GIT-CC-93/34

A Theory of Interaction and Independence in Sentence Understanding

Kavi Mahesh
mahesh@cc.gatech.edu

A THESIS PROPOSAL
Presented to
The Academic Faculty

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in Information and Computer Science

The Committee:
Dr. Kurt Eiselt (Advisor)
Dr. Susan Bovair (Psychology)
Dr. Ashwin Ram

Georgia Institute of Technology
February 1993

A Theory of Interaction and Independence in Sentence Understanding

Kavi Mahesh
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332-0280
mahesh@cc.gatech.edu

Abstract

Developing a complete and well-specified computational model of human language processing is a difficult problem. Natural language understanding requires the application of many different kinds of knowledge such as syntactic, semantic, and conceptual knowledge. To account for the variety of constructs possible in natural languages and to explain the variety of human behavior in sentence understanding, each kind of knowledge must be applicable independently of others. However, in order to efficiently resolve the many kinds of ambiguities that abound in natural languages, the sentence processor must integrate information available from different knowledge sources as soon as it can. Such early commitment in ambiguity resolution calls for an ability to recover from possible errors in commitment.

In this work, we propose a unified-process, multiple knowledge-source model of sentence understanding that satisfies all the constraints above. In this model, syntactic, semantic, and conceptual knowledge are represented separately but in the same form. The single unified process utilizes all knowledge sources to process a sentence. The unified process can resolve structural as well as lexical ambiguities and recover from errors it might make. We show that this model can account for a range of human sentence processing behaviors by producing seemingly autonomous behavior at times and interactive behaviors at other times. It is efficient since it supports interaction between syntactic, semantic, and conceptual processing. Moreover, the model aids portability between domains by separating domain-specific knowledge from general linguistic knowledge. We also present an early commitment, expectation-driven, bottom-up theory of syntactic processing that permits us to unify syntactic processing with semantic processing. We show several illustrative examples of ambiguity resolution and error recovery processed by our prototype implementation of the theory in a program called COMPERE (Cognitive Model of Parsing and Error Recovery).

Contents

List of Figures

1 Introduction

The proliferation of ambiguities in natural language processing together with the flexibility and ease with which human language understanders deal with them has captured the attention of researchers in artificial intelligence (AI), linguistics, psycholinguistics, and other cognitive sciences. Researchers trying to build cognitive models of language understanding have been puzzled by the architecture of the language understander which can produce the diverse behaviors observed in human language understanding, such as early commitment [Carpenter and Just, 1988; Frazier, 1987; Wanner and Maratsos, 1978], garden-paths [Crain and Steedman, 1985], delayed decisions [Stowe, 1991], parsing breakdown [Lewis, 1992], and error recovery [Carpenter and Daneman, 1981; Eiselt, 1989; Eiselt and Holbrook, 1991]. At the center of research on language understanding is an ongoing *modularity debate*—whether the language understanding faculties of the human brain have a modular architecture or whether they interact and are integrated to different degrees [Birnbaum, 1986; Birnbaum 1989; Crain and Steedman, 1985; Fodor, 1987; Frazier, 1987; Marslen-Wilson and Tyler, 1987; Tanenhaus, Dell, and Carlson, 1987]. Psycholinguistic studies have found evidence for modularity [Clifton and Ferreira, 1987; Fodor, 1983; Frazier, 1987] as well as for interaction between different faculties of language processing [Crain and Steedman, 1985; Tyler and Marslen-Wilson, 1977]. There have been many computational models proposed that are good at explaining either the modular or the interactive behaviors. There has been no computational model of language understanding that has satisfactorily explained all the different behaviors. However, recent models have attempted to cover a broader spectrum of phenomena than earlier ones [e.g., Jurafsky, 1991; Lewis, 1992; McRoy and Hirst, 1990].

A satisfactory answer to the modularity debate in language understanding would result in far-reaching benefits to the multiple disciplines that the issue has brought together, apart from its inherent value in revealing the nature of some part of human cognition. The focus in psycholinguistics would change from looking for evidence for different architectures of the language understander to finding out more about the "right" architecture. Natural language understanding programs in AI, which have not seen much success either in modeling human behavior or in producing general and flexible language understanding capabilities, would get a head start if they are built on the "right" architecture. In linguistics, there would be a clear distinction between formalisms that are meant to be tools for analysis and those that actually are used in language processing.

In AI, natural language understanding programs are built for different task environments such as story understanding, understanding design specifications, information retrieval and so on. Current architectures of language processing used in these programs result in inefficient systems with redundancies between task-specific lexicons and little portability apart from syntax [Peterson and Mahesh, unpublished]. A good architecture for the language processor would support the development of integrated intelligent systems [Goel and Eiselt, 1991] eliminating redundancies and increasing efficiency by providing for a rich interaction between syntax and other linguistic processes, and conceptual processing.

In this work, we propose a cooperative architecture with functionally independent syntactic and semantic processing that nevertheless interact with each other, as an answer to the modularity debate. Taking into account a wide range of results seen in AI and in psycholinguistics, we propose a unified model which can support interaction without sacrificing independence between the different faculties of language processing [Eiselt, Mahesh, and Holbrook, to appear; Holbrook, Eiselt, and Mahesh, 1992]. By unifying the processing of syntax, semantics, and other forms of linguistic knowledge, while keeping different knowledge sources independent of each other, this architecture can account for a range of apparently disparate behaviors observed in human subjects. To show how the architecture could produce the behaviors, we describe an implemented parser built on this architecture that can process written sentences, and build and disambiguate their structure and meaning.

Our model, COMPERE,[1] a Cognitive Model of Parsing and Error Recovery, reads written sentences in a left to right order and analyzes them at lexical, syntactic, and semantic levels. In doing so, it integrates information arising from its multiple knowledge sources at different levels and selects the interpretations that are best overall. It resolves different kinds of ambiguities such as syntactic and lexical ambiguities and recovers from errors made in its earlier decisions. It provides as output an interconnected set of syntactic and semantic structures as the representation of the meaning of the sentence. An important contribution of our model is to show that the same unified process can resolve ambiguities at different levels of sentence processing and recover from errors therein. Our model is able to exhibit seemingly modular processing behavior that matches the results of experiments showing the autonomy of different levels of language processing [e.g., Forster, 1979; Frazier, 1987] but is also able to display seemingly integrated behavior that matches the results of experiments showing semantic influences on syntactic structure assignment [e.g., Crain and Steedman, 1985; Tyler and Marslen-Wilson, 1977].

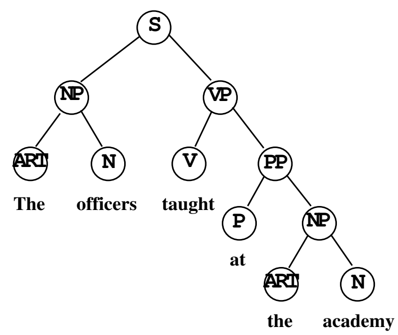

Figure 1: Garden Path: Main-Clause Interpretation.

For example, consider the sentence

(1) The officers taught at the academy were very demanding.

This is syntactically ambiguous at the verb *taught* since there is no distinction between its past-tense form and its past-participle form. In the simple past reading of *taught,* it would be the main verb with the corresponding interpretation that "the officers taught somebody else at the academy." On the other hand, if *taught* is read as a verb in past-participle form, it would be the verb in a relative clause, a reduced relative since there is no relative pronoun (such as *who*) marking the clause. The reduced relative would correspond to the meaning "the officers who were taught by somebody else at the academy ..." The sentence processor does not have the necessary information at this point to decide for sure which of the two is the appropriate interpretation. Yet people choose the first interpretation [e.g., Frazier, 1987] even though the relative clause is the right interpretation of this sentence. Only when they read up to the real main verb *were* do they realize that *taught* must have been in a reduced relative clause. This results in a garden-path effect upon reading this sentence for the first time, that is, the human sentence processor is led up a "garden path" and has to backtrack when later information shows that it was the wrong path to take. This behavior is not influenced

[1] A *Compere* is one who introduces and interlinks items of an entertainment.
— Chambers Twentieth Century Dictionary.

by semantic or conceptual preferences and can be perceived as a modular behavior [e.g., Frazier, 1987]. The output from our proposed model for this sentence is shown in Figures 1 and 2. It shows both the main-clause interpretation of *taught* initially pursued and the final reduced-relative clause interpretation resulting from an error recovery.

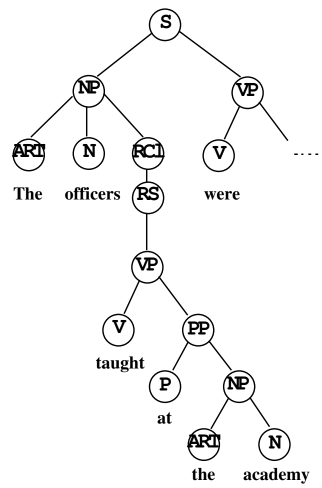

Figure 2: Garden Path: Reduced Relative Clause.

Now consider a sentence obtained from (1) above by changing the subject noun *officers* to *courses.*

(2) The courses taught at the academy were very demanding.

If there is conceptual[2] information to the effect that *courses* can only be *taught* and cannot teach themselves, then one can eliminate the main-clause reading of *taught* right away. Only the reduced relative clause interpretation is pursued so that there is no garden-path effect. The output from our model for this sentence is shown in Figure 3.[3] This shows how the same process that previously exhibited modular processing behavior can also produce interactive processing behavior when conceptual information is available.

