HANDLING CONCEPTUAL AND STRUCTURAL AMBIGUITIES IN DIRECT MEMORY PARSING

Ronald A. Sumida
Michael G. Dyer
Margot Flowers

April 1987
CSD-870018

Handling Conceptual and Structural Ambiguities in Direct Memory Parsing*

Ronald A. Sumida
Michael G. Dyer
Margot Flowers

Artificial Intelligence Laboratory
Computer Science Department
University of California
Los Angeles, CA, 90024

Abstract

This paper discusses a method for dealing with conceptual and structural ambiguities in natural language text. The approach presented in this paper features a constrained marker passing mechanism to search for both phrasal and role-level connections between words in the input, a short-term memory structure for storing these connections, and the incorporation of syntactic information using a phrasal approach. The short-term memory structure is used both to constrain search and to represent possibly conflicting interpretations of the input.

1. Introduction

Human readers are capable of understanding texts which are potentially misleading, both conceptually and structurally. Consider the following sentence:

S1. John put the pot on the stove.

After reading this sentence, it appears that John is using a cooking-container in the context of cooking on a stove. However, suppose the next sentence were:

S2. He picked it up and smoked it.

After reading S2, it is now apparent that in the first sentence, John was using the stove as a supporter (or lighter) for the marijuana cigarette (not a cooking pot). In addition, S1 and S2 are potentially ambiguous at the structural level, e.g. <X picked it up> could mean <X

*This research is supported in part under a contract to the second two authors by the JTF program of the DoD, monitored by JPL.

learned new information>, while <X put object on> could mean <X wear object>.

In this paper, we present a method for conceptually analyzing (i.e. parsing into a conceptual representation) sentences such as S1 and S2 above. This method views conceptual analysis as the process of finding appropriate interpretations in memory based on conceptual, structural, and contextual information associated with each input element. Our conceptual analyzer, called CAIN, operates by searching for both phrasal and role-level connections between words in the input, storing the connections found into a short-term memory structure (STMS) and using this structure both to constrain search and to represent possibly conflicting interpretations of the input. If later input indicates that these interpretations are incorrect, then the STMS contains enough information to allow the input to be re-interpreted.

2 Related Work

2.1 Expectation-Based Conceptual Analyzers

The goal of conceptual analyzers is to map natural language input into a representation of its meaning, which is often expressed in the form of Conceptual Dependency (CD) [Schank, 1975] structures, or other high level knowledge structures, such as MOPs [Schank, 1982]. Examples of these parsers include CA [Riesbeck, 1975] and BORIS [Dyer, 1983]. This mapping is accomplished by associating with each word in the lexicon a conceptualization and a set of expectation rules, or test-action pairs. In BORIS, these rules are called demons. When a word is read, its conceptualization is placed into working memory and its demons are *spawned*. When a demon's test condition is satisfied, the demon *fires* and its action is executed. For example, the figure below shows a simplified lexical entry for "put":

```
(PTRANS
    actor  ?X  (active-voice-demon-find-concept HUMAN before)
    object ?Y  (active-voice-demon-find-concept PHYS-OBJ after)
    to     ?Z  (prep (in on) PHYS-OBJ after))
```

If a word is ambiguous, then disambiguation demons are spawned, which select an appropriate conceptualization for the word. Two types of such demons are used: top-down demons, which select an appropriate meaning for one structure through expectations arising from another structure, and bottom-up demons, which examine context directly to determine which conceptualization should be chosen. For example, "pot" in S1 would be disambiguated by a bottom-up demon such as:

If the context involves COOK then interpret "pot" as cooking-container.

