PARSING = PARSIMONIOUS COVERING?
(Abduction in Logical Form Generation) *

Venu Dasigi
Department of Computer Science and Engineering
Wright State University Research Center
3171 Research Boulevard
Dayton, OHIO 45420
vdasigi@cs.wright.edu

Abstract

Many researchers believe that certain aspects of natural language processing, such as word sense disambiguation and plan recognition in stories, constitute abductive inferences. We have been working with a specific model of abduction, called *parsimonious covering,* applied in diagnostic problem solving, word sense disambiguation and logical form generation in some restricted settings. Diagnostic parsimonious covering has been extended into a dual-route model to account for syntactic and semantic aspects of natural language.

The two routes of covering are integrated by defining "open class" linguistic concepts, aiding each other. The diagnostic model has dealt with sets, while the extended version, where syntactic considerations dictate word order, deals with sequences of linguistic concepts. Here we briefly describe the original model and the extended version, and briefly characterize the notions of covering and different criteria of parsimony. Finally we examine the question of whether parsimonious covering can serve as a general framework for parsing.

1 Introduction

Natural languages are rife with ambiguity. There are lexical ambiguities; words in isolation may be seen to have multiple syntactic and semantic senses. There are syntactic ambiguities; the same sequence of words may be viewed as constituting different structures. And finally, there are semantic and pragmatic ambiguities, all of which may be resolved in context. Ambiguity and its context-sensitive disambiguation, it turns out, are two important characteristics of abductive inferences.

There have been various attempts at characterizing abductive inference and its explanatory nature [Appelt, 90; Charniak and McDermott, 85; Hobbs, *et* al., 88; Josephson, 90; Konolige, 90; Pople, 73; Reggia, 85, etc.]. While they differ somewhat in details, they all boil down to accounting for some observed features using potential explanations consistently in a "parsimonious" (often "minimal") way. Over the past decade, a formal model for abduction based on these ideas was developed at Maryland; this theory is called *parsimonious covering.* The theory originated in the context of simple diagnostic problems, but extended later for complex knowledge structures involving chaining of causal associations.

A diagnostic problem specified in terms of a set of observed manifestations is solved in parsimonious covering by satisfying the coverage goal and the goal of parsimony. Satisfying the coverage goal requires accounting for each of the observed manifestations through the known causal associations. Ambiguity arises here, because the same manifestation may be caused by any one of several candidate disorders. Ensuring that a cover contains a "parsimonious" set of disorders satisfies the goal of parsimony. There could potentially be a large number of covers for the observed manifestations, but the "parsimonious" ones from among them are expected to lead to more plausible diagnoses. The plausible account for a manifestation may be one disorder in one context and another disorder in a different context. Such contextual effects are to be handled automatically by the specific criterion of parsimony that is chosen.

For medical diagnosis, reasonable criteria of parsimony are minimal cardinality, irredundancy and relevancy [Peng, 85]. Minimal cardinality says that the diagnosis should contain the smallest possible number of disorders that can cover the observed symptoms. A cover is considered irredundant (not redundant) if none of its proper subsets is also a cover, i.e., if the cover contains no disorder by removing which it can still cover the observed symptoms. Relevancy simply says that each disorder in the cover should be capable of causing at least one of the observed manifestations.

Consider an abstract example where disorder d_1 can cause any of the manifestations m_1 and m_2; d_2 can cause any of m_1, m_2 and m_3; d_3 can cause m_3; d_4 can cause m_3 and m_4; and finally, d_5 can cause m_4. If the manifestations $\{m_1, m_2, m_3\}$ were observed, the disorder set $\{d_2\}$ constitutes a minimal cardinality cover; the irredundant covers that are not minimal cardinality covers are {d\, d_3} and $\{d_1, d_4\}$; and an example of a redundant, but relevant cover would be $\{d_1, d_3, d_4\}$. While $\{d_2, d_5\}$ is a cover that has an irrelevant disorder (d_5) in it, {d_3,

*This research was supported in part by the State of Ohio Research Challenge grant to the author.

