaker we know. Intuitively, there is a considerable amount of knowledge associated with each word.

One can easily argue that all such knowledge can be captured in a system of rules which make reference to the word; indeed, each rule we add to the system will incrementally extend our knowledge about the words to which it makes reference. To extend the system, one adds new rules and revises or deletes old ones. If asked what it knows about any given word, the system can report all patterns in which the word participates. During comprehension, if the patterns of the system have been written to include enough sentential and conceptual context, exactly the right rules will filter to the forefront to drive the parser in its extraction of meaning and/or syntax. Such a system is clearly theoretically adequate.

We believe, however, that the underlying assumptions of this traditional approach are not correct. Out of curiosity a while back, we posed the following question: What would the knowledge in a language comprehension model look like if organized around words rather than systems of rewrite rules? Superficially, this seems like an engineering matter; one symbol's forward pointer is another symbol's backward pointer! But, as we have found, a remarkably different set of issues arises, and a considerably different approach to parsing suggests itself when the model defines the unit of knowledge to be the word rather than the rewrite rule.

This paper describes our "word expert" view of natural language

comprehension. In the following sections, we describe why the theory seems more attractive than rule-based system, and we describe the architecture and operation of the Word Expert Parser, a LISP implementation of the ideas.

2. Word Experts

Our thinking on these matters was first stimulated by a simple observation about existing models. This observation was that no model (except for Wilks' model of preference semantics) seems ever to have made more than a token attempt at handling multiple senses of any given word in its vocabulary. Riesbeck's impressive parser, for example, while pioneering in many of its ideas about control via conceptual dependency schemata, was designed in a way that made it awkward to deal with highly ambiguous words; although the system did (and does) clever rule and expectancy switching on a limited set of examples of conceptual ambiguity, the system does not deal with the problem in generality. Wilks' system, on the other hand, is designed to deal with exactly this problem of word ambiguity, but at a more semantic (e.g., "things which are human eat things which are food"), rather than fully conceptual level where arbitrary world knowledge might be required for understanding. It is a mixture of Riesbeck's and Wilks' goals that most closely defines our goals, since we feel that word sense disambiguation in the context of open-ended world knowledge is of ultimate importance in a natural language system.

We therefore took the point of view that mechanisms for dealing with word ambiguity should be deeply built in to the model, perhaps even the most central mechanisms of the comprehension process. To gain some empathy for this point of view, consider almost any common English noun, verb or adjective. With twenty minute's thought, and perhaps a few reminders from Webster's, it is usually possible to identify dozens of different usages of that word. (We have looked at words like "take", "make" and "give" in some detail; each has dozens of distinct usages.)

How do we approach the problem of representing all the contextual usages of such highly ambiguous words? The traditional approach would be to write one rule for each usage. Each such rule would contain enough contextual probes so that it would fire at exactly the appropriate moments. But there are several negative features of this approach. First, each rule must be self-contained, and have enough information to fire only when the appropriate sentential and conceptual context pertained. This quickly leads to large, often highly redundant context descriptions in each rule. Second, some uniform interpreter capable of rule arbitration must exist to select the "most appropriate" competitor from among the several rules which will typically fire as potential interpretations. Third, being separate from all the other competitors, each rule must be able to place an absolute judgement on its relevance to the context at hand; there is no convenient way to have a rule decide on its relative appropriateness, with respect to its competitors.

These observations pointed away from rule systems, and toward the idea of writing more unified word experts. This led us to the notion of word (sense) experts. In our present model, each word expert is a procedural entity cognizant of all the possible contextual interpretations of the word it represents. When placed in a run-time context (i.e., informed of its position in the sentence, and given access to the neighboring experts and to a database of world knowledge context at that point in the comprehension of the larger fragment of which the sentence is a part), a word expert should be capable of issuing an orderly progression of inquiries about its context, and making a set of decisions based on the responses (or lack thereof) to its queries which lead to a selection of the best relative interpretation of its word in that context. Each word expert is, in effect, a unified compilation of what otherwise would be

*The research described in this report is funded by the National Aeronautics and Space Administration under grant number NSG-7253. Their support is gratefully acknowledged.

a collection of rules describing the contextual variants of the word. By unifying them, we create a central knowledge structure which can pose more concise questions to differentiate contextual usages, and which superimposes a unifying framework on all possible interpretations of a word.

In further pondering the nature of word experts, it occurred to us that what we are really talking about is not so much words, but rather lexical units that are capable of contributing meaning or structure to a sentence. From this idea, it becomes clear that we want experts not only for words, but also for common morphemes ("ing", "-ed", etc.), and possibly even for punctuation. The sentence (and a host of others like it, studied heavily in linguistics circles) "The man eating spaghetti growled," illustrates very clearly the desirability of an "ing" expert. Looking to its left and right experts (or the concepts they may already have constructed), "ing" should be able to decide how the concepts referenced by "man" and "eat" in this sentence best fit together. Because of observations like this one, we have come to take a broad view of sense experts that will allow meaning-contributing units other than words.