When all the demons have fired, the complete structure is instantiated, yielding the appropriate conceptualization, which is taken to be the result of the parse. For example, the result of parsing S1 would be:

```
(PTRANS
    actor JOHN
    object COOKING-CONTAINER
    to STOVE)
```

Once a complete CD structure is encountered, an attempt is made to recognize it in terms of a higher-level knowledge structure. Here is one heuristic that is used for this purpose:

> If a primitive CD ACT is encountered,
> Then examine the OBJECT of the ACT
> and If the OBJECT has an associated script or MOP
>   Then apply that script or MOP to the ACT

For example, when the OBJECT slot of the above PTRANS is found to be COOKING-CONTAINER, which has the MOP M-COOKING associated with it, M-COOKING is applied and the event is understood in this context.

There are three primary problems with this approach to disambiguation. First, if the disambiguation demons initially select the wrong meaning for an ambiguous word, it is very difficult to back-up and select the correct meaning, since the incorrect concept is saved and not the ambiguous word. Since the word itself is not available, neither are the alternative meanings. For example, since S1 is parsed as shown above, there is no easy way to determine that "pot" was used to express the concept COOKING-CONTAINER. Therefore, the marijuana meaning of "pot" will not be accessible when S2 is presented.

The second problem is that a conceptual structure is only recognized in terms of higher level knowledge structures after all the ambiguities have been resolved. To illustrate why this is problematic, consider the following sentence:

> John bought the pot.

Since the word "bought" builds an ATRANS structure whose OBJECT slot expects a PHYS-OBJ, and since all the possible meanings of "pot" are PHYS-OBJs, nothing is bound to the OBJECT slot. Therefore, the above heuristic cannot be applied, and hence the ATRANS cannot be interpreted in terms of higher-level knowledge structures. However, human readers are clearly capable of generating possible interpretations for this sentence despite the fact that the information required to disambiguate "pot" has not yet been supplied. For example, perhaps John is buying a cooking-container from a store, a plant-container from a nursery, or marijuana from a drug dealer.

Third, it is difficult to incorporate complex syntactic information in a demon-based analyzer. Notice that the demon which binds the ACTOR in the PTRANS is specified as an "active voice" demon. A different demon will be needed to find the actor for passive voice constructs, relative clause, and infinitival constructs, etc.

2.2 Connectionist Parsers

A major parsing paradigm which has recently emerged is connectionist parsing. Examples of these parsers include [Waltz and Pollack, 1985], [Cottrell and Small, 1983], and [McClelland and Kawamoto, 1986]. These models are composed of a highly interconnected network consisting of weighted nodes and links, and a function which computes a new activation value for each node based on its current value and on the activation level of its neighbors. All of the model's knowledge is incorporated in this network. As each word of the input is read, its corresponding nodes are activated, and activation or inhibition is passed on to connected nodes. Over several iterations, a dominant collection of well-connected nodes will emerge, and the concepts associated with these nodes are taken to represent the result of the parse.

To illustrate how the Waltz and Pollack model performs word sense disambiguation, consider the following example:

The astronomer married a star.

The three possible meanings associated with the word "star" are: MOVIE-STAR, CELESTIAL-BODY, and GEOMETRIC-FIGURE. When the word "astronomer" is encountered, activation is spread strongly from "astronomer" to ASTRONOMER, and then to CELESTIAL-BODY. In contrast, MOVIE-STAR will receive a very small amount of activation. Hence, CELESTIAL-BODY will be strongly primed, while MOVIE-STAR will be primed very little, if at all. Encountering the word "star" causes CELESTIAL-BODY to be very highly preferred initially, but the MOVIE-STAR meaning will eventually catch up and dominate because the object of MARRY should be human and animate, while CELESTIAL-BODY is inanimate.

One of the primary advantages of this approach is that the entire parsing process is "automatic" and is controlled by a relatively simple mechanism, rather than by a large group of individual rules. While it is somewhat early to judge the limitations of this model, a number of problems remain to be addressed. Foremost among these problems is assuring that the proper collection of well-connected nodes will indeed dominate. If activation is allowed to spread in an unconstrained manner from the node(s) representing a concept to the node(s) representing *all* of its related concepts, then a large number of inappropriate connections will be generated. For example, in S1, stove and all of the possible meanings of "pot" have the concept physical object as a common isa ancestor. Therefore, there will be an inappropriate, connected path between them. The issue of how to deal with these false positives has also proved problematic for marker passing systems.