Parsimonious Covering Theory (Diagnosis)	Natural Language Processing
SIMILARITIES:	
symptoms	words
disorders	internal assertions
intermediate syndromes	word senses and structures
symptoms with multiple causes	ambiguous words
pathognomonic manifestations	unambiguous words
observed manifestations (to be explained)	a sequence of words in input text (to be interpreted)
causal relation (between symptoms and disorders)	lexical and semantic associations (between words and senses)
diagnostic explanation (i.e., a set of disorders)	semantic interpretation (i.e., a set of related assertions)
DIFFERENCES:	
order of entities ignored	word order important
sets of entities	sequences of concepts
single type of knowledge (causal)	two types of knowledge (syntactic/semantic)

Table 1: *Similarities and Differences between Diagnostic Problem Solving and Natural Language Processing*

$d_4\}$ is a non-cover, since together the disorders in this set cannot account for all observed manifestations.

Several natural language researchers have been actively involved in modeling abductive inferences that occur at higher levels in natural language, e.g., at the pragmatics level. Abductive unifications that are required in performing motivation analysis, for instance, might call for making the least number of assumptions that might potentially prove false [Charniak, *88].* Litman uses a similar notion of unification, called consistency unification [Litman, 85]. Hobbs and his associates propose a method that involves minimizing the cost of abductive inference where the cost might involve several different components [Hobbs, *et ai,* 88]. Although [Charniak and McDermott, 85] indicate that word sense disambiguation might be viewed to be abductive, nobody has pursued this line of research. It is very clear that there exists a strong analogy between diagnostic parsimonious covering and concepts in natural language processing. There are, however, important differences as well. These similarities and differences are summarized in Table 1. We have tried to extend parsimonious covering to address some of the idiosyncrasies of language (contrasted to diagnosis) and apply it to low level natural language processing.

2 Covering and Parsimony in Language

Linguistic concepts are viewed in parsimonious covering to be much like disorders and manifestations in diagnostic problems. However, in order to account for word order and structural constraints in language on the one hand and to account for the lexical and semantic content on the other, two aspects are attributed to each linguistic concept. These two aspects are loosely referred to as syntactic and semantic aspects, respectively. Concepts are covered parsimoniously in these two aspects, and the processes of covering are called syntactic and semantic covering.

The notions of coverage and parsimony are briefly sketched here for syntactic covering through an abstract example here. Unlike in the case of diagnostic covering, the covers in syntactic covering are sequences rather than sets. Consider the following descriptions of categories c_1 through C_5 in terms of simpler categories (or words) w_0 through w_10 below (sequences are indicated by being enclosed between "<>"):

c_1: $<w_0\ w_1\ w_2\ w_4\ w_5\ w_3\ w_6>$
c_1: $<w_4\ w_1\ w_7\ w_0>$
c_2: $<w_7\ w_1\ w_8>$
c_3: $<w_9\ w_10>$
c_4: $<w_2\ w_6\ w_3>$
c_5: $<w_0\ w_7>$

The categories shown in bold face are mandatory categories, i.e., categories that must be present for the description to viably apply to a context. Semantic considerations govern whether a category is mandatory in a description. Depending on the domain, "the patient blind" might still make sense (indicating that the omitted copula is not mandatory), but "the patient" alone does not make complete sense (indicating that for this type of sentences, an adjectival complement is mandatory). See [Dasigi, 88] for discussion.

Suppose the input sequence is $<w_1, w_2, w_3>$. Some valid covers (covering sequences) are $<c_1>$, $<c_1, c_3>$, $<c_3, c_1>$, $<c_2, c_4>$, $<c_2, c_3, c_4>$, etc. Some non-covers are $<c_2>$, $<c_4>$, $<c_2, c_3>$, $<c_4, c_2>$, etc., either because they cannot account for all the categories in the input sequence or because they cannot account for the correct order. Note that although $<c_2, c_4>$ is a cover, $<c_4, c_2>$ is not a cover. For instance, it makes sense to cover "paint the wall" with the sequence <Verb Noun-Phrase>, but not by <Noun-Phrase Verb>. Irredundant covers include $<c_1>$ and $<c_2, c_4>$. Of these two irredundant covers, the former is also minimal (i.e., of minimal cardinality) and the latter is not. Insertion of c_5 into any valid cover causes it to be a non-viable cover since the category mandatory to c_5, namely, w_7 is not present in the input sequence to be covered. Thus, $<c_1, c_5>$ is a non-viable cover.