Parser Overview. Naturally, adopting the word and morpheme rather than the rule as the base unit expressing knowledge leads to differences in the organization and control of the parser. Rather than a uniform rule interpreter/applier, the Word Expert Parser is more a model of coroutine control and intercommunication. As we are about to describe, in the Word Expert Parser, the parse of each sentence involves the paging in of all required experts, the initialization of community and private workspaces, then the control of the numerous experts as they execute, suspend themselves, ask questions of one-another, and contribute fragments of meaning to the final interpretation of the sentence. Because of its organization, the Word Expert Parser must define not only the structure and contents of each expert, but also a cognitively plausible control paradigm for coordinating the experts.

The body of the paper is a description of our present model of word experts and their control. Briefly, each word expert is a generator-like coroutine whose decision logic is organized as a discrimination net. Terminals of the net are the distinct usages of the word (its senses), while non-terminals are multiple-choice questions which probe the run-time environment. Also at each non-terminal node is control information describing what to do in case the question is not yet answerable (e.g., make an assumption forward a suggestion to another expert, suspend temporarily). Meaning representation of the final interpretations (i.e., the meaning patterns stored at the terminal nodes of the experts) has not been an issue in the development of the parser to this point. We presently use predicates that are descriptive of each of the different senses of the word, and have not made attempts to develop a consistent representation. Although the nature of the parser may well be affected by eventual commitments to one particular representation, we have so far been able to develop the ideas about word expert structure and content, and about parser control without regard for the form of the final representation.

We hope to convey our intuitive feeling that word level organization of knowledge about language is both natural and powerful. In addition to dictating an interesting style of parser control, solving some important problems about word sense ambiguity, and addressing the issue of relative (rather than absolute) selection of the most appropriate interpretation of each word, the word expert organization leads to some other interesting ideas about language, particularly about language learning. We discuss several of these ideas in the final section. Additional discussion of the Word Expert topic can be found in [6], [7], [8], and [9].

3. The Word Expert Parser

The Word Expert Parser is a model of the human interface between an external language of inter-human communication, and an internal representation language of concepts and ideas. As a human reader processes a sentence, various conceptual conglomerates are created that provide the key to his understanding [10]. The process of creating these structures is as important as the structures themselves -- not only do stages in the processing of a sentence often depend on aspects of earlier processing, but sometimes the meaning of a sentence is explicitly influenced by how it is processed [11]. This should be quite clear for spoken language, but is so for written text as well if any degree of ungrammaticality or fragmentation is to be tolerated. The Word Expert Parser is a model of conceptual language analysis that strives for theoretical perspicuity of process.

3.1. Model Overview -- A Simple Example

We will first describe the execution of the Word Expert Parser on the phrase "the heavy rain" to illustrate the model's procedural framework on a simple example. Initialization of the parser commences with the creation of a sentence level workspace for this particular input. This workspace consists of a collection of word bins and their associated word sense discrimination experts (word experts or sense experts). A word bin is a repository for information about a single word (or sub-lexical morpheme) of the input, and its corresponding word expert contains the word-specific (and procedural) linguistic knowledge that directs this particular execution of the parser. Each word expert is capable of disambiguating its word in any context, and interactions among word experts deterministically disambiguate the meaning of the sentence.

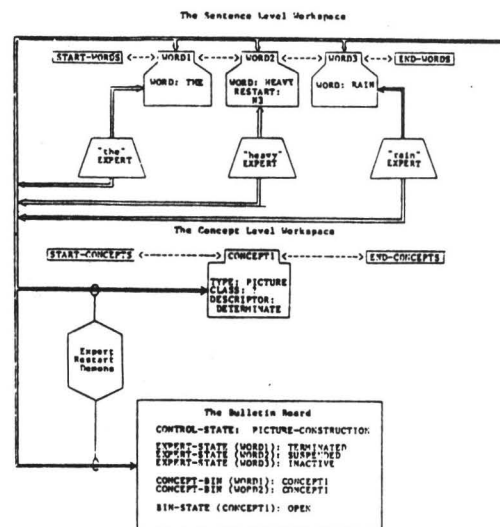


Figure 1: Parser Workspace

The "the" expert runs. The parser begins execution with the word expert for "the", which is the first on the RUN-ME queue (initialized to the input sentence). The flow of control now becomes unpredictable (in theory), as the word experts themselves completely determine control state changes and conceptual actions of the model. The "the" expert tries to determine its role in the sentence, knowing only that it is the first expert to run. Since the role of "the" is usually obvious (as it is here), its expert immediately changes the control state description of the parser from "new sentence" to "picture construction". This state involves the creation of a new conceptual object (called a concept bin) and associating with it a (partial) description. The expert proceeds to create this concept, mark it "determinate", and terminate.