We begin this article by stating the claims we make in this work in the next section. We then present overviews of experimental observations in human sentence understanding behavior in support of the claims. This is the bulk of the next section, where we not only try to characterize the modularity debate but also try to give explanations for the behaviors. We then summarize the architectural lessons learned from the study of behaviors and models by describing a set of dimensions of the space of possible sentence processing architectures. This sets the stage for a description of a series of significant classes of sentence processing models, both computational and psychological. Following this is a discussion of the assumptions we make, a characterization of the specific task the program performs, and then an architectural description of our own model, the

[2]The terms *conceptual* and *semantic* are used in the following sense in this work: *Semantic* knowledge is the grammatically relevant aspects of meaning (See Frawley, 1992); *Conceptual* knowledge refers to the aspects of meaning that are not relevant grammatically, i.e., those that are not reflected overtly in the surface form of a sentence in the language.

[3]The output in Figure 3 is a simulated output. The program does not actually produce this output for (2) at this point.

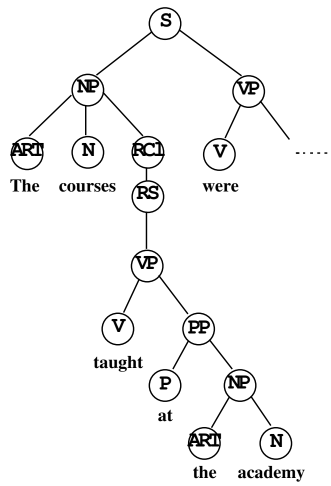

Figure 3: Reduced Relative Clause: No Garden Path.

representations of knowledge it employs, the unified process which operates on the representations, and examples showing the model in action. We then show how the unified process resolves structural and lexical ambiguities and how it recovers from its errors. Throughout this section, we try to provide justifications for many decisions we have made such as the form of the representations and the control structure the model uses. This is followed by a specification of how we plan to validate this model by making the program process a set of sentences. This section includes a set of sentences and the behavior we expect from the program for those sentences. Finally, we conclude the report by listing the problems and issues we address, the contributions this model makes, and further work we propose to take up both in fulfillment of the doctoral thesis requirements and beyond.

2 Claims

The purpose of this work is to show that a unified process applied independently to multiple knowledge sources enables the sentence processor to produce and explain the variety of behaviors observed in human language processing. We propose to develop a computational model of sentence understanding based on the unified process model to demonstrate that such a model is computationally feasible and to explore its properties. Briefly, we state our claims thus:

1. **Functionally independent integrated processing:** Different knowledge sources such as syntactic and semantic knowledge are accessible and applicable independent of one another. The Integrated Processing Hypothesis states that the language processor makes decisions as early as possible by integrating information from all available knowledge sources [Birnbaum, 1986]. The integrated processing hypothesis does not require integrated representations. Sentence processing can be an integrated process without having to integrate different sources of knowledge a priori.

2. **Unified process:** The sentence processor has a single unified process; this single process can analyze the syntax and semantics of a sentence and resolve structural as well as lexical ambiguities. The unified process is a combination of bottom-up and top-down processing mechanisms.

3. **Error recovery:** In order to meet time and memory resource limitations, the sentence processor must make commitments early and must be able to recover from its errors. The unified process can recover from errors in resolving lexical as well as structural ambiguities.

Though each of these claims addresses a different aspect of language processing, it is the combination of these claims that makes this work unique among theories of sentence understanding. They are related to one another and together they define a coherent theory of sentence understanding. The unified process, for instance, is a good way of meeting the need for functionally independent yet interactive processing. The claims for the kinds of early commitment proposed in syntactic and semantic processing require error recovery capabilities. Error recovery, in turn, is facilitated by the unified process.

3 Evidence for the claims

We now consider the import of each claim and take a look at evidence from related research in natural language processing, psycholinguistics and neurolinguistics to see why we are led to address these issues and make these claims.

Natural languages are replete with ambiguities. Understanding natural languages is all about coming up with possible interpretations in each faculty[4] of language processing and choosing one or more that are most preferred in the current context, by integrating information provided by various knowledge sources. Language understanders make errors at times in making decisions at ambiguous points since sufficient information may not be available yet at those points. Information coming later on in the sentence might alter the preferences for different interpretations. These ambiguities and errors occur at different levels of language processing. People are capable not only of resolving ambiguities but also recovering from such errors in many cases [Eiselt and Holbrook, 1991; Eiselt, 1989; Holbrook, et al., 1992; Holbrook, Eiselt, Granger, and Matthei, 1988].

Traditionally, the task of understanding a natural language has been decomposed into several subtasks, one each at the different levels of language processing, such as the morphological, lexical, syntactic, semantic, pragmatic, and discourse levels. How are these tasks composed to carry out the task of language understanding? What is the architecture of the human language understander? Do people exhibit the same behavior in resolving ambiguities, and recovering from error therein, at different levels? How do the levels interact with each other? These are questions that we attempt to answer in the course of this work.

3.1 Functionally independent integrated processing

According to the integrated processing hypothesis, every kind of knowledge available to the sentence processor must be applied at the earliest opportunity in making decisions while processing a sentence [Birnbaum, 1986; Birnbaum, 1989; Schank, Lebowitz, and Birnbaum, 1980]. (See also the "Immediacy principle" of Carpenter and Just [Carpenter and Just, 1988].) Functional independence, on the other hand, says that different knowledge sources are usable independent of each other [Caramazza and Berndt, 1978]. If, for some reason, a knowledge source fails to provide any information for making a decision, information arising from other knowledge sources can still be applied to make the best decision based on available information. Integrated processing has been interpreted to imply that different kinds of knowledge such as syntactic, semantic, and conceptual knowledge must be integrated a priori, thereby making the use of one dependent on others [Birnbaum and Selfridge, 1981; Lebowitz, 1983; Lehnert, Dyer, Johnson, Yang, and Harley, 1983; Ram, 1989; Riesbeck and Martin, 1986a; Riesbeck and Martin, 1986b]. Integrated systems, even when they represented different types of knowledge separately from one another [e.g., Lytinen, 1986], did not retain functional independence between the knowledge types; their processing architectures made the use of one type of knowledge dependent on the use of another.

In this work, we claim that knowledge sources can be represented independent of each other to retain functional independence between them. However, information arising from the different sources is integrated during the processing of a sentence so that decisions are still made by applying all kinds of knowledge available at every decision point. Information about the sentence coming from each source should be considered at the earliest, but the sources themselves should not be integrated a priori in a monolithic representation.

[4]The term faculties of language refers to things such as syntax and semantics. The neutral term faculty is used to eliminate the architectural commitments associated with other terms such as module, stage, phase, or level. However, we do not intend the term faculty to refer to isolable neuro-anatomical regions.

Based on examining human language processing behavior, there are at least two broad conclusions we can make: some decisions made in resolving ambiguities at a certain level of language processing seem to be based only on knowledge available at that level while some other decisions seem to be made by the interaction of knowledge at different levels.

3.1.1 Evidence for Integration

Intuitively, since the purpose of language is its communicative function and not syntactic grammaticality judgment, the human language processor should use the meaning of the text as early as possible in making any decision during the course of language understanding [Crain and Steedman, 1985; Tyler and Marslen-Wilson, 1977]. Thus, the human sentence understander is forever trying to integrate information arising from knowledge sources at different levels in order to come up with a single interpretation that makes the best sense overall (i.e., at the syntactic, semantic, conceptual, and other levels). This goal to integrate the preferences of every level results in immediate interaction between the different levels such as syntax and semantics.

In processing some kinds of sentences, people exhibit behaviors that show that they exploit an immediate feedback from semantics and pragmatics in resolving structural ambiguities in syntax. People garden-path on reduced-relative ambiguous structures such as in (1).

(1) The officers taught at the academy were very demanding.

However, sentence (2) which has the same surface structure as (1), does not result in garden-path behavior [Stowe, 1991].

(2) The courses taught at the academy were very demanding.

Since there is no difference in the apparent structure of the two sentences, the human sentence processor must be using some information about the meaning of the words "officers" and "courses" in relation to the meaning of "taught" and other words in the sentence to choose the main-clause interpretation in (1) and the relative-clause structure in (2) when reading the first verb "taught" in the two sentences. For example, such higher-level information might be the knowledge that only animate objects can be actors of teaching.

The More Constraints the Better: Natural languages appear to allow a high degree of local syntactic ambiguity [Crain and Steedman, 1985]. If the sentence processor supports interaction between syntax and higher levels, and if it produces fully interpreted semantic entities corresponding to incomplete fragments of the sentence, the context in which these entities are evaluated can be a powerful source of redundancy. Such redundancy between information arising from knowledge sources at different levels enables the sentence processor to handle a degree of local syntactic ambiguity which in purely syntactic processing terms would be intolerable.[5] In other words, combining preferences from different levels as early as possible might be imposing more constraints and shrinking the set of feasible interpretations rather than creating an information overload on the real-time processor.

Experimental Evidence Evidence for the interaction of semantic[6] and contextual information in immediate structural ambiguity resolution comes from the experiments of Tyler and Marslen-

[5] See [Jacobs, Krupka, McRoy, Rau, Sondheimer, and Zernik, 1990] for an example of a typical sentence from a newspaper that resulted in over 100 parses without the use of any semantic or conceptual preferences.

[6] Note that the term 'semantic' is used in a loose sense in this section and does not conform to its definition as the grammatically relevant aspects of meaning.