Consider the cover $<c_1, c_4>$. Superficially, it appears to be a redundant cover since c_1 by itself is a cover. When the second rather than the first description of c_1 is taken into account, however, there is no redundancy in the cover, in a certain sense. For more concreteness, consider the following two classic sentences that differ in a single word:

"John painted the wall with a crack."
"John painted the wall with a brush."

Now, suppose there exist the usual descriptions for noun phrases (Noun-Phrase) and prepositional phrases (Prep-Phrase). Although in both sentences, the highlighted words can be syntactically covered by the irredundant cover <Noun-Phrase>, the sequence <Noun-Phrase Prep-Phrase> is a more appropriate cover in the second sentence, and we would like to consider that cover as irredundant, too. This characterization of irredundancy is obviously important, and is somewhat similar to the notion of "relevant diagnostic covers" defined in the previous section.

For the sake of completeness, we briefly describe the salient features of semantic covering. A detailed account and algorithms may be found in [Dasigi, 88]. The conceptual objects manipulated by semantic covering are domain-specific semantic senses. For semantic covering, the *order* of the concepts being covered is *no longer* important. Semantic covering involves discovering the relationships underlying the domain-specific entities evoked by input words, so that a parsimonious semantic cover can be synthesized for them; this cover corresponds to the logical form of the original sequence of words. There are two types of semantic covering. The first type *of* covering involves covering individual content words by domain-specific senses corresponding to objects, attributes, etc. This type of covering involves only lexical associations. Here, a domain-specific entity semantically covers a content word if any of the content words in the name or synonyms of the entity is morphologically related to the word itself or a domain-specific or domain-independent synonym of the word.

The other type of semantic covering is based on the relationships in a domain-specific semantic network. A simple domain-specific entity may be represented by a single node in the semantic network, e.g., an attribute. Also, a non-atomic subgraph of the semantic network can represent a more complex domain-specific entity, e.g., an assertion that relates an attribute and a possible value for it. Either kind of domain-specific entity - whether represented by a single node or by a subgraph in the domain-specific semantic network - is said to be covered by any of its supergraphs. Since any super-graph of a domain-specific concept can cover it, for any domain-specific concept there are potentially a huge number of covers, some of which are very redundant. There should be some means of controlling the number and sizes of potential covers. Criteria of parsimony and other constraints are used to achieve this control.

A criterion of parsimony called *cohesiveness* is chosen, inspired by the fact that in order to be understandable, text must be cohesively connected. A set of semantic categories are designated as assertionals (loosely corresponding to the notion of a sentence or an independent clause in English). A semantic cover corresponding to a non-assertional category is considered to be cohesive if it is the smallest (in terms of nodes) connected graph covering the concepts in question. A semantic cover corresponding to an assertional category is considered to be cohesive if either it is the smallest connected graph covering the concepts being covered or it is a not necessarily connected graph of several such domain-specific entities belonging to assertional categories. If there is more than one unconnected cover for the same concepts, the smallest connected cover of such unconnected components is the cohesive cover. It can be seen that cohesiveness refers to the "size" of the covers, and it is similar to "minimal cardinality," used in early versions of parsimonious covering theory for diagnostic problems. Indeed, if minimality were to be extended to structured entities, it would be similar to cohesiveness above. Cohesiveness refers to how well a cover fits into its surrounding context, a generalization of the notion of minimal cardinality, applied to structured entities.

Consider two consecutive concepts that have the same domain-specific entity (say an object) as one of the many candidate covers. Since both concepts can be covered by the same entity, the entity is a minimal cover for both of them together. This example of parsimonious covering is essentially the same as minimal covering in the unextended parsimonious covering theory for diagnostic problem solving. However, suppose the two concepts involved *cannot* be covered by the same domain-specific entity. A minimal cover in the unextended parsimonious covering theory would consist of any pair of entities (pair - because there are two words to be covered) such that each entity in the pair covers one concept. But when structured entities with semantic associations among them arc considered, the entities in the pair must also *unify,* taking domain-specific associations into account.[1] Unification of such structures corresponds to a search in the domain-specific semantic network, say, by marker passing [Charniak, 83].