One approach to this problem is to have an evaluation mechanism which uses a set of heuristics to eliminate the inappropriate connections [Charniak, 1986], [Granger, 1986]. The problem with this approach is that it requires a separate, complex mechanism for selecting the proper connections from the inappropriate ones. Another proposal [Norman, 1986] is to have an extra, evaluative structure which monitors the behavior of the system and compares expectations with outcomes. However, it is not apparent what the structure of such a mechanism should be, or how it could be implemented in a connectionist model. Another major problem in these models is the lack of sufficient syntactic information.

2.3 Direct Memory Parsing (DMP)

The DMP approach of [Riesbeck and Martin, 1986] is similar in spirit to connectionist and marker passing approaches. In DMP, the objective is to find the most specific structures in memory to which the text is referring, rather than attempting to build new conceptual structures to represent the meaning. The parser, called D-MAP, integrates parsing knowledge into memory by attaching *concept sequences*, which are patterns of words and concepts, to memory structures. The dictionary in D-MAP associates words and concepts with the sequences which contain them. The parser's lexical and syntactic knowledge is encoded in the concept sequences. As an attempt is made to recognize the conceptual elements in concept sequences, more specific structures will be found, causing the representation to be changed from the more general structure to the more specific one. This process is called *concept refinement.*

A marker passing architecture is used in D-MAP. Two types of markers are employed. *Activation markers* are propagated up abstraction hierarchies until they meet *prediction markers*, at which point concept refinement can occur. Prediction markers propagate from concept sequences and indicate what may be expected to become relevant. The result of the parse is an activated set of memory nodes. D-MAP markers contain back pointers so that when the markers meet, their source nodes can be recovered. When a prediction and an activation marker intersect, two rules are used: (1) the prediction marker is passed down the abstraction hierarchy to the most specialized memory structure and (2) if the prediction marker's concept sequence has not yet been completed, then the prediction marker is passed to the next element of the concept sequence. To illustrate how concept refinement operates in this model, suppose that the concept ANIMAL is predicting ANIMAL-FOOT, and that HOOF passes an activation marker up the hierarchy to ANIMAL-FOOT. Using rule (1) above, ANIMAL is refined to HORSE, since it is the most specific node which packages HOOF.

Concept refinement in D-MAP essentially implements, with marker passing, the refinement rule used in MOPTRANS [Lytinen, 1984]. While concept refinement is an important method for finding the most specific nodes in memory, it is not completely adequate for handling ambiguities, for two reasons:

(1) *Multiple Refinement Decisions* - Consider the operation of D-MAP on sentence S1 just before the word "pot" is read. At this point, we would expect the system to be predicting PHYS-OBJ. When "pot" is read, activation markers will be placed on the KITCHEN-CONTAINER, PLANT-CONTAINER and MARIJUANA nodes, and each of these markers will be passed up the hierarchy to PHYS-OBJ, generating three intersections with the prediction marker. The system must now decide which action to refine to, since we would expect there to be nodes for PTRANS-KITCHEN-CONTAINER, PTRANS-PLANT-CONTAINER, and PTRANS-MARIJUANA in memory. However, not enough information has been supplied at this point to make the decision.

(2) *Refinement Commitment* - Once a refinement is made, it cannot be undone. For example, even if D-MAP can refine S1 to, say, PTRANS-COOKING-CONTAINER-TO-STOVE, when S2 follows, D-MAP would not be able to reinterpret this node as LIGHT-MARIJUANA.

3. Approach and Methodology

Three general strategies have been taken with respect to disambiguation:

A. Construct all interpretations and then select one from among them -- this approach suffers from a combinatorial explosion of interpretations.

B. Delay making a commitment -- this strategy is appealing but doesn't address the issue of how long to delay. As long as commitments are not made, the information (potentially useful to other processes) is unavailable. Delaying is problematic in cases where mutual disambiguation must occur. (I.e. the interpretation of ambiguous word W1 helps resolve the interpretation of ambiguous word W2, and vice versa, as in "dope deal".)