Wilson (1977), Crain and Steedman (1985), Cupples and Stowe [Holmes, Stowe, and Cupples, 1989], and Taraban and McClelland (1988). Tyler and Marslen-Wilson (1977) used Adjective-Verb ambiguous word-pairs such as "landing planes" in (3)

(3a) If you walk too near the runway, *landing planes...*

(3b) If you've been trained as a pilot, *landing planes...*

with a naming task to show that semantic information does affect the choice of syntactic structure even before the end of a clause. Their experiments showed that syntax does not behave autonomously until the boundary of a clause. The naming latency for a probe inappropriate with respect to the semantic context of the incomplete clause was longer than that for an appropriate probe. This shows that subjects had already used prior semantic context to choose one syntactic structure over another before the clause was complete. In fact, they even argued that alternative syntactic structures that are not preferred by the semantic context may never be computed explicitly. Their experiments showed that the differences between naming latencies to appropriate and inappropriate probes were similar following both ambiguous and unambiguous word pairs. However, this evidence did not directly demonstrate that only one syntactic structure was ever considered by the sentence processor.

Crain and Steedman (1985) used sentences with complement and relative clauses to show that semantic and referential context information can steer the sentence processor towards one or the other kind of syntactic structure. For instance, they used sentence pairs such as in (4).

(4a) The teacher taught by the Berlitz method passed the test.

(4b) The child taught by the Berlitz method passed the test.

Sentences such as (4b) were judged grammatical by the subjects significantly more often than those of type (4a). The semantic distinction between the two sentences, namely, that a teacher is more likely to teach whereas a child is more likely to be taught, explains the difference in grammaticality judgment. Their experiments showed that such semantic and contextual interaction happens well before any sentential or clause boundary is encountered. They argued that garden-pathing is a contextual phenomenon and can both be prevented and induced by the context in which a sentence is processed.

Cupples and Stowe [Stowe, 1991] used sentences such as in (1) and (2)

(1) The officers taught at the academy were very demanding.

(2) The courses taught at the academy were very demanding.

and measured word-by-word reading times to establish that semantic information such as the animacy of the subject influences the immediate assignment of syntactic structure.

Taraban and McClelland (1988) investigated the effects of sentential context preceding a prepositional phrase and showed garden-pathing effects in both directions based on differences in context. They used both minimal and nonminimal structures as the preferred ones in the contexts they created. An example of the kind of sentences they used is shown in (5).

(5a) The reporter exposed corruption in government.

(5b) The reporter exposed corruption in the article.

They found that people had no structural preference such as minimal attachment [Kimball, 1973; Frazier, 1987]. Instead, they found a significant interaction with sentential context. For instance, when the context predicted high attachment, VP attachment was easier (as in (5b)), but when the context predicted nominal attachment, attachment to the NP was easier instead ((5a)). They were able to show garden pathing in both directions based on differences in context, providing clear evidence against autonomy.

3.1.2 Evidence for Functional Independence

The term functional independence is used to describe that feature of the architecture which supports interaction between the different levels without sacrificing the independence between them. It allows each faculty to function if others fail for some reason [Caramazza and Berndt, 1978; Eiselt, 1989]. Functional independence need not necessarily mean neurological independence, according to which there are separate parts of the brain corresponding to the different functions of language processing. There are several reasons for hypothesizing a functionally independent architecture for the sentence processor. Intuitively, human sentence understanding behavior shows the independence of syntax and semantics. To illustrate this point, one can argue that, for instance, people can judge grammaticality independent of meaning as in sentences like (6) which make little sense if any [Chomsky, 1957].

(6) Colorless green ideas sleep furiously.

On the other hand, people can put up with imperfect syntax and get the meaning out of ungrammatical strings of words. For instance, even in the total absence of syntax as in (7) we still get some meaning out of the text.

(7a) Skid crash hospital. [Winograd, 1973]

(7b) Fire match arson hotel. [Charniak, 1983]

Lack of independence of syntax also leads to difficulties in accounting for syntactic generalizations across different semantic entities. An example of a syntactic generalization exhibited by the language processor in a variety of constructs is the Minimal Attachment principle [Frazier, 1987; Kimball, 1973] which subsumes a class of preferences for a syntactically minimal interpretation. A preference for a main-clause analysis rather than a reduced-relative analysis in (1), a preference for a simple-direct-object (rather than a sentential-complement) analysis of the ambiguous phrase in (8), and a preference for NP (rather than sentential) conjunction of the ambiguous phrase in (9), can all be explained by the syntactic generalization of minimal attachment. A variety of semantic and referential preferences using the specific concepts and contexts in the above sentences would have to be employed to explain the same set of behaviors if the language processor were unable to impose syntactic preferences independently.

(8) Mary knew *the answer* ...

(9) John kissed Mary *and her sister* ...

Syntactic Independence: Decisions in resolving ambiguities at lower levels such as the syntactic level seem to be made at times autonomously without taking the preferences of the higher levels into account [Ferreira and Clifton, 1985; Forster, 1979; Frazier, 1987; Frazier, Clifton, and Randall, 1983]. For instance, a choice between two different syntactic structures for a part of a sentence might be made using structural preferences without taking into consideration any semantic or pragmatic bias. This kind of behavior could result if there is no preference at the higher levels.

More interestingly, in some cases, people seem to exclude preferences from higher levels in making lower-level decisions. For instance, in (10)

(10) The editor played the tape *agreed the story was big.*

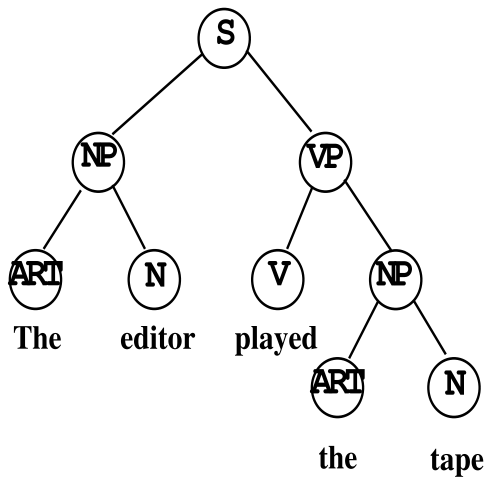

Figure 4: Main-Clause Interpretation.

people initially prefer the main-clause analysis (shown in Figure 4) even if the context of discourse is biased towards the reduced relative-clause structure such as when there are two editors one of whom has been played a tape (Fig. 5). People are led along the "garden path" of the main-clause structure and have to reprocess the text when they encounter the second verb. Such garden-path behavior happens irrespective of biasing semantic contexts in sentences such as (10). Thus, people must be making their decisions based on syntactic information alone.

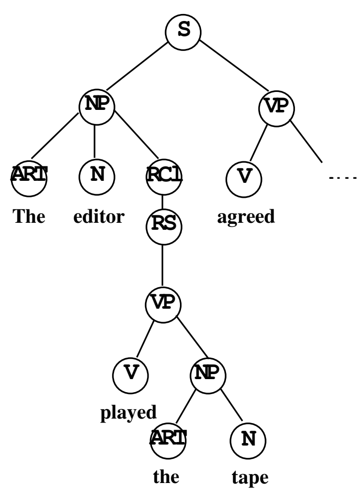

Figure 5: Reduced Relative Clause.

Similarly, in (11)

(11) Sam loaded the boxes on the cart *onto the van.*

they prefer the simpler VP-attachment of "on the cart" (as in Figure 6) than the complex NP shown in Figure 7 even in contexts where there are two sets of boxes (and it was asserted that one set had been placed on a cart).

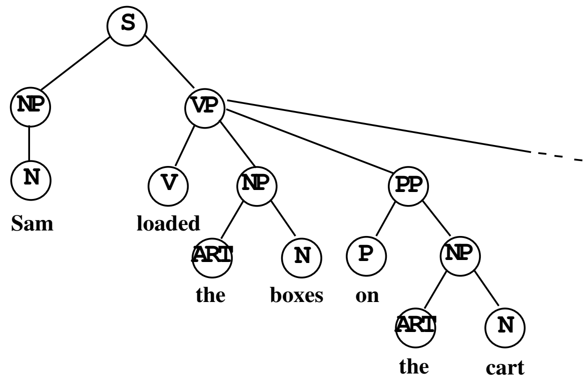

Figure 6: Minimal Prepositional Attachment.

These behaviors can be explained by a preference called the Minimal Attachment criterion [Frazier, 1987; Kimball, 1973] according to which the sentence processor prefers that structure which has the least number of nodes in it. That is, the interpretation which is minimal in structure gets selected. Minimal attachment is a very general criterion applicable to many different kinds of sentences and has been used as the criterion for syntactic decisions made at local structural ambiguities [Frazier, 1987].

However, there are competing structural preferences such as Right Association [Kimball, 1973] which seem to work at other times in sentences where minimal attachment fails. Right Association or Late Closure says that the incoming constituent is attached to the current structure rather than a previous structure higher up in the parse tree. For instance, in sentence (12),

(12) I saw the man with the horse.

right association would attach the PP to the NP (as shown in Figure 8) and not to the VP as minimal attachment would have it. In this sentence, the minimal interpretation would be turned down by semantic criteria (since a horse cannot be used as an instrument for seeing) and the structure preferred by right association turns out to be the correct parse.