One important remark about semantic covering is in order. Cohesiveness, as a notion of parsimony for semantic covering, is intended to capture how plausible a semantic cover is. But it is possible that a cohesive cover might turn out to be implausible when checked for well-formedness. Because of this possibility, there should be means to recompute the next most plausible (cohesive) cover. Thus, whenever a cohesive cover is found, all the irredundant covers must be saved so that the space of possibilities they constitute can be explored for cohesiveness if the cohesive cover that was found were to be rejected later. Consider the following abstract example. Let x_1 ,x_2, x_3, x_4 and x_5 be the senses of one ambiguous linguistic concept and y_1, y_2, ,y_3 and y_4 be the senses of another concept. If these two concepts were syntactically covered together by an open class semantic category, then semantic covering will be initiated. Now, what needs to be semantically covered is the conjunction of the following two disjunctions (representing 5*4 — 20

[1]This can be understood as follows: An assertion may be viewed as a predicate assert(?v,?a,?o), where ?v, ?a and ?o are variables such that ?v is a possible value of attribute ?a, which in turn is an attribute of object ?o. If one of the constituents is covered by a specific value vl and the other is covered by a specific attribute a2, the covers effectively specify the assertions assert (vl,?aa,?oo) and assert(?vv,a2,?ooo), respectively. Now unification may be performed in the usual sense.

combinations):

$$\{x_1\ x_2\ x_3\ x_4\ x_5\} \text{ and } \{y_1\ y_2\ y_3\ y_4\}$$

Suppose a cohesive cover is found between x_2 and t/3. Then the irredundant cover will be constituted by the following three conjunctions of disjunctions (which represent the remaining 19 combinations):

$$\{x_1\ x_3\ x_4\ x_5\} \text{ and } \{y_1\ y_2\ y_4\}$$
$$\{x_2\} \text{ and } \{y_1\ y_2\ y_4\}$$
$$\{x_1\ x_3\ x_4\ x_5\} \text{ and } \{y_3\}$$

If the cohesive cover that was discovered gets rejected, the next most cohesive cover might be computed from these irredundant covers.

Semantic covering interacts closely with syntactic covering. Irredundant syntactic covering has a very nice property, namely, when complete sets of irredundant syntactic covers are considered, they are transitive across any number of layers when more than two layers of covering (e.g., as in typical parse trees) are involved [Peng and Reggia, 87; Dasigi, 88]. However, for a sequence of items, the number of irredundant covers at the next layer grows exponentially [Dasigi, 88]. Heuristics are needed for focusing search in such an ocean of covers, and semantic considerations serve this role. In the space of irredundant syntactic covers, search would be focused on "plausible" semantic covers. Thus, the two routes of covering aid each other by syntactic covering providing a search space for semantic covering, and the latter focusing further syntactic covering at the next layer. Integration of the two routes of covering is facilitated by attributing *both* syntactic and semantic categories to distinguished linguistic concepts, called *open class* concepts.[2] In general, if the category that has just been postulated as a cover happens to be an open class category, it initiates semantic covering, thus integrating both the routes of covering.

3 Some Examples

A significant prototype was implemented to apply this algorithm in the context of an interface to an expert system. Instead of syntactic categories such as nouns, verbs, noun-phrases, etc., semantic categories were used in the syntactic covering process. Semantic covering was performed using domain-specific concepts defined in a knowledge base used by the expert system. In an OPS5-style expert system language, domain-specific concepts such as, patient, vision, blind, etc. were classified into semantic categories such as objects (obj), attributes (attr), values (val), etc. Two application domains were considered; the first domain is characterized by a sizable, prototype neurological knowledge base and the other deals with a toy chemical spills knowledge base. Some examples that were successfully handled by the prototype interfaces are:

"Visual acuity is blind."
"Visual acuity is blind on the left."
"Babinski on the left. Right unremarkable."
"The water is brown, radioactive and oily. Its pH is basic ..."