C. Commit to an interpretation and then apply recovery rules to back up and correct the error -- this approach is difficult to implement, since the longer the recovery is delayed, the more erroneous inferences will have been made, and the more sophisticated the recovery rules will have to be.

Connectionist approaches pursue strategies A and C, while at the same time avoiding to some extent, their associated problems. First, by propagating an activation strength from all nodes, (syntactic, semantic, etc.) mutual constraints are generated, which keep down the number of combinatorial interpretations. Second, delay is avoided since all relevant hypotheses are given some level of activation. Third, since activations are scalar (have varying strength), the potential for recovery is maintained in the nodes whose current strength of activation is below the threshold of attention.

Finally, connectionist approaches have a potential for neural realism and are able to account for certain psychological phenomena, such as priming. However, these models currently lack operations which are fundamental in higher-level functional NLP systems, specifically: variables, role-bindings, instantiation, and inheritance. Our methodology is to use symbolic operations available at the functional and marker passing levels, while attempting to incorporate connectionist features into our models, (such as strength of activation and levels of units).

4. CAIN: A Conceptual Analyzer Which Maintains Multiple Interpretations

There are three components in our model which function to handle conceptual and structural ambiguities:

4.1 A Short-term Memory Structure (STMS)

The STMS stores connections between roles and words, rather than between roles and conceptualizations* (as in other CAs). This means that an inappropriate interpretation for an ambiguous word can later be corrected, because the word itself is available rather than just the interpretation. Furthermore, it allows the parsing process to begin using higher level information before all of the ambiguities have been resolved. The STMS does not represent the result of the parse. Instead, it is used to direct the search for a matching structure in memory which will represent the result of the parse. Thus, this structure also serves to constrain the spread of activation by specifying what the system is searching for.

4.2 Three Levels of Representation

The system is composed of a three level network, consisting of a lexical level, a phrase level, and a concept level, as shown in figure 1.

The *lexical level* is the input level of the network. When a word from the input text is encountered, the nodes at this level are activated. This is basically the same as in [Cottrell and Small, 1983].

The *phrase level* represents the middle level of the network. Nodes at this level represent phrasal patterns of varying levels of generality [Dyer and Zernik, 1986], [Zernik and Dyer, 1987], [Wilensky, Arens, and Chin 1984]. Units at the lexical level are connected to their corresponding phrases at this level.

The *concept level* is the top level of the network. This level represents concepts and the relations between them. Units at the phrasal level are connected to their corresponding concepts at this level. Knowledge is represented declaratively using a semantic network.

*Storing the actual words for a short time after a sentence has been read is consistent with psychological data [Sachs, 1967], which indicates that people have a good memory for the exact wording of a sentence when tested *immediately* after hearing it.