Another source of evidence for autonomy comes from psycholinguistic experiments using eye-movement recording techniques to measure word-by-word reading times [Clifton and Ferreira, 1987; Frazier, 1987]. Such studies show that people take longer to read the disambiguating part of a garden-path sentence such as the emphasized parts of (10) irrespective of contextual bias.

(10) The editor played the tape *agreed the story was big.*

Further intuitive support for autonomy comes from the complexity of interaction and that of integrating preferences from several higher levels in making online decisions. The speed and automaticity of decisions at the lower levels of sentence processing seem to suggest a fast, autonomous, automatic process [Fodor, 1983; Fodor, 1987].

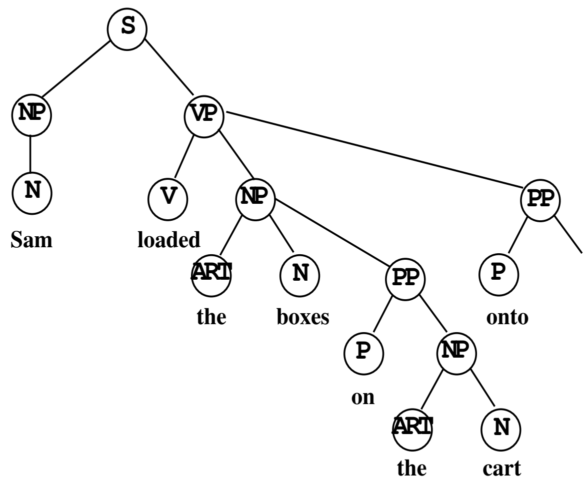

Figure 7: Non-Minimal Prepositional Attachment.

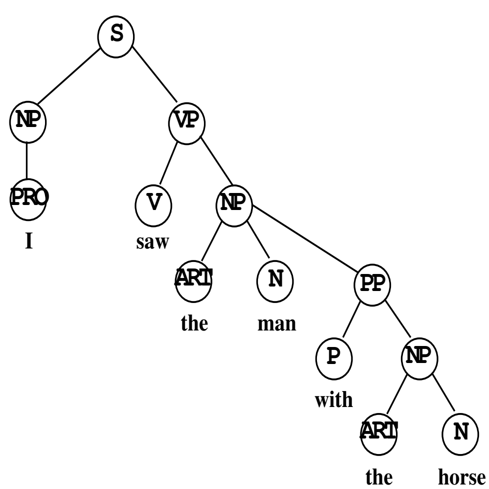

Figure 8: Right Association.

Evidence from Aphasic Studies: An important source of evidence for functional independence between levels of language processing lies in behavioral studies with aphasic subjects [Caramazza and Berndt, 1978]. People who have suffered focal neural damage in the Broca's area, Broca's aphasics, exhibit impaired syntactic processing abilities. Those with a damage in a different part, Wernicke's aphasics, appear to retain syntactic processing abilities but have trouble producing coherent semantic content. Studies of language production and comprehension in aphasic people have resulted in the conclusion that although processing at different levels may interact, they are functionally and neurologically independent and can be selectively affected by damage to the brain.

Cross-linguistic Aphasia and the Competition Model: While the above studies were mostly done with English speaking aphasics, some recent cross-linguistic studies of aphasics have not shown such a clear distinction in the kind of breakdown in language processing behavior occurring in aphasics with different parts of their brain affected by focal neural damage [Bates, Wulfeck, and MacWhinney, 1991]. An interactive activation model of language processing called the *Competition Model* has been proposed by Bates and MacWhinney [Bates, et al., 1991] to account for the cross-linguistic data. This model is based on the notion that there are a set of cues (such as word-order, case terminations, prepositions, and so on) in each language which determines the semantic role assignment in a sentence. Different cues have different cue validities in a language. For instance, in English, word order has the highest cue validity. A cue also has a cue cost associated with it which can be measured by quantities such as the amount of memory required to store and compare cues across a sentence for agreement, or degree of perceptual difficulty posed by different types of case markers in speech comprehension. The competition model is supposed to account for a variety of language processing phenomena by graded activation in a dynamic network. These cross-linguistic results have been interpreted as evidence that it was the "accidents" of English which resulted in such a clear correlation between the region of neurological damage and the faculty of language processing affected. Deficiencies in the linguistic behavior of aphasics are not a simple result of the loss of particular types of knowledge or linguistic cues. Different linguistic cues are still largely available for language processing. Nevertheless, systematic differences between patient groups such as Broca's and Wernicke's aphasics are observable in cross-linguistic aphasiology. In order to account for these findings, competition models [Bates, et al., 1991] should make sure that the cue costs change differently for different kinds of knowledge. In other words, the cost of processing different kinds of knowledge changes differentially when there is focal neural damage. This is clear evidence that the use of different kinds of knowledge is at least partially independent of each other. All types of knowledge cannot be represented together in an inseparable form. Cross-linguistic evidence is against neurological, but not functional independence between the different faculties of language processing. It should also be noted that in cross-linguistic studies, one cannot assume that what is syntactic knowledge in one language is also syntactic knowledge in another language. The terms syntax and semantics may refer to different sets of cues in different languages.

In the "family" of language processing faculties, there is more than one member contributing to the overall communicative function of language. If for some reason circumstances force one or more members to lose their contributions, language processing still goes on with the contributions of the others. However, when the going is good, all the faculties interact with each other and together accomplish the goal of language understanding.

3.1.3 Synthesis from evidence for integration and independence

Since the above evidence shows that different kinds of knowledge are integrated early in sentence processing and at the same time, are applicable independently of each other, we would like to show

how the architecture of the human sentence processor can support both integrated processing and functional independence. We claim that integrated processing does not preclude functional independence. A number of people have (implicitly) assumed that integrated processing implies integrated representations. For instance, models such as CA, IPP, and the recent model by Jurafsky have no separable representation of syntactic knowledge [e.g., Jurafsky, 1992; Lebowitz, 1983]. In fact, the approach taken by Jurafsky, namely, to represent every kind of knowledge in integrated knowledge structures called *grammatical constructions,* rules out the independent access and application of any one type of knowledge.

Integrated processing showed that the human sentence understander does not have a sequential architecture and gave semantics and world knowledge their proper role in sentence understanding. However, without functional independence between syntax and semantics, they could not maintain the proper role of syntax. The role of syntax in language understanding is to determine the hierarchical and (left-to-right) order relationships between parts of a sentence. These relationships are important cues in assigning semantic roles to the parts. A model gives syntax its proper role if such relationships are applied to make decisions in resolving an ambiguity, though such information may not be necessary or sufficient in every case to assign the proper roles. Syntax was ignored in integrated processing models attempting to fill the proper role of syntax by semantic or conceptual expectations [Birnbaum and Selfridge, 1981; Lebowitz, 1983; Lehnert et al., 1983]. However, "the use of such a criterion in parsing is bound often to be unreliable, since the reason for having syntax at all is presumably that real events frequently contradict such expectations" [Crain and Steedman, 1985; Forster, 1979]. In integrated models, information available right away from syntax could not be utilized in making sentence processing decisions. For instance, Lytinen's semantics-first model of language processing called MOPTRANS [Lytinen, 1986] used animacy cues to determine the actor of a sentence though syntax would tell right away who the actor was (by using the fact that the actor occurs to the left of the verb in an English sentence in active voice). By retaining functional independence, we can give each kind of knowledge that goes into sentence understanding its proper role. Only then can a model of sentence processing adhere to the integrated processing hypothesis [Birnbaum, 1986] by applying knowledge of any and all types whenever possible to make ambiguity-resolution decisions. We can make the best use of each kind of knowledge available and thereby extend the scope of the model in terms of the kinds of sentences it can process and for which it can explain human behavior.

3.2 Error recovery

Language understanders have to make early commitments in resolving ambiguities so as to cope efficiently with the vagaries of natural languages. As a result, language understanders commit errors at times in making decisions at ambiguous points since sufficient information may not be available yet at those points. Information coming later on in the sentence might alter their preferences for different interpretations. People are capable not only of resolving ambiguities but also recovering from such errors in many cases [Carpenter and Daneman, 1981; Eiselt, 1989; Eiselt and Holbrook, 1991]. Not all errors in resolving an ambiguity lead to a garden path. People can often recover from the errors they make locally when later input shows the error.

Lexical and Pragmatic Error Recovery: The ATLAST model [Eiselt, 1989] proposed a Conditional Retention Mechanism [Holbrook et al., 1988] for error recovery in resolving lexical and pragmatic ambiguities. According to this, the sentence processor selects the best interpretation in the current context and makes an early commitment when possible. However, it does not discard the alternative interpretations; it retains them for possible later use. The retained alternatives are suppressed for the present but are not completely inactive. If later text proves an earlier decision wrong, the sentence processor reactivates the retained alternatives to reevaluate the decision. It would then switch to the interpretation that is now best with respect to the new information as well as the earlier. This theory of conditional retention and error recovery in lexical processing was verified psychologically by experimental results [Eiselt and Holbrook, 1991; Holbrook, 1989]. The computational model demonstrated that the same mechanism could account for recovery from errors in resolving pragmatic ambiguities.

Syntactic Error Recovery: Studies of structural ambiguity resolution in syntax showed error recovery behaviors similar to that in semantics as described above. Stowe's model (1991) of limited delayed decision in syntactic ambiguity resolution (see below), though not a computational model, had much in common with the ATLAST model of lexical and pragmatic ambiguity resolution. Stowe's model was based on experimental evidence for delayed decisions (see below). Both models proposed an early commitment where possible. Both models had the capability to pursue multiple interpretations in parallel when the ambiguity forced it. Both models explained error recovery as an operation of switching to another interpretation maintained in parallel by the sentence processor. Conditional retention modeled resource constraints on the processor just as limited delay does. Finally, both models made decisions by integrating the preferences from syntax and semantics.