The "heavy" expert runs. The RUN-ME queue now consists of "heavy" and "rain" and the expert for "heavy" is removed and started executing. Since a concept bin is currently active, "heavy" knows to contribute its conceptual descriptions to that bin. If no appropriate concept bin is active when an expert runs, it creates one (as "the" almost always does). If the word describes the same concept as the one preceding it in the sentence, on the other hand, its expert performs as the "heavy" expert does here. This first step completed, the word expert must now discriminate among the many senses of "heavy" and resolve on a single contextually accurate word sense. Since it has not yet seen the word to its right (as we have!), it cannot discriminate among the senses meaning "having large mass", "being in large quantity", or "being serious or emotional".

This ambiguity causes the expert to suspend its execution temporarily until it has seen the word to its right. For an expert to suspend, it must specify the conditions upon which it is to be reawakened and the location in the RUN-ME queue at which it should be placed. The "heavy" expert specifies resumption when the word expert to its right terminates, the end of the sentence reached, or a new concept created. Before suspending, however, this expert also posts a memo to the experts on its right, telling them to expect one of the conceptual object types that "heavy" could describe (e.g., "abstract object"). Expectations in the model are treated differently from constraints -- they provide guidance but are not binding. The "heavy" expert is now suspended, and the execution state of the model corresponds to Figure 1.

The resumption of suspended word expert processes is through spontaneous mechanisms which respond to both conceptual information and changes in the control states of model processes. The restart conditions may be quite complex in general, and have been modeled after the spontaneous

computations of Rieger [12]. When a restart condition involves process control states, it is called a control state demon. For control state demons to be practicable, a central tableau of information about the control structures of the model must be maintained. In the Word Expert Parser, the bulletin board is the center of this control knowledge. Control state descriptions of word experts, the parser itself, and the concept bins are delineated on the bulletin board. In addition, the model's expectations, such as those contributed by the heavy expert above, are posted on this board. By making this knowledge of process available to the word experts, control states and expectations of various model components can affect the execution of the others.

The "rain" expert runs. Processing now continues with the expert for "rain" which has become the first on RUN-ME. This expert process contributes its conceptual information to the existing concept bin, and as a consequence of "heavy's" expectations and "rain's" simple usage, immediately determines its sense and terminates. Termination of rain causes the spontaneous resumption of the "heavy" expert, which continues processing where it left off. Since two of "heavy's" three senses become meaningless in describing rain, the "heavy" expert resolves upon being in large quantity and terminates. This causes the model's single concept bin to close, completing the parse.

The control possibilities are clearly quite complex, and the processing is not always going to give as clear and successful parse as it did on "the heavy rain". However, the Word Expert Parser is a model of cognitive processing, and as such, the state of the parser after incomplete processing must be of psychological appeal. The parser's actions on ambiguous input (extremely rare when considered contextually) demonstrates its cogency -- the end result would be a collection of partially completed suspended processes, coupled with a set of reawakening conditions specifying the reasons for the ambiguity (actually, the requirements for its resolution).

3.2. Workspace Representation

The sentence level and concept level workspaces of the model are represented as partitions of a PLANNER-like associative database. The data in a bin of either workspace is therefore accessible by any of its contents, and in fact, declarative representations can be retrieved associatively from any bin. Descriptors of a particular conceptual object can be accessed by the internal name of the concept (its bin), and more significantly, the objects in the concept-level are retrievable by their descriptions. In this way, a word expert determines the nature of any concepts in the workspace that could influence its actions.

The Sentence Level Workspace. In the initial state of the parser, the word bins contain two pieces of information -- the literal word corresponding to the name given it by the parser (e.g., "the" for WORLD) and the left and right neighbors of the word in the input sentence. These word bins acquire a more subtle and interesting role, however, as the word expert processes are initialized. At this point, each word bin contains the restart information (stack frame [13]) for the process, including its local variables and bindings, and instruction pointer. By representing this information in the word bins, a word expert can easily constrain another by simply altering its local variables or the place that it starts (or resumes) execution.*

The Concept Level Workspace. Each concept in the parser has a type which determines the kind of information found in its concept bin. The parser counts three such conceptual object types, pictures, settings, and events. A picture concept [14] involves physical and abstract objects and their descriptions. A setting is a time, location, or situation. The creation of a setting is frequently introduced by a preposition in the sentence. For example, "in the morning", "in the house", and "in trouble" are examples of time, location, and situation examples (respectively) of setting concepts. Event concepts consist of conceptual case schemata.

The Bulletin Board. Posted on the bulletin board is a collection of control state information about the processing by the model. The process state of each word expert in the model, the status of each conceptual object, and a description of the state of the entire parser are found on this board. Any spontaneous computation triggered by some indication of control state is attached to a memo posting function (or server [15]) on the data channel to the bulletin board.