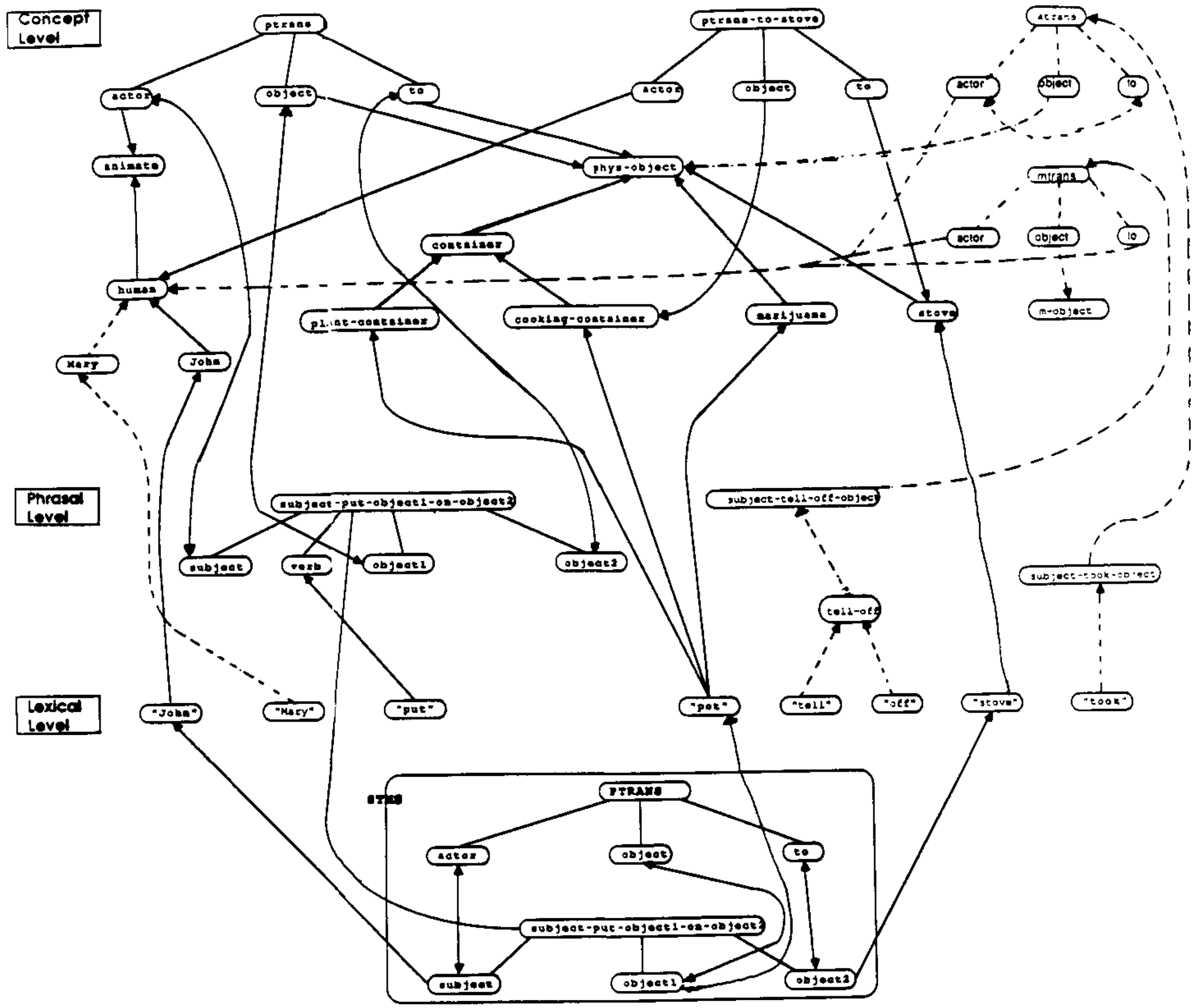

Figure 1

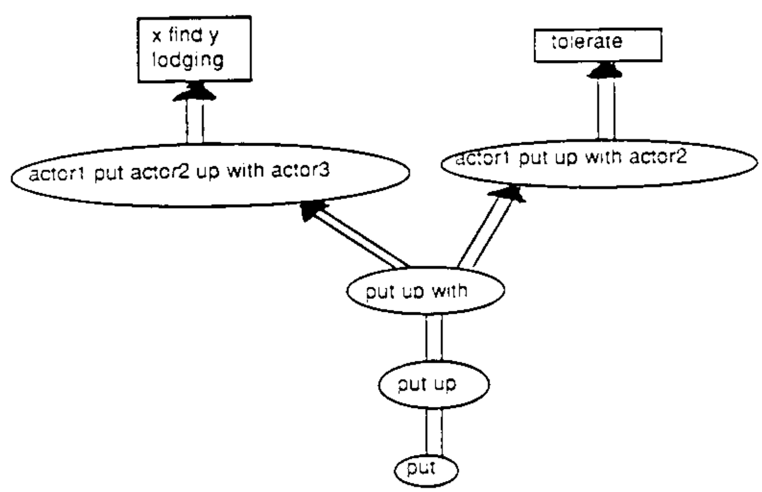

Figure 2

4.3 Context-Directed Marker Passing Scheme

As each word is encountered, an attempt is made to integrate it into the STMS. In parallel with this process, the contexts associated with the word's concept(s) are searched in an attempt to find higher level knowledge structures which can be applied. Both of these processes are performed by a marker passing mechanism. The following three types of markers are used: (1) *activation markers* (AM) are used to find connections between a concept and a role which it could potentially fill, (2) *role markers* (RM) are used to find a connection between a role and its potential fillers, (3) *search markers* (SM) are used to find knowledge structures which potentially match the STMS.

The activation and role markers bear some similarity to the markers used in D-MAP, since they are passed under roughly the same conditions and they contain back pointers to their source nodes. However, these markers differ from those in D-MAP in two important respects.

(1) Rather than being used for concept refinement, they are used to find connections which will stored in the STMS, or to verify whether a connection is found which matches one in the STMS.

(2) Each marker can be passed with *varying degrees of strength*, depending on the strength of the connection between its source and its destination.

5. An Example

To illustrate this approach, we will discuss the conceptual analysis of sentence S1. The goal of parsing this sentence is to find the structure PTRANS-TO-STOVE in the COOK context. We assume that this node is already in memory and that it represents the abstraction of a number of individual episodes in which a cooking-container was placed on a stove in the context of cooking. The following is a description of how the parsing process finds this structure. Note that in figure 1, the bold lines indicate connections which the system actually finds in the process of parsing the sentence, while