3.2.1 Synthesis from Error Recovery Evidence

Since the sentence processor is capable of recovering from errors in lexical as well as structural ambiguity resolution, and since the error recovery behavior seems to be the same in both kinds of ambiguities, we argue that the simplest theory which accounts for this behavior is one in which there is a single unified method that can recover from errors at different levels of sentence processing. In Section 5, we demonstrate the feasibility of this theory with a computational model. This demonstration lends support to the claim that sentence understanding is a unified process. Moreover, it shows the feasibility of a theory of syntax processing and structural ambiguity resolution put forward by Stowe (1991).

From an AI point of view, error recovery is essential for the integration of a sentence processor with a reasoner or problem solver. An online model of sentence processing cannot wait for feedback from the reasoner since the reasoner may take an unbounded amount of time to generate such feedback even when it can provide feedback given a partial interpretation of an incomplete sentence. At a later time, when the reasoner can provide useful feedback, the sentence processor must be able to change its interpretation of a sentence in accordance with the feedback. Thus error recovery capabilities are crucial to the design of an integrated language processing and reasoning system.

3.3 Unified process

The unified process claim for syntactic and semantic error recovery can be interpreted to say that the human sentence processor is best described as a single unified process operating on several independent knowledge sources [Eiselt et al., To appear; Holbrook et al., 1992]. To say that the two kinds of errors are handled the same way is to say that the overall control structure of the sentence understanding process is the same for syntax and semantics. This is supported by Stowe's model of limited delayed decision among other things (see below); the process of syntactic analysis by multiple access, early commitment, and retention of alternatives is itself very similar to the process of semantic analysis. These findings enable us to take the more parsimonious alternative of putting language processing together with a single unified process as opposed to a commitment to many different processes.

Since the overall process of sentence understanding can be viewed as resolving ambiguities by selecting the best from a set of interpretations and making amends for any erroneous decisions along the way, it is reasonable to propose a single unified process that carries out the overall job of sentence understanding. The differences if any may lie in some of the operations used on the different kinds of knowledge that result in the proposed interpretations and the differential preferences for them.

The unified process claim is also derived in part from the intuition that syntactic analysis must be more like semantic analysis than has traditionally been projected by theories of grammar and parsing. Syntactic processing should be such that there is an effective way to interact with semantic analysis. This issue is especially important when we look at cross-linguistic studies. Syntactic theories have been largely designed to deal with word order as the single most important cue, since that is true of English. When we look at other languages which rely more on other cues (such as case markings and semantic cues) than word order to determine the relationships between objects and events in a sentence, there seems to be very little information useful for semantic role assignment that can be extracted by the kind of syntactic parsing commonly proposed. For instance, phrase-structure rules which are the most common form of representing syntactic knowledge are inherently based on the notion of word order and are not suitable for languages where a sentence is a set of syntactic elements instead of a sequence.[7] Syntactic analysis in such languages would be much less different from semantic analysis than it has been in current theories of language understanding. For instance, one might have to analyze case markings extensively to determine the same information that is determined by word order in English. Since syntax and semantics differ significantly in their reliance on word order to determine the meaning of a sentence, a lesser validity attached to the word-order cue (see the description of competition models in Section 3.1.2) in a language would make syntactic and semantic analyses less different in that language. It must be noted in addition that what is called syntactic knowledge or processing in one language need not be considered syntactic in another language.

Recent studies in cross-linguistic aphasiology [Bates, et al., 1991] have shown similarities between the way in which different faculties such as syntax and semantics are affected by neural damage even when the damage is focal. Neither faculty alone suffers significantly more damage than the other; impairment is always graded. The results observed in cross-linguistic aphasic behaviors point towards common processing mechanisms operating on representations of syntactic and semantic knowledge in a common format .

> "These similarities are compatible with models in which lexical and grammatical forms are represented in a common format and/or accessed by a common set of processing mechanisms. ..." [Bates, et al., 1991]

Uncontrolled interaction between separate processes as it happens in certain connectionist network models for instance [Cottrell, 1985; Waltz and Pollack, 1984; Waltz and Pollack, 1985], does not seem to be capable of producing the kind of ambiguity resolution behaviors observable in limited delayed decisions and error recoveries. The sentence processor needs a control structure monitoring the interaction between the different faculties, arbitrating their preferences, and negotiating optimal decisions. Interaction between syntactic and semantic processing appears to be so intense and continual, yet controlled, that the overall process is best described as one unified process.

[7] Note, however, that few languages have a completely free word order. It is more common to find a mixture where a sentence can be any of a set of permissible sequences, some more appropriate to the context than others.

Unified Process and Delayed Decisions Another advantage of a single unified process appears in delayed-decision phenomena in ambiguity resolution. Let us now look at different situations where decisions are delayed to see how a single process could help model such behavior. Immediate interaction with information from higher levels need not mean an immediate decision in resolving ambiguities. The human sentence understander does not always select an alternative immediately. It is capable of delaying a decision and waiting for more information which might resolve the ambiguity. It can pursue multiple interpretations in parallel[8] until such information arrives or until it is forced to choose one at the cost of others by its own resource limitations. Such delayed decision behaviors have been observed at different levels of sentence processing.

Delay in Categorial Ambiguities When there is a categorial lexical ambiguity, that is, when a word can be assigned more than one syntactic category such as noun, verb, and so on, people pursue multiple categories until all but one are ruled out by following text. This was shown experimentally by Seidenberg, Tanenhaus, Leiman, and Bienkowski, (1982) who by cross-modal-priming studies found selective access of lexical entries for certain noun-noun ambiguous words (eg., spade as in (13)) but multiple access for noun-verb ambiguous words (eg., watch as in (14)).

(13a) You should have played the spade.

(13b) Go to the store and buy a spade.

(14a) I bought a watch.

(14b) They decided to watch.

Similarly, eye-movement studies by Frazier, Rayner and others, [Frazier and Rayner, 1987; Frazier, 1989] have shown that categorially ambiguous words do not result in immediate selection. In one experiment, they used sentences in (15) and found that the target items (emphasized) in the ambiguous forms (15a and 15b) take less time to read than the disambiguated sentences (15c and 15d); however, the following (disambiguating) items take longer to read in the ambiguous forms (15a and 15b) than in the disambiguating forms (15c and 15d). This is precisely what one would expect based on the notion of delayed decision in categorial ambiguities.

(15a) The *warehouse fires* numerous employees each year. (Ambiguous, N-V)

(15b) The *warehouse fires* harm some employees each year. (Ambiguous, A-N)

(15c) That *warehouse fires* numerous employees each year. (Disambiguated, N-V)

(15d) Those *warehouse fires* harm some employees each year. (Disambiguated, A-N)

Immediate Decision in Subcategorization While the sentence processor delays decisions at lexical category ambiguities for basic syntactic categories (such as nouns and verbs), it makes an immediate selection, argues Frazier, (1989) when there are ambiguities in subcategorization. For instance, if a verb can be both transitive and intransitive, the sentence processor selects one

[8] While the alternative explanation that the sentence processor conducts a partial, underdetermined analysis of the sentence so that it can continue any of the multiple analyses after seeing later evidence, is an interesting one, it cannot explain garden path behaviors caused by the inferiority of certain interpretations [Frazier, 1987].

subcategorization based only on general syntactic preferences such as the minimal attachment principle. For instance, in (16), *know* can take either a noun phrase or a sentential complement.

(16) Karen knew the answer to the difficult problem was correct.

However, people show processing difficulties when they see the disambiguating phrase "was correct." This shows that the processor goes ahead and assigns the simpler noun phrase structure taking *the answer* as the direct object before the time when subcategorization ambiguities are resolved. Note, however, that this is not evidence against immediate semantic interaction since there is no semantic preference significant in this context. Moreover, there is no definitive evidence about how subcategorization ambiguities are resolved.

While Frazier argued that the sentence processor does not have access to subcategorization information while making the immediate decisions, Tanenhaus and Carlson (1989) showed using filler-gap sentences with long-distance dependencies that subcategorization (or some other equivalent argument structure) information is accessed immediately from the lexicon during sentence processing. A filler-gap sentence has a "filler" that must be associated with an empty category, or "gap," that occurs later in the sentence. For example, in sentence (17), the question phrase, "Which book", must be semantically interpreted as the direct object of "read."

(17) *Which book* did the little boy read __ in school?

They used such sentences in an embedded-anomaly technique to look for increases in processing load that result from a filler's being associated with a gap that results in an implausible interpretation. Their experiments have demonstrated the immediate use of argument-structure information in local ambiguity resolution. They have argued that there could be alternative explanations to the results of experiments by Frazier and colleagues.

Delay in Subcategorization Not all lexical syntactic ambiguities other than main categorial ambiguities are processed by making immediate decisions. To see why, let us look at an example. We have seen that a sentence similar to (10) such as in (18a) leads to a garden path.

(18a) The reporter saw the woman was not very happy.

(18b) The student realized the answer was not clear.

(18c) The reporter saw the woman who came in was not very happy.

One explanation to why (18a) is a garden path is minimal attachment: the sentence processor prefers the simpler main-clause interpretation to start with and has to change its interpretation later on.

