# HEAD-DRIVEN MASSIVELY-PARALLEL CONSTRAINT PROPAGATION:
## Head-features and subcategorization as interacting constraints in associative memory

HIDETO TOMABECHI and LORI LEVIN
Center for Machine Translation
Carnegie Mellon University

## ABSTRACT

We will describe a model of natural language understanding based on Head-driven Massively-parallel Constraint Propagation (HMCP). This model contains a massively parallel memory network in which syntactic head-features are propagated along with other information concerning the nodes that triggered the propagation. The propagated head features eventually collide with subcategorization lists which contain constraints on subcategorized arguments. These mechanisms handle linguistic phenomena such as case, agreement, complement order, and control which are fundamental to linguistic analysis but have not been captured in previous marker-passing models.

## INTRODUCTION

This paper describes a theory of Head-driven Massively-parallel Constraint Propagation (HMCP). The main motivation for our proposal is the inadequacy of traditional marker-passing models for handling syntactic phenomena such as word order, agreement, case marking, control, and unbounded dependencies. The HMCP paradigm diverges from traditional massively-parallel marker-passing schemes (and connectionist models for that matter), in that the model explicitly allows for syntactic and semantic constraints to be propagated in structured-markers.[1]

In the HMCP paradigm, markers that are propagated contain: 1) syntactic head-features which constrain the combination of constituents, 2) the identity of the activation source, and 3) cost information which is relevant to ambiguity resolution and other inferences. In this paper, we concentrate on the necessity of propagating syntactic *head* feature information to properly capture some important linguistic phenomena.

Under the HMCP model, conceptual nodes representing argument-taking predicates carry subcategorization features which specify syntactic properties (such as case) of constituents which can fill their argument positions. Syntactic information such as case, number, and person is propagated up from noun phrases in a package of 'head features' which eventually collides with the constraints in subcategorization frames.

## FAILURE OF MARKER-PASSING MODELS TO CAPTURE SYNTACTIC PHENOMENA

In typical marker-passing (and connectionist) based natural language processing models, syntactic knowledge is handled in either of two ways: 1) a linear ordering of concepts with sequential prediction markers (Riesbeck&Martin[1985], Tomabechi[1987]) which we call a *concept sequence* scheme in this paper[2] 2) nodes configured in a context-free manner (Waltz&Pollack[1985], Bookman[1987], Sumida, *et al*[1988]), which we call a *categorial tree* scheme (i.e., the localist type connectionist scheme). Also, some of these schemes require external modules to handle syntax, using the marker-passing scheme solely for contextual inferences (Charniak[1983/1986], Granger, *et al*[1984], Hendler[1986], Norvig[1987], Tomabechi&Tomita[1988]).

In Riesbeck[1986], the *concept sequence* [*actor* PTRANS-word *dest*] is a template for inputs such as *John flew-to Paris*. This scheme has the advantage of being able to capture the temporal ordering of concepts in an utterance regardless of their levels of abstraction. Thus, the idea of *concept sequence* allows for a model of *phrasal lexicon* (Becker[1975]) which is extended to contain entities from different levels

---

[1] The HMCP algorithm is described in Tomabechi&Tomita[ms].

[2] There are also schemes to use trained networks for sequential activations (e.g., Servan-Schreiber, *et al*[1988]). Discussions of *concept sequence* schemes should apply to such schemes as well.

of abstraction in the same phrasal lexicon[3]. For example, in the above sentence, the abstract concept *actor* which may be linked to specific scriptal knowledge is coexisting with a more surface specific entity, PTRANS-word. On the other hand, this extended notion of *phrasal lexicon* is all the syntax these models have. Precisely because these models abandoned the notion of syntactic category (and other syntactic features), any generalizations that are captured as interactions between different syntactic categories are either lost or redundantly specified by each *concept sequence.* The *cateogrial tree* scheme is in essence similar to the *concept sequence* scheme in terms of the expressivity of the context-free rules, except that the notion of grammatical categories is introduced and semantic features are represented through separate links.