The Word Expert Parser can be in one of the following global states: NEW-SENTENCE, PICTURE-CONSTRUCTION, SETTING-CONSTRUCTION, EVENT-CONSTRUCTION, or UNKNOWN. These designations describe the actions of the parser as a whole and are based on lower-level descriptions of expert states and concept states. A concept can be UNOPEN, OPEN, or CLOSED, depending on its state of completion. Word experts are processes and are therefore described by the states INACTIVE, SUSPENDED, and TERMINATED. As work on the Word Expert Parser continues [16], refinements of these state descriptions will emerge, clarifying the strength of our work as a process model of human cognitive behavior.

*The modularity of the experts requires that external manipulation of restart pointers be carefully controlled. Each word expert contains a descriptive header that explicitly states how the expert may be constrained.

3.3. Word Sense Discrimination Experts

The principal knowledge structure in the Word Expert Parser is the word sense discrimination expert. A word expert is a representation of the linguistic knowledge (procedural) needed to disambiguate the meaning of a single word in any context. A word's context includes both sentential and conceptual information -- not only are the immediate lexical neighbors of a word important, but conceptual knowledge of the entire text and even the physical world are frequently crucial in understanding it. A word expert performs its discrimination processing by asking context-related questions of the various components of the parser or the story comprehension model in which it is embedded (see [7] and [8] about the GRIND story understanding model).

The important aspects of a word expert are the questions that it asks (and of whom it asks them) and the actions that it performs upon receiving the answers. Sometimes an action is taken because no answers were forthcoming, namely, suspending execution until the information is available or until control states are reached at which point the availability is likely (or non-availability is significant). Other word expert actions include creating and building up concepts, constraining other experts, posting expectations on the bulletin board, and changing the control state of the parser.

3.3.1. Word Expert Structure

Each word expert exists in two forms in the model -- a declarative representation which is a directed acyclic graph and called a word sense discrimination network (or sense net), and a procedural representation like a CONNIVER generator (or stack group [17]) which is called a word expert process. Both representations are briefly discussed here, and their *raison d'être* explained.

Network Representation. A word sense discrimination network is a graphic representation of the contextual disambiguation of a single word. This structure was motivated by the central theoretical notion of the sense net theory of language, that each word of English carries with it the information necessary to determine its contextual meaning. Each node in a sense net represents an ordered set of decisions that converge on a single appropriate sense of a word.

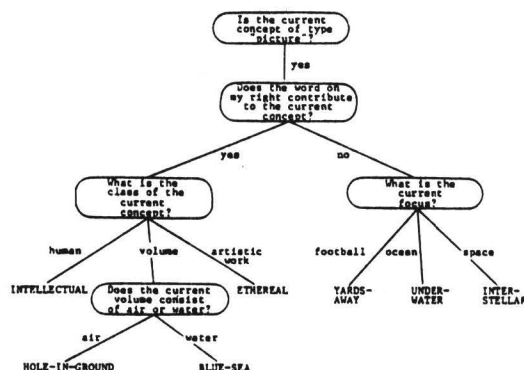


Figure 2: Sense network for "deep"

An example network for "deep" is shown in Figure 2, and shows several of its senses. The first two questions of the network are aimed at determining if "deep" denotes an object ("the deep") or a description (e.g., "the deep thought"). As a question is answered, one branch of the graph is favored over others, and the number of possible senses of a word is reduced. Finally, a terminal node is reached, and a single word sense chosen.

The possibility arises that a question cannot be answered because it depends on information from other networks that is not yet available. In such a case, a sense net must remember its location in the graph and wait for the information it needs to continue. Another aspect of inter-network activity is when a sense net determines constraints on another -- a network waiting for certain data is told to restart at its root node, or told to resume at a different node than it expects. The influence of one sense net on another reflects the crucial contextual nature

of their word sense disambiguation tasks.

Process Representation. The Word Expert Parser compiles each sense discrimination network into an executable process when it is first called to perform its task. Although every word expert has a representation as a sense network, this graphic representation lacks important aspects of a true executable expert process. In particular, whereas the details of an expert's suspension and resumption, or interactions with other experts, may seem irrelevant at certain levels of explanation, they are in reality quite significant to the view of the parser as a process model of human language analysis. The network representation also fails to capture the important side-effect actions performed by an expert. The structure and execution of word experts therefore will be outlined through the process representation. Use of an expert's implicit graph structure as a metaphor, however, will certainly be useful.

The compilation of a word expert is significant in that certain nodes of a sense network may be pruned away before it becomes an expert process. In this way, contextual information about the plot of a story, its characters, the current focus (see [18]), and so forth, can be used to eliminate a word's irrelevant senses. In reading "the ship cruised" in a nineteenth century story, for example, a sense of "ship" denoting something which travels between planets can be pruned away immediately. Certain constraints cannot be made a priori, however, and in such cases, an already compiled process must be manipulated.