However, (18b) is not so much of a garden path. Minimal attachment would predict difficulty in this sentence as well. In their experiments, Holmes, Stowe and Cupples (1989) looked at sentence pairs such as (18a and 18b) where one is a garden path and the other has a semantic bias that prevents a syntactic garden path. They found that people in fact take longer (than for versions of the sentences having the disambiguating grammatical marker *that*) to read the disambiguating parts in sentences such as (18a) but not in sentences such as (18b) where there is a semantic bias toward the sentential complement reading. For instance, in (18a), *saw* accepts *the woman* as a direct object whereas in (18b), *realized* requires a concept and prefers the complement rather than a direct NP. This shows that semantic information is used immediately in resolving subcategorization ambiguities in syntax.

However, this result is more than yet another piece of evidence for the immediate semantic decision hypothesis. When they varied the processing load on the subjects by increasing the length of the sentences, minimal attachment seemed to be the right explanation for the behavior they observed. People had difficulty with the more complex syntactic structure (the complement structure as in (18b)) even when there was a semantic bias in favor of it (as in 18b) [Stowe, 1991]. For instance, in (18c), the sentence in (18a) has been made longer by introducing the relative clause *who came in*. People seemed to have trouble with the more complex syntactic structure even when there was a semantic bias towards it; they went ahead with the minimal syntactic structure and had to make repairs upon reading the disambiguating items.

To explain these results, Stowe [Stowe, 1991] hypothesized that the absence of garden-path behavior when there is semantic bias towards the more complex structure may not be because the semantically preferred more complex structure had been chosen as proposed by Crain and Steedman (1985) and others. It may be because the decision has been delayed and neither structure has been chosen to the exclusion of the other. However, since the sentence processor has finite processing resources (such as working memory, for instance), there is a limit on the delay. According to what Stowe called the Limited Delayed Decision hypothesis, the sentence processor delays a decision when there is a conflict between syntactic and semantic preferences (as in (18b)) and pursues multiple interpretations in parallel until it runs out of some resource. At this point, when it can no longer afford to maintain parallel interpretations, it is forced to make a decision which it does by syntactic criteria such as minimal attachment and not by contextual bias.

According to this hypothesis, in (18a) both syntactic and semantic preferences pointed toward the simpler direct-object structure which was selected resulting in the garden-path effect later on. In (18b), syntax and semantics were in opposition and the decision was delayed until seeing the disambiguating region *was not clear*. At this point, both structures were available and the complement structure was chosen since the other one was syntactically infeasible at this point. In (18d), however, the sentence processor might reach its resource limits and select the direct-object structure with a consequent difficulty in the disambiguating part of the sentence.

(18d) The student realized the answer which was taught was not clear.[9]

Though it is possible that further data might negate the Limited Delayed Decision Hypothesis, and though there may be other models that can explain such complex behavior, a unified process model explains such behavior. The single process can make the kinds of delayed decisions seen above.

3.4 Summary

In summary, current psychological evidence tells us that the sentence processor has these characteristics:

- It has functionally independent faculties.
- The different faculties interact at every opportunity.
- Preferences from different faculties for alternative interpretations are integrated to select one that is best overall.
- It delays disambiguation decisions when information from different faculties are in conflict or when there is insufficient information for making a decision; the sentence processor then pursues multiple interpretations in parallel.

[9]This sentence was not taken from a published experiment and may not be a good example.

- The limited resources that the sentence processor has at its disposal place limits on possible delay.
- Preferences and constraints come from the goal to arrive at a completed and integrated interpretation of the input text and from the semantic and conceptual features of the world.
- The sentence processor can often recover from erroneous decisions in resolving different ambiguities.

4 Sentence Processing Models

There have been many computational architectures proposed in models of sentence understanding. In order to be able to compare them and to see the need for the architecture we are proposing, let us try to map the different dimensions along which architectures of sentence processors could differ. The architecture of a sentence processor has the following **architectural elements:**

- **Knowledge Sources:** One or more representations of different kinds of knowledge including those built while processing sentences.
- **Processes:** One or more processes where a process is a set of operators on different kinds of knowledge which are embedded in a control structure. Control structures are made of control elements such as sequence, recursion, coroutines, and so on.

Note that we do not talk about *processors.* A processor is an element of the implementation, not an architectural element [Marr, 1982 (see Chapter 1: The Philosophy and the Approach)]. However, we do refer to the entire architecture as the "language processor." Architectures connect their architectural elements together in various ways such as in series, in parallel, and so on.

4.1 Architectural Dimensions

One architecture can differ from another in the following features:

- **Language faculties:** A language faculty is a combination of one or more knowledge sources and processes that accomplishes a particular subtask of the language understanding task. For example, lexical processing, syntax, and semantics are distinct language faculties in most theories. The term *levels* is sometimes used to refer to the same elements as faculties. Note however that levels usually refer to stages while faculties are not the same as stages. There is no implication of sequentiality of the different faculties. Sequential models place the faculties in a sequence of modules; parallel models place them in parallel; some decompose the task of language processing *orthogonally* along each faculty. Integrated models integrate all faculties into one module. Thus models vary in decomposing the language processing task into various faculties.
- **Nature of decomposition:** The overall task of sentence understanding can be decomposed either along the lines of language faculties such as syntax, semantics, and conceptual analysis, or orthogonally into different processes running in each of the faculties. As an example of an orthogonal decomposition, one can divide language understanding into access, integration, and selection along each of the faculties.
- **Configuration:** The units obtained by either kind of decomposition can possibly be arranged in one of three configurations: in a *sequence* of *stages* as in sequential models, in a single module as in integrated models, or else in *parallel* as in the case of certain interactive models.
- **Nature of interaction:** Interaction between the units of an architecture can happen either through a shared representation (a knowledge source) that is understood by all units that are participating in the interaction, or it can happen between the control structures of different processes by transfer of control or by exchange of messages which may have to be translated if there is no shared representation.

- **Modularity:** When a task is decomposed, the resulting units can not only be arranged in different ways, they can also interact with each other in various ways. The amount and nature of interaction between the different modules can be described in terms of different kinds of modularity [Fodor, 1987; Marslen-Wilson and Tyler, 1987; Tanenhaus et al., 1987].

  - **Representational modularity,** also known as **information encapsulation,** means that each module has its own exclusive representation of a kind of knowledge that is not accessible to any other module. Knowledge sources are exclusive to processes. It is a strong form of modularity where interaction could not happen through shared representations.
  - **Process modularity** means that each module has its own process whose control structure does not interact with any other process. The only form of interaction possible is through a common representation such as a blackboard.
  - **Functional modularity** means that the modules can interact in any way as long as they are functionally independent, that is, one can perform its function if the other fails to run. Parts of a module may be functionally dependent on other parts of the same module, but are independent of other modules. Functional modularity permits interaction either through common representations (such as in a blackboard architecture) or message passing between the control structures of the modules. Functional modularity permits considerable freedom with respect to the kind of interactions possible.

  Thus, a *module* is an exclusive knowledge source plus a process in the first kind of modularity, an isolated process in the second notion of modularity, and does not have a well-defined meaning in a functionally independent architecture. For instance, if there are independent representations of knowledge sources with a single unified process operating on them all, there is neither just one module nor many different modules. Functional independence does not preclude sharing of knowledge sources or processes, as long as it does not render the accomplishment of one function inevitably dependent on another.

- **Grain size or intimacy of interaction:** To say that two modules interact during processing does not carry much meaning until we specify the grain size of the interval of interaction. For instance, if syntax and semantics interact only at the end of a sentence, that is not interactive. On the other hand, if they interact after each word, that would make a highly interactive parallel model. It is important to note that an architecture can be described differently at different grain sizes. For instance, what looks like a parallel architecture at the size of a clause (because of interaction going on at a grain size finer than a clause) might actually be sequential at a smaller grain size. This is particularly true in the case of the interaction between syntax and semantics. The lexical entry of a word has to be accessed before any significant syntactic or semantic analyses can begin. Similarly, syntactic processing of a word resulting in its binding to a higher structure has to happen at least partially before any semantic processing can begin. If not, valuable information that syntax can provide to semantics would be ignored by semantic processing which could be wasteful. However, not all syntactic processing for a word need be done before semantics can take off. We will see this in greater detail when we see how our model works. In short, it is sometimes desirable to have sequentiality at a fine grain size but independence and parallelism at coarser grain sizes.

There have been many computational models proposed to account for human language understanding capabilities. Some sentence processing models are good at explaining certain kinds of behavior more than others. For instance, some models assume autonomy of the syntactic level and hence cannot explain interactive behaviors. Some models explain only some subtask of the task of sentence understanding by focusing on only one aspect of language processing. For instance, there are models of lexical ambiguity resolution, of syntactic parsing, of pronoun reference in relative clauses, and so on. In this work we are trying to build a complete model of sentence processing, one that analyzes the lexical information, syntax, and semantics, of a sentence to arrive at the best interpretation possible.[10]

The role of a computational model is much more than just producing the same behaviors that people do and showing the computational feasibility of a theory. A model is supposed to *explain* computationally how the sentence processor could arrive at the interpretation that it does. A model could point to parts of a behavior that had not been noted in human performance. A model might lead to the generation of hypotheses about particular details of the sentence processor's architecture which could be used to design further experiments.