In general, there are three problems with current traditional marker-passing models: (1) they do not adequately represent syntactic information such as syntactic category, case, number, and verb form. (2) They do not have a notion equivalent to the *head of a phrase* (Jackenoff[1977]) which is subcategorized for the syntactic features of its complements. (3) Interactions are strictly local. Because each sequence is independent, there is no way for an element of a concept sequence to see inside another element of the same sequence. For example, the concept sequence [ *PERSON *MTRANS-word that *ACTION] used in recognizing *Sue said that Mary ran* will be activated by any instance of *MTRANS-word and any instance of *ACTION. There is no way to ensure that if the *MTRANS-word is *say*, the set of entities that caused the activation of *ACTION must contain a finite verb. (Unless we create many new concepts such as *fINITE-FORM-RUNNING-ACTION-TAKING-NOMINATIVE-SUBJECT, in which case we will lose generalizations.) The same holds with the *categorial tree* scheme because the contents of embedded nodes are (by the definition of context-free tree) invisible to the external nodes.

The rest of this section lists some syntactic phenomena which have not been addressed in traditional marker passing models.

## SUBCATEGORIZING FOR HEAD FEATURES OF COMPLEMENTS

(1)

a. I believe John studies at CMU.

b.*I believe John study at CMU.

c.*I believe John studying at CMU.

The contrast between (1)a, (1)b, and (1)c, is that *believe* is subcategorized for an embedded clause whose head verb takes finite form but base form or present participle form. Correct treatment of grammaticality in these examples requires the non-local operation of passing up head features from *study* to *believe.* As we have already described, traditional concept sequence schemes do not adequately handle non-local interactions and do not have a method for passing up head features of embedded verbs so that they can be constrained by a higher verb.[4]

## AGREEMENT OF ANAPHORS AND CONTROL

The following sentences from Pollard[ms] involve additional non-local interactions in *control* and *agreement.*

(2)

a. He tried/seemed to wash himself/*herself.

b. He promised her to wash himself/*herself.

c. She persuaded him to wash himself/*herself.

d. She believed him to be washing himself/*herself.

e. She appealed to him to wash himself/*herself.

---

[3] See Hovy[1988] for the use of such a scheme in generating a natural language.

[4] Some recent connectionist research (such as Elman[1988] and Servan-Schreiber, *et al*[1988]) has shown some promising results in training simple recurrent networks to develop expectations to capture grammatical category, and to develop some expectations about concepts and words in embedded sentences. However, they have yet to capture the complexity of grammatical constraints in natural language.

Correct assignment of meaning (and grammaticality judgements) is not possible in traditional marker passing schemes because: (1) There is no way to specify generalizations about behavior of groups of syntactic (categorial) nodes such as, Governing Category in GB, Chomsky[1981]. In traditional marker passing schemes, the pronominals and anaphors in the embedded clauses would probably be bound to any contextually salient entity as long their semantic content agrees (i.e., *MALE-PERSON, etc.). (2) There is no way for the main verb to determine anything about the subject of its complement (for example, that it is controlled by the main clause subject) because the concept sequence containing the main verb cannot see inside the concept sequence corresponding to the controlled clause.

## WORD ORDER CONSTRAINTS BASED ON OBLIQUENESS

In English, the order of a verb's arguments is partly determined by their relative obliqueness; less oblique complements precede more oblique phrasal complements. (Pollard&Sag[1987]) In the examples in (3) the constraint is that "adverb phrase is the most oblique sister of the post-verbal complements, and hence must follow them all" (Pollard&Sag). Concept sequences can specify the well formed orderings (3)a and (3)b by writing surface-specific concept sequences for the possible combinations, but because they do not represent grammatical functions or a hierarchy of obliqueness of arguments, they will miss the generalization about complement order.

(3)

a. He looked up the number quickly.
b. He looked the number up quickly.
c.*He looked the number quickly up.
d.*He looked quickly up the number.
e.*He looked quickly the number up.
f.*He looked up quickly the number.

## WORD ORDER CONSTRAINTS BASED ON THE POSITION OF THE HEAD

Since traditional marker-passing schemes do not have a notion of syntactic head, they cannot capture generalizations about the ordering of complements with respect to the head. (e.g., that the head is always final or that the head is always initial). Japanese allows free ordering of the complements of a verb, but the verb has to come after all of its complements. However, without a notion of head, the only way to specify head-final word order in Japanese is to write surface-specific concept sequences for all possible orders of complements all of which have the head at the end, but this fails to capture generalizations about free complement order in Japanese.