the dotted lines indicate additional information that is in memory that is inapplicable. Such information is present in the figure to emphasize the manner in which the model finds the correct structures, from a large number of inappropriate ones. We will first focus on the conceptual aspects of the analysis, then we discuss the phrasal issues.

5.1 Marker Passing

When the word "John" is read, an AM is passed from the lexical node for "John" to the node representing the concepts JOHN, HUMAN and ANIMATE, using the rule: *When a node receives an AM, an AM is passed to its parents*. Reading the word "put" causes the phrasal nodes which contain it, such as SUBJ-PUT-OBJ1-ON-OBJ2, to receive an AM. RMs are now passed sequentially to the roles of each phrasal node which received an AM. Each phrasal node has sequence links between its roles, which indicate the order in which they occur (these are not shown in the figure). The node SUBJ-PUT-OBJ1-ON-OBJ2 therefore passes an RM to its SUBJECT role. Note that roles at the phrase and concept level may be linked by an equivalence relationship. For example, the SUBJECT role of SUBJ-PUT-OBJ1-ON-OBJ2 is equivalent to the ACTOR role of PTRANS, as indicated by the double headed arrow in the figure. This organization of memory is based on [Gasser, 1986]. Using the rule, R1: *When a role node receives an RM, an RM is passed to its parents and to its equivalent nodes*, an RM is passed to the ACTOR of PTRANS, and then to ANIMATE. Since an AM was already placed on that node by "John", the following rule applies: R2: *If an RM meets an AM at the concept level, and the source of the RM is a phrase node, then add the phrase, its concept, and the source of the AM to the STMS.* We therefore build a SUBJ-PUT-OBJ1-ON-OBJ2 phrasal structure with the lexical node "John" as its SUBJECT, and a PTRANS structure whose ACTOR is equivalent to the SUBJECT. In anticipation of the next word, an RM is also passed to the OBJECT1 role of SUBJ-PUT-OBJ1-ON-OBJ2, then to the OBJECT role of PTRANS, and then to PHYS-OBJ.