4.2 Sequential Models

Models of sentence understanding can be classified as sequential, integrated, or interactive models depending on how knowledge representations and processing at different levels relate to each other. In sequential models, a lower level does not get any feedback from a higher level. Each level receives the output of processing at the previous lower level and sends out its output to its next higher level. Traditionally, the task of language understanding has been decomposed into the subtasks of syntax, semantics, and pragmatics which are arranged in a syntax-first sequential architecture as shown in Figure 9. A typical example of a sequential model is Forster's Levels of Processing Model (1979) shown in Figure 10. Such a sequential model has the advantage of accounting for the fast, autonomous processing at syntactic and other earlier stages. An interestingly different sequential model, with semantics preceding syntax, was proposed by Lytinen (1986) and implemented in the MOPTRANS program.

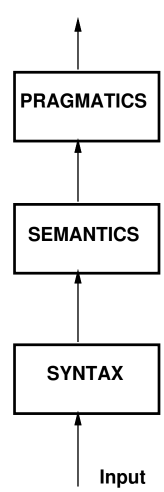

Figure 9: Sequential Model.

[10]However, we do ignore phonology, morphology, and so on; also, there is no model of pragmatic or discourse processing at the present time.

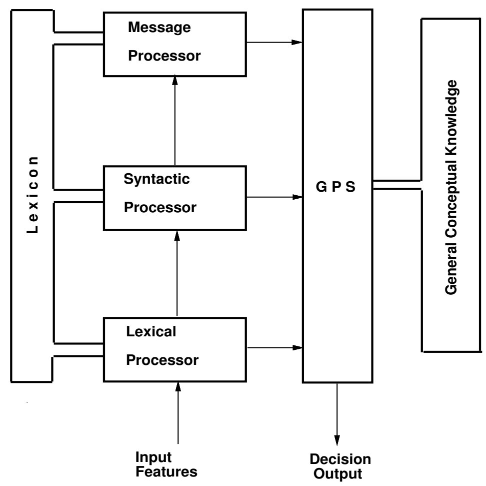

Figure 10: Forster's Levels of Processing Model.

4.3 Integrated Models

The Integrated Processing Hypothesis [Birnbaum, 1986; Birnbaum, 1989; Schank et al., 1980] states that the language processor applies syntactic, semantic, and other kinds of knowledge at the earliest opportunity in processing a piece of text. Thus, it is essentially a different expression of the same hypothesis that Stowe called the Immediate Semantic Decision Hypothesis. However, integrated models of language understanding (Figure 11) assume more than just this; they have integrated representations of knowledge, not just the integration of the information provided by the representations during processing. They do not retain independence in the use of the different kinds of knowledge involved in linguistic competence. As a result, integrated models do not account for functionally independent behaviors in language understanding. Examples of integrated models include IPP [Lebowitz, 1983], BORIS [Lehnert, Dyer, Johnson, Yang, and Harley, 1983], DMAP [Riesbeck and Martin, 1986a; Riesbeck and Martin 1986b], and AQUA [Ram, 1989]. . Many other models such as Jurafsky's (1991) also have integrated representations with all the different kinds of knowledge represented in a monolithic knowledge base.

4.4 Interactive Models

Models which retain independent representations of different kinds of knowledge support online interaction between different faculties without integrating them so much so that they are not independent anymore to any degree. Integration of information happens during processing a text. Interactive models can adhere to the integrated processing hypothesis without having to sacrifice the independence between the use of different kinds of knowledge.

4.4.1 Stowe's Model

An example of an interactive model was proposed by Stowe. Stowe's Limited Delayed Decision model, though not a computational model, had syntax and semantics running in parallel. Any decision was made by taking syntactic and semantic preferences into account. When the two preferences were in conflict, the parser would delay the decision and pursue multiple interpretations.

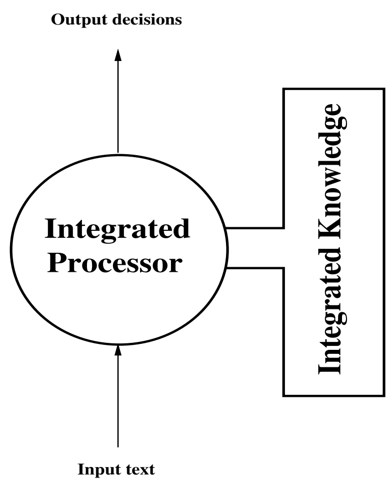

Figure 11: Integrated Model.

Such a delay was limited by resource limitations. At the limit, the sentence processor always made a decision by syntactic criteria alone. Stowe's model (1991) has the following interesting features:

- It made immediate decision taking both syntactic and semantic preferences into account when there was no conflict between them.
- The basis for syntactic preferences was the need to complete an incomplete structure.
- Decisions were delayed if the syntactic default was in conflict with the semantic bias.
- The delay was limited by resource constraints.
- Selection at the limit of delay was determined by syntactic preferences alone.

4.4.2 Orthogonal task decomposition.

Apart from the traditional sequential decomposition and the no-decomposition architecture of the integrated models, one can decompose the task of understanding a sentence orthogonally into the following subtasks:

- Access lexical entries for a word;
- Propose feasible ways of integrating the word into current interpretations;
- Select the most preferred interpretation(s).

Such a decomposition was used by Jurafsky who called the three steps above *access, integration,* and *selection* [Jurafsky, 1991; Jurafsky, 1992]. Jurafsky's model was also an integrated model since all types of knowledge were represented in integrated units called grammatical constructions. Models with orthogonal decompositions could differ from one another in their architectural details such as whether they propose all possible interpretations and then select them, or they propose and select one at a time, and so on. For instance, the NL-SOAR model [Lehman, Lewis, and Newell, 1991; Lewis, 1992] can only pursue one interpretation of a sentence at a time. Another interesting feature of the NL-SOAR model is that it loses independence between its knowledge sources gradually as the model chunks them together into monolithic units.

4.4.3 The ATLAST model

A model which used this decomposition, though it did not actually model all levels of sentence processing, is the ATLAST model [Eiselt, 1985; Eiselt, 1989]. ATLAST was a model of unified lexical (semantic) and pragmatic ambiguity resolution and error recovery. Syntactic knowledge was assumed to be processed separately using an augmented transition network (ATN) parser. Syntactic structure assignments within individual phrases were made independently without any interaction with semantic information by a module called Capsulizer. However, inter-phrasal syntactic decisions were made using semantic information by the Filter module. Semantic interpretations were proposed by the Proposer module using a marker passing spreading activation mechanism. The architecture of the model is shown in Figure 12. The model achieved disambiguation using multiple access of meanings for lexical items and pragmatic situations, choosing the meaning that matched previous context.

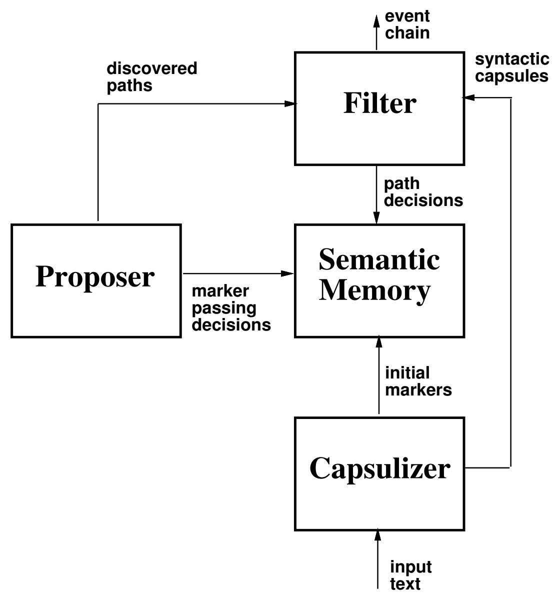

Figure 12: The ATLAST Model.

ATLAST was not intended to be a model of syntactic ambiguity resolution. It had separate processes for syntax and semantics. There was no interaction of semantic information in syntactic processing. Nor was initial semantic analysis (by the marker passing mechanism) constrained by syntactic decisions. Since the syntactic parser was an ATN, it could deal with only one syntactic structure at a time and would have to backtrack often if it were to be extended to more complex and interesting syntactic constructs such as relative clauses. Since semantic processing was a different kind of process that maintained multiple interpretations in parallel, there was no straightforward way to extend the model to address structural ambiguity resolution as well other than unifying the two processes into a single process. That would of course imply replacing the ATN parser with one that could go hand-in-hand with semantic processing without unwarranted backtracking in addition to converting ATLAST's syntactic knowledge (which was represented as a set of rules) into a form common with its semantic and pragmatic knowledge.

4.5 Unified Process Architecture

Having looked at a variety of sentence processing architectures, let us see why we are hypothesizing a unified process for resolving structural and lexical ambiguities and for recovering from possible errors. We hypothesize that there is a unified process in sentence processing that resolves ambiguities and attempts to recover from its errors at all levels of language processing. By a unified process we mean that the same kinds of operations are applied to different kinds of knowledge and that the control structure in which the operations are applied is the same for different levels such as syntax and semantics. We will develop the architecture of a unified process sentence understander in the next section.

4.5.1 Unified Process: Why?

Why do we need a single unified process for all faculties of sentence understanding instead of multiple interacting processes for the different faculties? We know that there are different kinds of knowledge used in language understanding. In fact, we defined language faculties based on the use of different kinds of knowledge. We do not want to integrate all the sources of knowledge; we want to keep their use independent of each other in order to be able to support functionally independent behavior. Nor can we afford completely modular processes for each faculty since we want to support interactive behavior; processing in different faculties must interact closely and quickly.