These examples demonstrate the use of lexical information, limited ability to handle ungrammatical sentences, interpretation of sentences in a discourse context rather than in isolation, etc. Note that the first few words of the first two inputs are the same. Their interpretations are, however, significantly distinct in the context of the knowledge base that was used, illustrating a form of non-monotonic inference in text interpretation. All but the last input is from the neurology domain and the last one is from the other.

A very simple example of parsimonious covering is given below to convey the flavor of the approach; more detailed examples of dual-route parsimonious covering may be found elsewhere [Dasigi, 88; Dasigi, 90]. Details are omitted due to space considerations, and we appeal to the reader's intuition in making sense out of this brief example. Suffice it to say that the category assert (and its variations) corresponds to sentences or clauses; obj and attr (and their variations) correspond to noun phrases; and val (and its variations) correspond to noun phrases or adjective complements. The category asg-verb stands for "assignment verb" (e.g., "is"). There are different ways an assert may be described in terms of the other categories mentioned so far. Often, val is a mandatory category in describing an assert (that is, it is unlikely that an assert makes semantic sense if a val is not present). Now, suppose a sentence begins with

"Vision is

and is to be covered syntactically. One sequence of terminal categories that cover the first two words in this sentence is <attr, asg-verb> among others, since vision is an attribute and the word "is" is an instance of asg-verb. Since, this is an embedded sequence of what is expected of the above description of assert, the category assert is postulated to be a *non-viable syntactic cover* for the first two words. It is a *cover* because the two semantic categories occur in the description of assert, in the correct order. But the cover is *non-viable* nevertheless, because, not all mandatory categories in this particular description, namely, val, have occurred yet. When all expected mandatory categories occur, the cover will be considered viable. Further, viable or not, the cover is tentative because other possible covers exist and one of the other covers might prove to be globally more plausible. Now, suppose the sentence ends as follows:

"... *impaired"*

Then, since *impaired* is a domain-specific value, the mandatory category is also encountered; so assert is confirmed as one of several viable syntactic covers for the given words. To keep things simple for the present purposes, it is assumed that assert turns out to be the most plausible syntactic cover.

The covering category in this example, namely assert, was designated as an open class category. In general, if the category that has just been postulated as a cover

[2]This notion is very similar to that of open class words in languages. Non-open class concepts only have syntactic aspect, and correspond to "syntactic sugar" in language. See [Dasigi, 88] for more discussion.

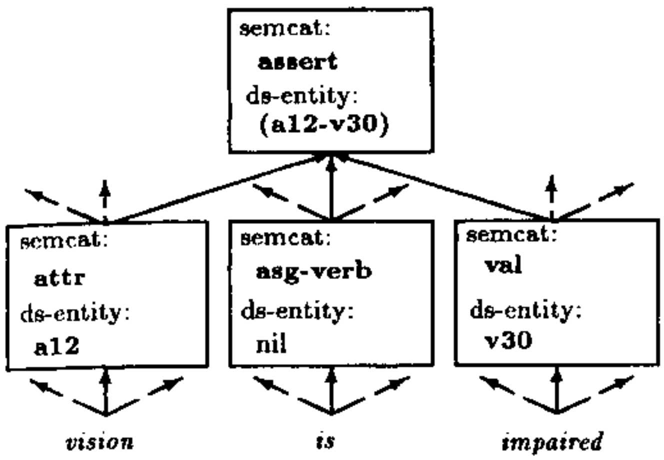

Figure 1: Interleaving of syntactic and semantic covering. The dashed arrows indicate other concepts that are evoked, e.g., other attributes named by "vision," other types of verbs that "is" evokes and many other concepts named by "impaired."

happens to be an open class category, it initiates semantic covering (with the standard notion of compositionality), thus integrating the use of both (that is, syntactic and semantic) aspects of knowledge. Now, we continue the example from the viewpoint of semantic covering. Recall, however, that this process is interleaved with syntactic covering, and does not necessarily follow it. See Figure 1.