The problems we have identified in this section are inherently problematical in parsing natural language input and are fatal in generating grammatical sentences.

## HEAD-DRIVEN MASSIVELY-PARALLEL CONSTRAINT PROPAGATION (HMCP) PARADIGM

## CONSTRAINT PROPAGATION

The underlying philosophy of our model is that words (or some smaller linguistic unit) in the input string trigger the propagation of structured markers through a network. The markers carry information about the source of the activation, including many syntactic features. Concepts that represent heads of phrases contain bundles of syntactic features which constrain their complements. When activations of complements collide with activations of heads, the syntactic features of the complement and head are unified. Using propagated constraints and features in this way, it is possible for a head to constrain syntactic properties of its complements such as syntactic category, case, agreement features and whether they can be expletive. It is also possible to specify principles of word order based on obliqueness and the head initial/final distinction. Because most of the features that are propagated and constrained are labeled as *head-features* in linguistic theory (such as GPSG (Gazdar, *et al*[1985]), and HPSG), our model is named Head-driven Massively-parallel Constraint Propagation (HMCP) model.

## THE NOTION OF HEAD

The *lexical head* of a phrase is a word which determines many of the syntactic properties of the phrase as a whole (Jackendoff[1977], Pollard&Sag[1987]). Thus the lexical head of a verb phrase or sentence is a verb, the lexical head of a prepositional phrase is a preposition, and so on. The features of the head, determine what syntactic environments the phrase can occur in. For example, a clause headed by a finite verb can occur as a complement of the verb *believe*, but verb phrases headed by present participles cannot occur in this environment. In the HMCP model, we are currently adopting head-features similar to those postulated in the HPSG framework. These include major category, case features, verb forms (tensed, finite, base, and participle forms), noun forms (expletive and normal), and many others. We also include agreement features which are not treated as syntactic head features in some syntactic theories.

These head-features are propagated upward from the lexical head (i.e., the node that is a head and that was activated by the input) and are carried as constraints on future marker collisions and further spreading activation.

## THE NOTION OF SUBCATEGORIZATION

Lexical items are organized into subcategories depending on the number and kind of other nodes that they combine with in oder to recognize (or generate) a sentence. Subcategorization is different from the notion of *concept sequence* in that it is independent of the surface order of constituents. In our system, as in HPSG, the subcategorization list reflects the obliqueness order of the grammatical functions that are subcategorized for. It is also a list of constraints that need to be satisfied by nodes that fill argument positions in order to make recognition (and generation) complete. These constraints are most likely to be syntactic head-features that are propagated by lexical activations; however, there may also be semantic constraints on the fillers of argument positions. *Concept sequences*, on the other hand, represent only linear order of concepts and fail to capture syntactic constraints (i.e., head-features) that need to be satisfied in order to complete the subcategorization. Thus, the notion of subcategorization and *concept sequence* should not be confused.

# LAYERED NETWORK

## FOUR LAYERS OF NETWORK

Under the HMCP model, the network has different layers (not to be confused with the hidden layers in neural-net frameworks) that are independent of the semantic-net based abstraction hierarchy. Currently the layers are 1) the Static Layer (SL) 2) the Potential-activation Layer (PL), 3) the Active-Layer (AL) and 4) the Decaying-Layer (DL). The SL and AL are perhaps analogous to short-term memory (STM) and long-term memory (LTM) in the traditional psychology literature. The layers represent groups of nodes which are differentiated by the level (and time) of activation.

The SL is the layer which nodes belong to by default before the first utterance in the discourse. It is an associative network of memory with nodes corresponding to memory structures that represent entities at different levels of abstraction from phonemic nodes to discourse level nodes.

The PL contains nodes that are potential candidates for filling slots in subcategorization lists.

The AL contains nodes that are activated by words in the current sentence using a standard upward activation scheme (e.g., DMA) and that meet the subcategorization constraint check. At the end of the sentence, the AL will be the nodes that correspond to the elements of all the accepted subcategorization lists. Syntactic constraints such as complement order constraints and parameter-based discourse constraints (such as CENTER and PIVOT constraints (Tomabechi[ms])) apply at this layer. Discourse functions such as *Forward-looking Center (Cf)* (Grosz, *et al*[1986]) and *potential foci* (Sidner[1983]) are also defined at AL.