Word expert processes are implemented in the model in a manner resembling CONNIVER generators. They can be suspended in the middle of execution and resumed spontaneously according to a variety of trigger patterns. These include not only the standard patterns of conceptual data, but data about the control states of other processes as well. Processes can communicate by affecting the local variables of another, including those which tell the process where to start (or resume) execution. This effectively prunes the implicit graph structure of the expert. Word expert processes also provide information to other experts by posting control data and expectations on the central bulletin board.

3.3.2. What Word Experts Can Do

The role and significance of word experts in the model are most easily understood through an examination of both their internal actions like the questions they ask, and their side effects, actions that affect the rest of the model. A word expert can ask questions about the other words of the sentence, concepts introduced, expectations, the control state of the parser, or the control states of other experts. An expert can affect the data found in word bins, concept bins, the bulletin board, and the process restart queue of spontaneous computations.

Internal Process Actions. The fundamental internal actions of a word expert are question-asking and branching. One of the important contributions expected in development of the Word Expert Parser is the taxonomy of context-probing questions asked by experts. The nature of some of these questions is emerging and is of several varieties -- lexical, conceptual, global, and control. This last category consists of questions about the state of the cognitive processing task thus far, representationally encompassing the states of word experts, the parser state, and the states of the bins.

The existence of a particular word to the immediate left or right of a given word can often discriminate among its senses. A word expert asks about the nature of the words arbitrarily far to its left and right in the sentence. Furthermore, it can ask about a word's senses (in totality or only those still remaining) through the word's conceptual categories, subclasses of these categories, or the explicit names given to its senses. When a word expert is constrained, the pruning may be done along any of these same dimensions.

Conceptual information is usually very important in discriminating among word senses, as are the posted expectations about this information. The concept bin to which the executing process contributes also contains conceptual data from other experts which is often crucial to understanding the current word. A simple example was given in the model overview, when the "heavy" expert needed information from its own concept bin (i.e., data regarding "rain") before it could determine the correct sense. The concepts to the left and right of the active concept are useful in the same way. It is likely, however, that in the initial left to right scan of the sentence, concept bins to the right of a given one may not yet exist. This fact indicates that a strict ordered scan of a sentence is sufficient to parse it only if no expert needs information from concepts to its right. The organization of the Word Expert Parser therefore supports our common sense intuition about conceptual parsing by humans.

The Word Expert Parser is embedded within the GRIND story comprehension model, and as such, has access to the global world of a children's story. Relevant aspects include the personalities of the characters, interactions among characters, the plot line, anaphoric references, and current foci of interest. The plot of a medieval story, for example, might tell the reader that "knights", "lords", "knights", and "workers" take certain strict interpretations in local contexts. Or the focus of understanding in a story might be the Algonquin Round Table, in which case "the deep" ought to be interpreted as referring to those intelligent persons who regularly ate lunch together at that hotel. Global context significantly influences the understanding of certain sentences of texts.

Control State Descriptions. A new and important aspect of the Word Expert Parser is that control information is accessible to (and often used by) the discrimination processes. The fact that the expert to the immediate left of the executing expert is suspended, or that the expert to its right has not yet executed, is often the central datum enabling its successful termination. The state of a concept bin is also useful information, particularly in deciding where one concept ends and the next begins. A concept bin is closed only when the conceptual notion it represents is complete; that is, when a change in its data requires the kind of severe backtracking that humans perform extremely rarely (on so-called "garden paths"). In the example parse of "the heavy rain", the control state of the parser aided the executions of the "heavy" and "rain" experts. Control state knowledge is fundamental to the successful conceptual analysis of most sentences. In both the internal nature of the expert processes and their relationship to the entire model (suspension and resumption), the self-examining knowledge of control flow and processing states is fundamental.

Process Side Effects. Conceptual analysis is an inherent part of natural language understanding but not the whole part. As such, word sense discrimination by the Word Expert Parser must provide the information it acquires to other components of a large understanding model to complete the task. Assimilation of new data into a picture of the world (see [19] for a discussion of modeling belief systems) by making the necessary inferences, disambiguating references, and so forth (see [8]), constitute the other facets of understanding a text. The side effects of the processing by a word expert provide the information required for further stages in the conceptual analysis phase of understanding, as well as conceptual data for the subsequent phases.

Conceptual knowledge is augmented by creating new concepts and storing new facts about old concepts. A word expert's first task is to determine the conceptual notion to which it contributes. If a concept bin is active at the time, the expert could choose to contribute to it. This would mean that the word it is trying to disambiguate participates in the same concept as the word to its left in the sentence. If no concept bin is active, or if the expert determines that its word starts a new concept anyway, the word expert would open a new one. It is possible (in certain circumstances likely) that an expert would not know which bin to contribute to -- in this case it would suspend its execution until the expert to its right has executed (this is usually enough).