The word "vision" is covered, among other things indicated above, by a concept that has the semantic category attr. Category attr is of open class and so not surprisingly the concept that covers "vision" also has a domain-specific entity, say a12, that uniquely characterizes it. In effect, this one linguistic concept covering "vision," has two facets: the semantic category attr and the domain-specific entity a12. Similarly, the word "impaired" is covered by, among others, a concept of the semantic category val that has the unique domain-specific entity, say v30, associated with itself. The verb "is," however, is covered by a concept of the category asg-verb and since asg-verb is a not an open class category, it does not have a corresponding domain-specific entity.

As already explained in the course of syntactic covering, assert is computed to be a syntactic cover; it also turns out to be a parsimonious syntactic cover. For semantic covering, what needs to be covered is the set of entities grouped under this category, i.e., a12 and v30, by identifying domain-specific associations that relate them. Definitions of parsimony and covering in the semantic route attempt to capture these intuitions, and the concept characterized by the assertional, semantic category assert and the domain-specific entity constituted by

(attr=a12, val=v30)

becomes the integrated parsimonious cover for the given sequence of words.

4 Discussion

The ability of parsimonious covering to handle ungrammatical sentences, as exemplified earlier, does not call for any special (or ad hoc) handling. It is a natural consequence of the very definition of covering itself. One could argue that a conventional production rule approach may easily be augmented to achieve the same effect. For instance, it might be possible that a description such as:

assert: attr asg-verb val,

where val is mandatory, can be encoded into the following production rules:

assert → attr asg-verb val
| attr val
| val
| . . . ,

the number of such rules can grow exponentially in the number of non-mandatory categories.

The previous paragraph should not be misconstrued as downplaying the significance of syntax in language. Indeed, the verb is plays a crucial role in disambiguating sentences such as,

"Flying planes is/are dangerous."

Our point is that omission of the copula in such sentences still does not make them incomprehensible. It does leave the sentence ambiguous, to be sure. At present, the semantic covering process does not worry about number agreement between the verb and subject, unless ambiguity arises. The underlying assumption here is that people try to make sense, and are not always grammatical.[3]

In summary, parsimonious covering provides a framework to view parsing natural language as an abductive process. A proof of concept is provided by implementing the basic ideas in an application independent interface shell. Admittedly, the semantic knowledge used is very restricted in nature, at the moment appropriate only to an object-oriented class of applications. While the presumed logical form is also, correspondingly, of a limited generality, in this work, it has allowed for the construction of an interface shell. Further, it has been noted that an abductive approach to word sense disambiguation may make it possible to define word senses naturally in context using descriptions of scenarios, as opposed to rules [Charniak and McDermott, 85; Dasigi, 88; Dasigi, 89].

Many significant linguistic issues remain to be answered in this framework, however. Two features of this preliminary work (namely, use of a semantic grammar-like descriptions that are closely related to the class of expert systems for which interfaces could be generated, and reliance on the assumption that ambiguity resulting from ungrammatically is resolvable in context) make it hard to predict the generality of the technique for unrestricted natural language. Our ongoing work further extends the model in the directions of using regular syntactic categories, and incorporation of further structure

[3]The majority of test inputs used by the prototype came from physicians' anonymous case descriptions, where insuring the grammaticality of sentences was, apparently, not the first priority.

into verb definitions (consequently making the logical form much more general) as follows.

The logical form employing triples, as in the work described here, specifies a value for an attribute of an object. Natural language sentences that can be represented by such logical form generally involve stative verbs. Such sentences describe a state rather than an action or an event. While such sentences capture most of the input text to diagnostic expert systems (which was the context in which the prototype was developed), they fall significantly short of characterizing open-ended English text, where actions and events are of a great significance. A more general logical form, such as the following as described in [Allen, 87], is more appropriate for sentences involving actions and events:

(specifier name type modifiers)

Here, the *specifier* specifies some detail of the concept (e.g., the tense for actions or events, the quantifier for nominal concepts, etc.), while the *name* serves to identify the concept. The concept itself is, to a large extent, characterized by its *type* (e.g., break-action, state, person, dwelling, fruit, etc.), while the *modifiers,* If any, allow for specifying attributes that modify the concept being represented.