The DL is simply the AL of the preceding utterance. The levels of activation of nodes in DL decay with time and the nodes will eventually return to SL. The least oblique element of the immediately preceding utterance in this layer corresponds to *Backward-looking Center (Cb)* (Grosz, *et al*) and *discourse focus* (Sidner[1983]).

## HMCP CONSTRAINT PROPAGATION

The following three things are propagated from lexically activated nodes: 1) head-features attached to the node, 2) identity of the instance node that is associated with the current lexical activation (i.e., which specific instance should be associated or created with the current lexical activation) and 3) the specific cost (weight) associated with the given lexical activation. The last two are discussed in detail in Tomabechi, *et al*[1989] and Kitano, *et al*[1989] and therefore, will not be discussed in this paper.

## EXAMINATION OF THE MODEL WITH A CONTROL CONSTRUCTION

We will describe the HMCP parsing model by walking through the parse of *John tried to give Mary the book.* The equi verb *try* specifies that the entity associated with its subject be shared with that of the unexpressed subject of its VP complement. In other words, *try* specifies that it subcategorizes for a complement which is itself unsaturated and there is a dependency between the embedding subject and the embedded subject. This phenomenon is known as *control.*

Before parsing the sentence, all nodes that potentially satisfy an element of a subcategorization list are put into the Potential-activation Layer (PL). In this example, nodes corresponding to *try* and *give* contain subcategorization lists as the value of the subcat feature. In the node corresponding to *try*, NP[NOM] in the subcategorization list is coindexed with *PERSON in the *trier* role, so *PERSON is added to the PL. *ACTION, *OBJECT, and all other concepts coindexed with subcategorized positions are concurrently added to the PL (massive parallelism). All other nodes in the network are in the Static Layer (SL).

We will be using a network of semantic memory similar to the ones described in the DMA and associative memory literature following the tradition of semantic networks since Quillian[1968,1969] using structured memory nodes (such as Mops, Schank[1982]). For example, the lexical concepts representing the verbs *give* and *tried* are encoded in the network as below:

```
(lex-node *GIVE
 (is-a (*ACTION))
 (phonology </g/ /i/ /v/>)
 (syn-head-feature ((MAJ V) (VFORM BSE) (AUX MINUS)))
 (giver (*PERSON 1))
 (receiver (*PERSON 2))
 (given (*OBJECT 3))
 (subcat <(NP[NOM] 1), (NP[ACC] 2), (NP[ACC] 3)>))

(lex-node *TRY
 (is-a (*ACTION))
 (phonology </t/ /r/ /a/ /i/ /d/>)
 (syn-head-feature ((MAJ V) (VFORM FIN) (AUX MINUS)))
 (trier (*PERSON 1))
 (circumstance (*ACTION 2))
 (subcat <(NP[NOM] 1), (((MAJ V)
                           (VFORM INF))
                          subcat<(NP 1)>)
                        2)>))
```

The list (NP[NOM] 1) in the subcat feature is a short-hand for (((MAJ N) (CASE NOM)) 1).

The first word *John* activates the node *JOHN and the activation is propagated upward in the abstraction hierarchy along with the head features. When the activation reaches a node in the PL, in this case *PERSON, the head features carried in the activation are checked against the constraints on the position that it fills in a subcategorization list. In this example, the activation triggered by *John* carries the head feature NP[NOM], which is checked against and satisfies the NP[NOM] constraint on the *trier* role. When the constraints on a node in the PL are met, it moves to the AL. In this case, *PERSON moves to the AL. If the constraints are not met, the node moves back to SL.

The constraints on *PERSON are checked concurrently for every other verb and every other role that can be filled with *PERSON. In each parallelly spawned[5] (forked) environment for each concurrently recognized subcategorization, *PERSON is either moved to AL or SL. Thus the processing is massively parallel in nature.

The next word, *tried*, activates the node *TRY, which is subcategorized for NP[NOM] coindexed with *PERSON. In the environment (for the evaluation that was spawned) where *PERSON was trying to get into the *trier* role of *TRY, NP[NOM] is removed from the subcategorization list and the parse continues looking for the other subcategorized argument of *try*. In parallel environments where *PERSON was trying to fill roles for other verbs, nothing happens.