When a word expert has determined its concept, it may store facts there. If it opens the bin itself, it must provide a type (picture, setting, or event) to the bin. Otherwise, it may provide the concept's class, or augment its descriptor. An expert that completes a concept may close its concept bin, indicating that its word is the last (rightmost) one contributing to that concept. If all experts participating in the concept have not yet terminated when one of them attempts to close it, however, the model will plant a demon to close the bin when the last of the relevant experts terminates.

A word expert also affects certain internal data structures, useful solely by the parser. These are extremely important and include suspending itself and constraining other experts. The example execution of "the deep pit" (see below) demonstrates this constraining action -- since neither "deep" nor "pit" can determine its correct sense on the first try, "deep" finally tells "pit" to be one of the things that "deep" can describe, namely, a hole in the ground or a person. It does this by changing the place in "pit's" network where it begins execution, thus making it believe that it had successfully discriminated farther than it really had. After constraining it in this way, the parse can complete. If "pit" had not yet been compiled, the sense net itself could have been pruned before compilation.

An expert suspends its execution if the model's lack of certain information keeps it from proceeding. When this occurs, the expert stops execution and plants a demon to wake it up when the fact it needs becomes available. These facts can be of any variety, including data regarding the states of other processes. A typical suspension consists of an expert process needing something from a not-yet-executed process to its right, and waiting until it has executed (SUSPENDED or TERMINATED) to resume. Another example in "in the morning" involves a setting, where the "in" network waits until a picture bin is closed before it resumes. At that time, it can complete its own setting concept, knowing the class and description of the picture to its right. As a result of these suspensions and spontaneous resumptions, the range of control flows in the conceptual analysis model is wide and complex -- our confidence in the model's cogency is only strengthened by this variety.

3.4. Example: "the deep pit"

The Word Expert Parser is illustrated below through the example phrase "the deep pit". An actual trace of the parser's execution is presented, accompanied in the righthand column by some explanatory annotations. The example phrase causes the creation of a single concept of type "picture". The word sense ambiguity of the component words causes a number of expert suspensions and resumptions, demonstrating the strength of the system.*

Execution of "the deep pit" is an interesting example of the complexity of control flow in the Word Expert Parser. Since both "deep" and "pit" have more than a single sense, they must exchange pieces of information about themselves before either can succeed at determining its sense.

```
EVAL: (ZPARSE: THE DEEP PIT)
CONTROL-STATE: NEW-SENTENCE
VALUE: T
```

```
EXECUTING 'THE' EXPERT: WORD1
****TRANSLATING SENSE EXPERT FOR WORD1
**EXPERT COMPILE TIME: 56
**OPTIMIZATION TIME: 1237
****EXPERT FOR WORD1 TRANSLATED
----> AT NODE: N1
NEW CONCEPT: CONCEPT1 TYPE: PICTURE
CONTROL STATE: PICTURE-CONSTRUCTION
EXPERT TERMINATED: WORD1
```

```
EXECUTING 'DEEP' EXPERT: WORD2
****TRANSLATING SENSE EXPERT FOR WORD2
**EXPERT COMPILE TIME: 281
**OPTIMIZATION TIME: 5808
****EXPERT FOR WORD2 TRANSLATED
----> AT NODE: NO
----> AT NODE: N1
EXPERT SUSPENDED: WORD2
```

```
EXECUTING 'PIT' EXPERT: WORD3
****TRANSLATING SENSE EXPERT FOR WORD3
**EXPERT COMPILE TIME: 307
**OPTIMIZATION TIME: 5852
****EXPERT FOR WORD3 TRANSLATED
----> AT NODE: NO
----> AT NODE: N1
----> AT NODE: N2
**RESTART DEMON FOR WORD2 TRIGGERED:
* (*EXPERT-STATE* WORD3 SUSPENDED)
**EXPERT FOR WORD2 QUEUED FIRST
EXPERT SUSPENDED: WORD3
```

```
EXECUTING 'DEEP' EXPERT: WORD2
----> AT NODE: N1
----> AT NODE: N4
**RESTART DEMON FOR WORD3 TRIGGERED:
* (*EXPERT-STATE* WORD2 SUSPENDED)
**EXPERT FOR WORD3 QUEUED LAST
EXPERT SUSPENDED: WORD2
```

```
EXECUTING 'PIT' EXPERT: WORD3
----> AT NODE: N2
----> AT NODE: N3
CONTROL STATE: UNKNOWN
**RESTART DEMON FOR WORD2 TRIGGERED:
* (*EXPERT-STATE* WORD3 TERMINATED)
**EXPERT FOR WORD2 QUEUED FIRST
EXPERT TERMINATED: WORD3
```

```
EXECUTING 'DEEP' EXPERT: WORD2
----> AT NODE: N1
----> AT NODE: N3
----> AT NODE: N2
EXPERT TERMINATED: WORD2
CONCEPT CLOSED: CONCEPT1
```

```
VALUE: T
```

```
EVAL: (ZSCAN 'CONCEPT1)
(*DESCRIP* (SIZE-DIMENSION DEPTH) WORD2)
(*DESCRIP* (SIZE LARGE) WORD2)
(*DESCRIP* (VOLUME-TYPE HOLE-IN-GROUND) WORD3)
(*DESCRIP* (CLASS VOLUME) WORD3)
(*DESCRIP* DETERMINATE WORD1)
(*CONCEPT-CLASS* VOLUME)
(*CONCEPT-TYPE* PICTURE)
```

```
VALUE: CONCEPT1
```