Reading "pot" causes AMs to be placed on each of its possible meanings and then up the hierarchy to PHYS-OBJ. By the rule given above, "pot" is assigned to the OBJECT role of the STMS. In parallel with this process, the contexts associated with each of the possible meanings of "pot" are searched in an attempt to find a structure which matches the STMS. The rule R3: *When a node receives an AM, an SM is passed to all of its decendants that are role nodes* causes an SM to be passed from the COOKING-CONTAINER node to the OBJECT role of PTRANS-TO-STOVE. Note that this rule was applicable when the concepts HUMAN and ANIMATE received AMs, but since these nodes are so general, the *strength* of the SM that is passed is negligible. The strength of the SM in this case will tend to be moderate because the concept COOKING-CONTAINER is involved in other actions, but all of these occur in the context of cooking. The rule R4: *When a role node receives an SM, an SM is passed to its owner* now applies, and an SM is passed from the OBJECT role to PTRANS-TO-STOVE. The contexts associated with the MARIJUANA and PLANT-CONTAINER meanings of "pot" are searched using the same rules.

When "stove" is read, it is integrated into the STMS and the resulting structure is shown in figure 1. Using the rules R3 and R4, an SM is also passed to PTRANS-TO-STOVE. This causes the strength of the SM on this node to be strong enough to merit a comparison to the STMS. The PTRANS-TO-STOVE node now passes RMs to each ot its roles which is also a role of the STMS, namely, ACTOR, OBJECT, and TO. Hence, the STMS provides constraints on the RMs that need to be passed when a node receives an SM. Each of these markers intersects an AM at the nodes HUMAN, COOKING-CONTAINER, and STOVE, respectively. Note that the rule for adding information to the STMS will not be triggered, since PTRANS-TO-STOVE is not a phrase node. Since the source of each AM, namely, "John", "pot" and "stove", is the same as the filler of the equivalent role of the STMS, PTRANS-TO-STOVE is found to match the STMS.

5.2 Role of STM in Reinterpretation

As each sentence is parsed, old markers are removed, while leaving an activation residue. When a new sentence is presented, such as S2, markers are repropagated and since word information (along with focus) is maintained in the STMS, a new configuration of highly activated nodes can be re-generated. Hence "pot" is reinterpreted as MARIJUANA and "stove" in S2 is interpreted as a lighter for the marijuana.

This example illustrates the two key features of this model. First, the STMS allows an initially mistaken interpretation to be corrected. Second, the markers are passed in a very restricted fashion (due to the STMS), and the sentence is parsed without generating a large number of inappropriate connections.

5.3 Phrasal Discussion

In addition to the simple structure of "put" in sentences S1 and S2, consider the following syntactic variations of "put":

1. Mary put her daughter in kindergarten. <placement>
2. Mary put up with John's mother. <tolerate>
3. Mary put John up with Bill for Christmas. <lodging>
4. Mary put up 5 dollars as a bet. <commitment>

These sentences indicate that when "put" is encountered in sentence S1, a number of other nodes receive activation in addition to SUBJ-PUT-OBJ1-ON-OBJ2. For example, the node for the phrase PUT-UP receives activation, since it contains "put" as one of its constituents. This is illustrated in Figure 2. In this simplified figure, each rectangle represents an abbreviation for meanings and roles shown explicitly in Figure 1. The double arrows represent an abbreviation for the links between the roles of the phrasal pattern and the roles of its conceptual subset. Now if the phrase "put up" is encountered, it will send some activation to PUT-UP-WITH, as shown in the figure. Had we been parsing the sentence "Mary put up with John's mother", the existence of the phrasal nodes PUT-UP and PUT-UP-WITH assure that the node SUBJ-PUT-UP-WITH-OBJ does not receive an AM until the entire phrase "put up with" is encountered. The other phrases which "put" activated will pass markers to their roles in a similar fashion. For example, PUT-UP will pass an RM to "up". When the next word is encountered and these markers fail to find a connection, the RM on the role and the AM on its source will be removed.