Recognition of *to give Mary the book* continues in a similar manner. *Mary* fills the *receiver* role and *the book* fills the *given* role and (NP[ACC] 2) and (NP[ACC] 3) are removed from the subcategorization list. When items are removed from a subcategorization list, the new subcategorization list and the head features are propagated upward. In this case, *GIVE propagates a subcategorization list of one element, NP[NOM], and the head features of *give*, ((MAJ V) (VFORM bse) (AUX minus)). The concept *ACTION in the PL receives this activation, which satisfies the constraints on the *circumstance* role of *TRY. *TRY specifies that the NP[NOM] which fills the *trier* role is coindexed with the NP[NOM] inside the VP which fills the *circumstance* role. This now indicates that *John* fills the *giver* role in *GIVE. This way, the phenomenon known as control is handled in the HMCP model.

## CONCLUSION

It has been accepted in the linguistic and psychological communities that syntactic constraints play an important role in many types of linguistic phenomena. Yet it is our claim that in the current marker-passing and connectionist based natural language schemes, very little has been accounted for in terms of syntactic constraints and the interactions of syntax, semantics and pragmatics. Methods that have been employed for capturing syntactic phenomena have been mostly ad hoc. For example, in the traditional marker passing schemes, the notion of *concept sequence* has been accepted as a central method of capturing English word-order. However, it has been observed that English word order is best described in terms of the obliqueness order of grammatical functions. In *categorial tree* schemes, the nodes were simply organized in a context-free manner and constraints based upon the internal features of the embedded nodes, which are vital in handling phenomena such as control, have not been captured (similarly with the *concept sequence* schemes).

The HMCP model attempts to model interactions among various syntactic features as well as between syntax, semantics and pragmatics in a principled manner. We have seen that HMCP handles case, agreement, and control based upon subcategorization and head-feature propagation. The method of handling subcategorization in HMCP (not presented here for reasons of space) allows for capturing generalizations based on categories and syntactic-features of complements. Therefore, our analysis does not suffer from the adhocness associated with the traditional marker-passing schemes. The layered network based on activation status allows for complement order constraints based on obliqueness, head-initial/head-final distinctions, and discourse-parameter based constraints to be applied at the AL only and thus, the number of parallel constraint applications are controlled to be minimum.

---

[5] Our algorithm requires a pure parallelism (such as supported by Multilisp (Halstead, *et al*[1986])) in that in order to evaluate two things in parallel (i.e., consideration of two concurrently recognized subcategorizations by two different verbs requiring NP[NOM] filling *person), a new task is spawned to evaluate one of them. The environment of the spawned task must be exactly the same environment as was in effect when the spawn occurred. This means we have massively-parallel worlds (environments) representing each subcategorization check.

The HMCP model is massively-parallel in nature and evaluations are spawned (provided with independent environments) for each subcategorization that is active. HMCP is based upon a massively-parallel structure-passing (MSP) algorithm which presupposes a neural-network that is capable of passing around some amount of information. (One such neural-net architecture, Frequency Modulation Neural Network (FMNN) and its phenomenological plausibility are described in Tomabechi&Kitano[1989]). Currently the MSP algorithm is supported on MULTILISP which is a true parallel Lisp developed at MIT (Halstead, *et al*[1986]) which runs on MACH (Rashid, *et al*[1987]) at CMU.

## ACKNOWLEDGEMENTS

The authors would like to thank Jaime Carbonell, Masaru Tomita, Hiroaki Kitano, Hitoshi Iida and Carl Pollard for their assistance in various aspects of work reported in this paper. Thanks are also due to the members of the Center for Machine Translation for fruitful discussions.

### References

[1] Becker, J.D. (1975) 'The phrasal lexicon'. In *Theoretical Issues in Natural Language Processing*.

[2] Bookman, L.A. (1987) 'A Microfeature Based Scheme for Modelling Semantics'. In *Proceedings of the IJCAI-87*.

[3] Charniak, E. (1983) 'Passing Markers: A theory of Contextual Influence in Language Comprehension'. *Cognitive Science* 7.

[4] Charniak, E. (1986) 'A neat theory of marker passing'. In *Proceedings of the AAAI-86*.

[5] Chomsky, N. (1981) *Lectures on Government and Binding*, Foris.

[6] Elman, J. (1988) Finding structure in time. CRL TR-8801. Center for Research in Language, University of California, San Diego.