*Execution traces of the Word Expert Parser are available from the authors. All examples referred to in this paper as well as later examples of full sentence parses are obtainable on request.

The ZPARSE function initializes the parser workspace by retrieving (from a disk file) the word experts for the input phrase, and creating for each one an associated data repository called a "word bin". Each word expert is placed on the RUN-NE queue, and then executed in turn, with the experts themselves eventually determining the flow of control in the model.

The word expert for "the" is translated from a network representation into a coroutine, which is then executed. It creates a new concept of type PICTURE, changes the parser state to PICTURE-CONSTRUCTION, and terminates.

The "deep" expert is translated and started executing. After a short time however, it needs information from the word expert to its right. Since the "pit" expert has not yet run at all, "deep" temporarily suspends execution until "pit"'s word expert has SUSPENDED or TERMINATED. Although permitted to do so, "deep" posts no expectations and makes no constraints, thereby insuring that the expert for "pit" has full freedom.

"Pit" now runs, but cannot discriminate between its two noun senses of fruit pit and "hole in the ground". It suspends its execution until something happens to help resolve its word sense deadlock. Since "pit" has now executed, "deep" is spontaneously reawakened.

Resumed by the suspension of the "pit" expert, the "deep" expert now says: "I gave the word to my right a chance to be what it wanted and it had no idea. Now I'll constrain it to be something that I can describe -- a hole in the ground or a person." At his point, therefore, "deep" constrains "pit" and suspends itself until "pit" terminates. Since "pit" was waiting for exactly this sort of activity, one of its restart demons fires, and the word expert for "pit" is queued up.

"Pit" now runs for the last time, its constraints focusing the discrimination. Since "pit" terminates, the restart demon for "deep" is triggered, and "deep" is queued.

Finally, "deep" runs for the last time, terminating with the sense of "large volume". Although "pit" tried to close its concept bin, "deep" still had not terminated, and so a demon was planted to close the concept when "deep" finished. Thus, the picture concept is now closed and the parse complete.

As the Word Expert Parser analyzed the input phrase, it created and augmented a data repository of conceptual information. This "concept bin" contains description of the meaning content of the phrase as well as various pieces of control state data. The concept bin for the deep pit is called CONCEPT1 and contains several descriptors which together describe a large volume of air in the ground.

4. Summary and Conclusions

We have mapped out a theory of organization and control for a meaning-based language understanding system. In this theory words, rather than rules, are the units of knowledge, and assume the form of procedural entities which execute as generator-like routines. Parsing a sentence in context demands a control environment in which these experts can ask questions of each other, forward hints and suggestions to each other, and suspend. Our theory is a cognitive theory of both language representation and parser control. It has stemmed from feelings about the fundamental inadequacies of rule-based theories.

In addition to the advantages already discussed, the word expert approach enjoys a number of other pragmatic and theoretical advantages. By way of conclusion, we enumerate what we feel are the important theoretical and practical characteristics of the theory.