The modifiers may involve recursive substructures representing other concepts. Such recursive substructures represent the partial semantic interpretations associated with grammatical subconstituents, while the entire logical form represents the semantic interpretation corresponding to the grammatical structure of the entire sentence. The overall logical form covers (i.e., is constructed from) the individual partial interpretations, and it does so "parsimoniously." An appropriate notion of parsimony here seems to be essentially the same notion of "structural minimality" or "cohesiveness" of the semantic structures built, much the same way the notions are used in this paper. It is hoped that this ongoing work will help answer the important questions raised above.

Acknowledgements

Past support from Jim Reggia of the University of Maryland and his comments on earlier versions of this paper are gratefully acknowledged.

References

[Allen, 87] Allen, J., 87: Natural Language Processing. Benjamin/Cummings Publishing Co., 1987, Chapter 7.

[Appelt, 90] Appelt, D., 90: A Theory of Abduction Based on Model preference, *AAAI Spring Symposium on Automated Abduction,* Stanford, March, 1990, pp. 67-71.

[Charniak, 83] Charniak, E., 83: Passing Markers: A Theory of Contextual Influence in Language Comprehension, *Cognitive Science,* 7(3), 1983, pp. 171-190.

[Charniak and McDermott, 85] Charniak, E. and D. McDermott, 85: An Introduction to Artificial Intelligence. Addison Wesley, 1985, Chapters 8 and 10.

[Charniak, 88] Charniak, E., 88: Motivation Analysis, Abductive Unification and Nonmonotonic Equality, *Artificial Intelligence,* 34(3), 1988, pp. 275-295.

[Dasigi, 88] Dasigi, V., 88: Word Sense Disambiguation in Descriptive Text Interpretation: A Dual-Route Parsimonious Covering Model. Ph.D. Dissertation. TR-2151, Department of Computer Science, University of Maryland, College Park, MD, 1988. Also available as WSU-CS-90-03, Department of Computer Science and Engineering, Wright State University, Dayton, Ohio.

[Dasigi, 89] D asigi, V., 89: A Non-Rule-Based Approach to a Natural Language Interface Shell. In *Methodologies for Intelligent Systems, 4,* Ras, Z. (Ed), North-Holland, New York, 1989, pp. 183-190.

[Dasigi, 90] Dasigi, V., 90: A Dual-Route Parsimonious Covering Model of Descriptive Text Interpretation. In *Computational Intelligence II - Proceedings of the International Symposium on Computational Intelligence 89,* Milano, Italy, Elsevier North-Holland, 1990.

[Hobbs, *et al.,* 88] Hobbs, J., M. Stickel, P. Martin and D. Edwards, 88: Interpretation as Abduction, *Proc. ACL-88,* 1988.

[Josephson, 90] Josephson, J., 90: On the "Logical Form" of Abduction, *AAAI Spring Symposium on Automated Abduction,* Stanford, March, 1990, pp. 140-144.

[Konolige, 90] Konolige, K., 90: A General Theory of Abduction, *AAAI Spring Symposium on Automated Abduction,* Stanford, March, 1990, pp. 62-66.

[Litman, 85] Litman, D., 85: Plan Recognition and Discourse Analysis: An Integrated Approach for Understanding Dialogues. Ph.D. Dissertation, TR 170, Department of Computer Science, The University of Rochester, Rochester, NY 14627.

[Peng, 85] Peng, Y., 85. A Formalization of Parsimonious Covering and Probabilistic Reasoning in Abductive Diagnostic Inference. Ph.D. Dissertation. TR-1615, Department of Computer Science, University of Maryland, College Park, MD 20742, January, 1986.

[Peng and Reggia, 87] Peng, Y. and J. Reggia, 87: Diagnostic Problem Solving with Causal Chaining, *International Journal of Intelligent Systems* 2, 1987, pp. 265-302.

[Pople, 73] Pople, H., 73: On the Mechanization of Abductive Logic, *Advance Papers from the 3rd IJCAI,* Stanford, CA, 1973, pp. 147-152.

[Reggia, 85] Reggia, J., 85: Abductive Inference, *Proc. of IEEE Symposium on Expert Systems in Government,* Kama, K. N., (Ed). McLean, VA, 1985, pp. 484-489.