[7] Gazdar, G., Pullum, G., and Sag, I. (1985) *Generalized Phrase Structure Grammar*. Harvard University Press.

[8] Granger, R. and Eiselt, K. (1984) 'The parallel organization of lexical, syntactic, and pragmatic inference processes' In *Proceedings of the First Annual Workshop on Theoretical Issues in Conceptual Information Processing*.

[9] Grosz, B., Joshi, K, and Weinstein, S. (1986) Towards a computational theory of discourse interpretation. Preliminary draft.

[10] Halstead, R., Loaiza, J., and Ma, M. (1986) *The Multilisp Manual*. Massachusetts Institute of Technology.

[11] Hendler, J. (1986) *Integrating Marker-Passing and Problem Solving: A Spreading Activation Approach to Improved Choice in Planning*. Department of Computer Science, University of Maryland.

[12] Hovy, E. (1988) *Generating Natural Language Under Pragmatic Constraints*. Lawrence Erlbaum Associates.

[13] Jackendoff, R. (1977) *X-bar Syntax: A study of Phrase Structure*. MIT Press.

[14] Kitano, H., Tomabechi, H and Levin. L., (1989) 'Ambiguity Resolution in ΦDMTRANS'. In *Proceedings of the fourth Conference of the European Chapter of the Association for Computational Linguistics*.

[15] Norvig, P. (1987) 'Inference in Text Understanding'. In *Proceedings of the AAAI-87*.

[16] Pollard, C. and Sag, I. (1987) *An Information-based Syntax and Semantics*, Volume I. CSLI.

[17] Quillian, M.R. (1968) 'Semantic Memory'. In *Semantic Information Processing*, ed. Minsky, M. MIT Press.

[18] Quillian, M.R. (1969) *The teachable language comprehender*. BBN Scientific Report 10.

[19] Rashid, R., A. Tevanian, M. Younge, D. Youge, R. Baron, D. Black, W. Bolosky and J. Chew (1987) 'Machine-Independent Virtual Memory Management for Paged Uniprocessor and Multiprocessor Architectures'. CMU-CS-87-140. Carnegie Mellon University.

[20] Riesbeck, C. (1986) 'From Conceptual Analyzer to Direct Memory Access Parsing: An Overview'. In *Advances in Cognitive Science 1*, ed. Sharkey, N. Ellis Horwood Ltd.

[21] Riesbeck, C. and Martin, C. (1985) *Direct Memory Access Parsing*. Yale University Report 354.

[22] Schank, R. (1982) *Dynamic Memory: A theory of learning in computers and people*. Cambridge University Press.

[23] Servan-Schreiber, D., Cleeremans, A., and McClelland, J. (1988) 'Encoding sequential structure in simple recurrent networks'. CMU-CS-88-183, Carnegie Mellon University.

[24] Sidner, C. (1983) Focusing in the comprehension of definite anaphora. In M. Brady and R. Berwick, ed., *Computational Models of Discourse*, MIT Press.

[25] Sumida, R., Dyer, M. and Flowers, M., (1988) 'Integrating Marker passing and Connectionism for Handling Conceptual and Structural Ambiguities'. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society*.

[26] Tomabechi, H. (1987) 'Direct Memory Access Translation'. In *Proceedings of the IJCAI-87*.

[27] Tomabechi, H. and Tomita, M. (1988) 'The Integration of Unification-based Syntax/Semantics and Memory-based Pragmatics for Real-Time Understanding of Noisy Continuous Speech Input'. In *Proceedings of the AAAI-88*.

[28] Tomabechi, H. and Tomita, M. Manuscript. 'Massively Parallel Constraint Propagation: Parsing with Unification-based Grammar'. Center for Machine Translation, Carnegie Mellon University.

[29] Tomabechi, H., and Kitano, H. (1989) 'Beyond PDP: the Frequency Modulation Neural Network Architecture'. In *Proceedings of the IJCAI-89*.

[30] Tomabechi, H., Kitano, H., Mitamura, T., Levin, L., Tomita, M. (1989) *Direct Memory Access Speech-to-Speech Translation: A Theory of Simultaneous Interpretation*. CMU-CMT-89-111, Carnegie Mellon University.

[31] Waltz, D. L. and Pollack, J. B., (1985) 'Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation.' *Cognitive Science* 9(1).