1. It allows for the modular growth of language knowledge in the implemented model. Unlike a rule-based system (where obscure interactions and dependencies among seemingly unrelated rules can occur), each word expert is a relatively perspicuous, central aggregate of knowledge about one easily identified unit of language knowledge, i.e., the word. One can look at that structure and more easily say whether or not it takes any given usage into account. If it does not, it is clearer where knowledge about that new usage should be grafted in.
2. It encourages a healthy type of redundancy in the codification of language knowledge. When writing a word expert, one's goal is to create a stand-alone agent able to function in a wide variety of run-time contexts. We have noticed that this nearly always leads to bi-directional handshaking when two experts interact; they come to some mutual agreement because they find common ground. This type of redundancy is healthy, since it is then less likely that the shortcomings of one expert will block the comprehension of the sentence or concept in which that expert occurs.
3. It suggests some approaches to language acquisition. Since each word expert reflects the state of knowledge about a given word, when new usages of the word are perceived, rather than writing absolute rules to describe the new context, it is possible to grow a new branch within the word expert. This branch need only capture one relative difference between the existing usages and the new usage, an intuitively simpler and safer operation than generating an entirely new rule. Since the scope of the change is intuitively smaller, it is less likely to interfere with other experts in the system.
4. It encourages the codification of idiosyncratic uses of language by providing uniform data structures as focal points for that diverse knowledge. If, for example, there is an esoteric sense of the word "take", one need only sprout a branch of the "take" expert's decision logic to include it. Because of the branching logic organization of the experts, little overhead is incurred by adding special cases. We have felt all along that an organization which encourages knowledge about individual words to grow to large proportions without degrading the efficiency of the system at large is highly desirable.
5. It adapts well to largeness. Even if there are 100,000 word experts, each of which on the average requires 1000 words of computer memory, actual memory space required to parse any given sentence will be small. This is because the parser requires only the experts involved in the sentence at hand to be in core. This is a modest requirement when compared to large rule systems where all the rules must be kept in core at once (for efficiency), or where potentially complex paging systems must be devised to interact with a rule system on mass storage. While this is strictly an engineering issue, it has immediate consequences on today's computers!
6. It forces the parallel development of cognitively plausible theories of sequential, routine-like process models of humans. If the assumption about word (rather than rule) primacy in humans is correct, we stand a chance of making reasonable conjectures about flow of control in a human language understander. That is, some fairly crisp pictures of word expert control follow directly from the structure of the word experts themselves. As we discover how to control the experts in a LISP model, we may also be discovering how humans think when comprehending language.

Although much remains to be worked out, we find ourselves increasingly convinced of the appropriateness of the word expert approach. In the near future, we hope to be able to demonstrate the system at work on complete sentences, and hope to provide convincing evidence that large experts with many word usage alternatives are practical to develop and coordinate.

ACKNOWLEDGEMENTS

The authors would like to thank the following people for interesting discussions that helped refine the ideas presented in this paper: Phil Agre, Milt Grinberg, Phil London, Jim Reggia, and Rich Wood. In addition, Steve Small wishes to thank Professor James Meehan of the University of California at Irvine and Professor Robert Simmons of the University of Texas at Austin for comments on the research proposal which led to the parser implementation discussed here.

REFERENCES

- [1] Colby, K., Artificial Paranoia, Pergamon Press, 1975.
- [2] Riesbeck, C., Computational Understanding: Analysis of Sentences and Context, AI-Memo 238, Stanford University, 1974.
- [3] Riesbeck, C. and R. Schank, Comprehension by Computer: Expectation-based Analysis of Sentences in Context, Research Report 78, Yale University, 1976.
- [4] Wilks, Y., Preference Semantics, AI-Memo 206, Stanford University, 1973.
- [5] Wilks, Y., Making Preferences More Active, AI-Report 32, University of Edinburgh, 1977.
- [6] Small, S., Conceptual Language Analysis for Story Comprehension, Technical Report 663, University of Maryland, 1978.
- [7] Rieger, C., GRIND-1: First Report on the Magic Grinder Story Comprehension Project, Discourse Processes, vol. 1, no. 3, 1978.
- [8] Rieger, C., Five Aspects of a Full Scale Story Comprehension Model, Associative Networks -- The Representation and Use of Knowledge in Computers, Fiedler (ed.), Academic Press, 1979.
- [9] Rieger, C., Viewing Parsing as Word Sense Discrimination, A Survey of Linguistic Science, Dingwall (ed.), Greylock Pub., 1977.
- [10] Adams, M. and A. Collins, A Schema-Theoretic View of Reading, Technical Report 32, Center for the Study of Reading, 1977.
- [11] Lakoff, G., Some Remarks on AI and Linguistics, Cognitive Science, vol. 2, no. 3, 1978.
- [12] Rieger, C., Spontaneous Computation in Cognitive Models, Cognitive Science, vol. 1, no. 3, 1977.
- [13] McDermott, D. and G. Sussman, The Conniver Reference Manual, AI-Memo 259a, Massachusetts Institute of Technology, 1974.
- [14] Schank, R., Conceptual Dependency: A Theory of Natural Language Understanding, Cognitive Psychology, vol. 3, no. 4, 1972.
- [15] Bobrow, D. and T. Winograd, An Overview of KRL, A Knowledge Representation Language, Cognitive Science, vol. 1, no. 1, 1977.
- [16] Small, S., Word Expert Parsing, Proceedings of the 17th Annual Meeting of the Association for Computational Linguistics, 1979.
- [17] Lisp Machine Group, LISP Machine Progress Report, AI-Memo 444, Massachusetts Institute of Technology, 1977.
- [18] Grosz, B., The Representation and Use of Focus in Dialog Understanding, University of California at Berkeley, 1977.
- [19] London, P., Dependency Networks as a Representation for Modeling in General Problem Solvers, Technical Report 698, University of Maryland, 1978.