6. Future Work

A number of interesting issues remain to be addressed. One of these issues is determining how other (external) modules interact with the STMS, and when its information should be replaced. This could be governed by the amount of interference between the STMS of the current sentence and that of the next sentence. Another issue is how learning should occur when no pre-existing structures are present in memory which match the input. A third issue is how to include knowledge structures which connect sentences together at a higher conceptual level, such as goals and plans [Wilensky, 1983]. Also, more phrasal patterns need to be added to incorporate more knowledge of syntax. Further work is also needed in integrating the top-down processing of D-MAP and MOPTANS with the bottom-up approach presented here. At the connectionist level, we are interested in investigating the role of inhibition and relaxation, and how the model could be implemented using a distributed representation. Finally, we need to expand the model to include a question answering capability.

7. Conclusions

In this paper we have presented an approach to direct memory parsing which features: (1) a short-term memory structure which provides for role binding and allows an initially inappropriate interpretation to corrected, (2) a constrained marker passing mechanism which prevents a combinatorially explosive number of connections from being generated, and (3) the incorporation of syntactic information using the phrasal approach by embedding phrases into the network.

References

Charniak, E., A Neat Theory of Marker Passing. *Proceedings of the National Conference on Artificial Intelligence (AAAI-86)*, Philadelphia, 1986.

Cottrell, G. and Small S. A connectionist scheme for modeling word-sense disambiguation. *Cognition and Brain Theory, 1*, 89-120

Dyer, M.G. *In Depth Understanding*. Cambridge, MA: The MIT Press, 1983.

Dyer, M.G. and Zernik, U. Encoding and Acquiring Meanings for Figurative Phrases. *Proceedings of the Annual Meeting of the Association of Computational Linguistics* (ACL-86), New York, 1986.

Gasser, M. *Dynamic Lexical Memory*, Technical Report UCLA-AI-86-6, UCLA, Department of Computer Science, AI Lab., 1986.

Granger, R.H., Eiselt, K.P. and Holbrook, J.K., Parsing with Parallelism: A Spreading Activation Model of Inference Processing During Text Understanding. In J. Kolodner and C. Riesbeck (Eds.), *Experience, Memory, and Reasoning*. Hillsdale, NJ: Lawrence Erlbaum, 1986.

Lytinen, S. *The Organization of Knowledge in a Multi-lingual, Integrated Parser.* Technical Report 340, Department of Computer Science. Yale University, 1984. Ph.D. Dissertation.

McClelland, J.L., Kawamoto, A.H. Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences. In McClelland & Rumelhart (eds.) *Parallel Distributed Processing:.* Vol 2. Cambridge, MA: The MIT Press, 1986.

Norman, D.A. Reflections on Cognition and Parallel Distributed Processing. In McClelland & Rumelhart (eds.) *Parallel Distributed Processing:.* Vol 2. Cambridge, MA: The MIT Press, 1986.

Riesbeck, C.K. Conceptual Analysis. In Schank (Ed.), *Conceptual Information Processing* American Elsevier, NY, 1975.

Riesbeck, C.K. and Martin, C.E. Direct Memory Access Parsing. In J. Kolodner and C. Riesbeck (Eds.), *Experience, Memory, and Reasoning.* Hillsdale, NJ: Lawrence Erlbaum, 1986.

Sachs, J.S. Recognition Memory for syntactic and semantic aspects of connected discourse. *Perception and Psychophysics*, 1967, 2, 437-442.

Schank, R.C. *Conceptual Information Processing*, American Elsevier, NY, 1975. Fundamental Studies in Computer Science, Volume 3.

Schank, R.C. *Dynamic Memory: A theory of Reminding and Learning in Computers and People.* NY: Cambridge University Press, 1982.

Waltz, D. and Pollack, J. Massively Parallel parsing: A strongly interactive model of natural language interpretation, *Cognitive Science*, 9 (1985), 51-74.

Wilensky, Robert, *Planning and Understanding.* Reading, MA: Addison Wesley, 1983.

Wilensky, R., Arens, Y., and Chin, D., Talking to UNIX in English: an Overview of UC, *Communications of the ACM*, 27(6), pp. 574-593, June, 1984.

Zernik, U. and Dyer, M.G., The Self-Extending Phrasal Lexicon. *Computational Linguistics*, in press.