Given these requirements, we can either hypothesize separate processes for each faculty and then try to find ways of making them interact or we can start with a single process and then try to find where if at all syntactic and semantic processing must differ because of the use of different kinds of knowledge. We have taken, for several important reasons, the latter approach of hypothesizing a single process and then trying to identify small differences in the ways by which different faculties use their knowledge to come up with their proposals for attachment and their preferences. We are trying to put language processing together using as few assumptions of separate processes as possible while keeping in mind the significant claims of modular theories.

An obvious reason is parsimony. Why start with many different processes if we can do with just one? A more important reason is the evidence from Stowe's experiments which showed that people's ambiguity resolution and error recovery behavior in syntactic ambiguities is much the same as what ATLAST [Eiselt, 1989] demonstrated for lexical and pragmatic ambiguities. Given that these ambiguities require the use of different kinds of knowledge to be resolved, we have some evidence for the existence of the same kind of ambiguity resolution and error recovery happening in different faculties. Since the overall process of sentence understanding is all about resolving ambiguities by selecting the best from a set of interpretations and making amends for any erroneous decisions along the way, it is reasonable to propose a single unified process that carries out the overall job of sentence understanding. The differences may lie in the operations used on the different kinds of knowledge that result in the proposed interpretations and the differential preferences for them.

Stowe's work (1991) showed that syntactic analysis as it happens in human sentence processing is itself pretty much the same process as the semantic analysis process such as the one proposed in ATLAST [Eiselt, 1989]. For instance, it accesses multiple structures in parallel, can pursue multiple interpretations in parallel, can switch from one interpretation to another, and interacts with other faculties of language processing.

A third possible hypothesis is that there are separate processes for different faculties, but there is unrestricted interaction between them. Such a hypothesis can be found in certain connectionist systems [Waltz and Pollack, 1985].[11] However, uncontrolled interaction leads to difficulties in

[11] One might argue that connectionist systems with many interacting networks have the same process of spreading

accounting for the kind of limited delay behavior seen in human sentence processing [Stowe, 1991]. The sentence processor needs a control structure monitoring the interaction between the different faculties.

activation running in each network and not separate processes as described here. However, in a connectionist system, the control structure of a process is translated to the configuration of the network itself; the nodes and links in the network along with their weights represent both the knowledge and the control structure of the computational process. In that sense, all connectionist networks of the same kind (such as backpropagation networks or recurrent networks) have the same (spreading activation) process.

5 Unified Computational Model: COMPERE

In the last section we saw several models of sentence processing with a variety of architectures from the sequential architecture of Forster's model to the integrated architecture of IPP to the parallel architecture of ATLAST [Eiselt, 1989; Forster, 1979; Lebowitz, 1983]. Before looking at our model of sentence understanding, let us take a moment to see the set of assumptions on which COMPERE is based.

5.1 Assumptions

As any exploratory theory, COMPERE has been designed by assuming several things. Some of our assumptions are made so commonly in language processing research or are so intuitively obvious that they rarely get mentioned explicitly. Below is a perhaps incomplete set of assumptions we have made:

1. Human language understanding can be studied independently of other cognitive faculties, even including language generation. This is a convenient assumption made almost always in language processing.[12]

2. We can build a model of adult language understanding without accounting for developmental language acquisition.

3. We can ignore phonology, morphology, other speech processing issues, stress and accent, diagram understanding, and so on, and still make important progress in the field of natural language understanding.

4. We can study just one natural language and in particular, English. Though English is very different from many languages for instance in its relentless reliance on word order information [Bates et al., 1991], we believe that the overall process and architecture of language processing is the same across languages. Differences, if any, would be in the relative importance of sources of information and in the strategies used for extracting the information from linguistic markings in the text. Though we are not doing any significant cross-linguistic studies, we believe that the flexible and general architecture of our model would accommodate the structures and processes of other languages quite well.

5. Sentence understanding is a valid subtask of language processing. We will however attempt to take the effect of the context of a sentence into account. At some point we will also extend the model to understand more than single sentences in isolation.

6. Sentence understanding happens the same way no matter what overall goal is being pursued. This, perhaps, is one of the most significant assumptions we are making, though it is not a very uncommon one. What it says is that the methods that are applied and the way the task of sentence processing is carried out is invariant over the kinds of higher-level language processing goals, such as whether the text is being processed to acquire knowledge, or for pleasure, or to answer questions, and so on. Certain differences in sentence processing do exist between language processing for different goals. For instance, one might skim through the text quickly or read in depth to various degrees. However, we are working with the assumption that the architecture of the sentence processor can be determined by considering

[12] However, we are investigating the relationships between language processing and design problem solving in a separate project.

only the routine behavior in sentence processing. This in fact is the reason we are dealing with architectural theories as opposed to functional theories. A functional theory would be useful to develop as an abstract theory if there were different ways of decomposing the task and different methods of solving them depending on the overall reasoning task. In the case of sentence processing with the above assumption, human behavior suggests that only one kind of decomposition of the task with only one kind of architecture (the one with parallel, interactive yet independent modules) is feasible. A theory of sentence understanding which is merely functional and which does not make architectural commitments is thus not as interesting or useful.

There is an important piece of evidence supporting this assumption. It is that you can not turn off processing at any level without doing neurological damage to the human subject. Syntax, for instance, gets processed all the time no matter whether the current task (such as answering a particular question) requires it or not. There is no choice as to the kinds of processing the sentence processor can do at any time; it always has to do all the kinds of processing shown in our model.

7. People normally read from left to right and do not reprocess surface-level representations repeatedly by eye movement unless they are stuck in a garden path. This has been supported by experimental studies such as reading time and eye-movement studies [Carpenter and Daneman, 1981; Carpenter and Just, 1988; Frazier and Rayner, 1982; Rayner, 1978]. We also assume that people do not reprocess surface structures internally in some kind of buffer which would store parts of the sentence they are currently reading.

8. We do not need structural transformations in syntax. We are assuming, as others have done before [e.g., Jurafsky, 1991], that constructs such as questions for which transformations have been proposed can be handled in semantics without having to do transformations in syntax.

9. The language processor has a universal goal to resolve ambiguities by choosing one from a set of possible interpretations.

10. The language processor also has a universal goal to complete an incomplete entity at any level of description. For instance, it has a goal to complete a syntactic construct such as a phrase or a clause that is currently incomplete. The completion goal is the source of expectations which provide top-down guidance to the process.

5.2 The Sentence Understanding Task

Before we can look at the architecture of our model, we need to specify the task to be performed in greater detail. The task being modeled can be described in terms of its input and output as follows. The model accepts a sentence in the form of a left to right sequence of words. It does not process any punctuation marks at this time. The sentences have to be in the declarative mood. It does not process questions. The sentences can have prepositional phrases, relative clauses, and so on.

The output of the program has two parts:

- One or more parse trees representing the surface structure of the sentence. The program does not do syntactic transformations.

- A graph of the events, objects, and states in the sentence along with the thematic roles that relate them. For this purpose, we have adapted a set of thematic roles or cases listed in the classic textbook on Natural Language Understanding by James Allen [Allen, 1987]. They are:

1.	Agent	2.	Theme
3.	Experiencer	4.	Beneficiary
5.	Instrument	6.	At-Location
7.	To-Location	8.	From-Location
9.	Co-Agent	10.	Co-Theme

Objects will be distinguished from one another by a very simple discrimination tree of object categories based on features of objects such as animacy. The model will have a small lexicon sufficient to demonstrate the feasibility of the aforementioned claims through a small set of sentences. No attempt will be made even to represent the semantics of all the closed-class elements in English. For instance, the program will not have representations of the different meanings possible of all the 45 or so prepositions in English.

Thematic roles alone do not of course completely capture the semantics of a natural language. We plan to extend the semantics in COMPERE to include other aspects of linguistic semantics such as time and tense, space, modification, and so on [Frawley, 1992].

5.3 The unified model: COMPERE

Our model of sentence understanding called COMPERE (Cognitive Model of Parsing and Error Recovery)[13] (Figure 13) [Eiselt et al., To appear; Holbrook et al., 1992] has a single unified process operating on independent sources of syntactic and semantic knowledge. This is made possible by a uniform representation of both kinds of knowledge in the same form. The unified process applies the same operations to the different kinds of knowledge and has a single control structure which performs the operations on syntactic and semantic knowledge in tandem permitting a rich interaction between the two, both through transfer of control and through a shared representation of the interpretations of the input text being built by the unified process.

5.4 Knowledge Representation

Syntactic and semantic knowledge are represented in separate networks in which each node is a structured representation of all the information pertaining to a syntactic or semantic category or concept. A link, represented as a slot-filler pair in the node, specifies a parent category or concept of which the node can be a part, together with the conditions under which it can be bound to the parent, and the expectations that are certain to be fulfilled should the node be bound to the said parent. In addition, nodes in either network are linked to corresponding nodes in the other network so that the unified process can build online interpretations of the input sentence in which each syntactic unit has a corresponding representation of its thematic role and its meaning and vice versa. COMPERE also has conceptual knowledge of objects and events in the world and the relationships between them. Semantic role knowledge acts as a bridge between syntactic and conceptual knowledge forming a continuum of kinds of knowledge from the purely syntactic knowledge of the surface structure of the language to deep conceptual knowledge of the world. In addition, there is a lexicon as well as certain other minor heuristic and control knowledge that is part of the process. Before we take a look at COMPERE's unified process, let us see the different kinds of knowledge and their representation.

[13] A *Compere* is one who introduces and interlinks items of an entertainment.
— Chambers Twentieth Century Dictionary.

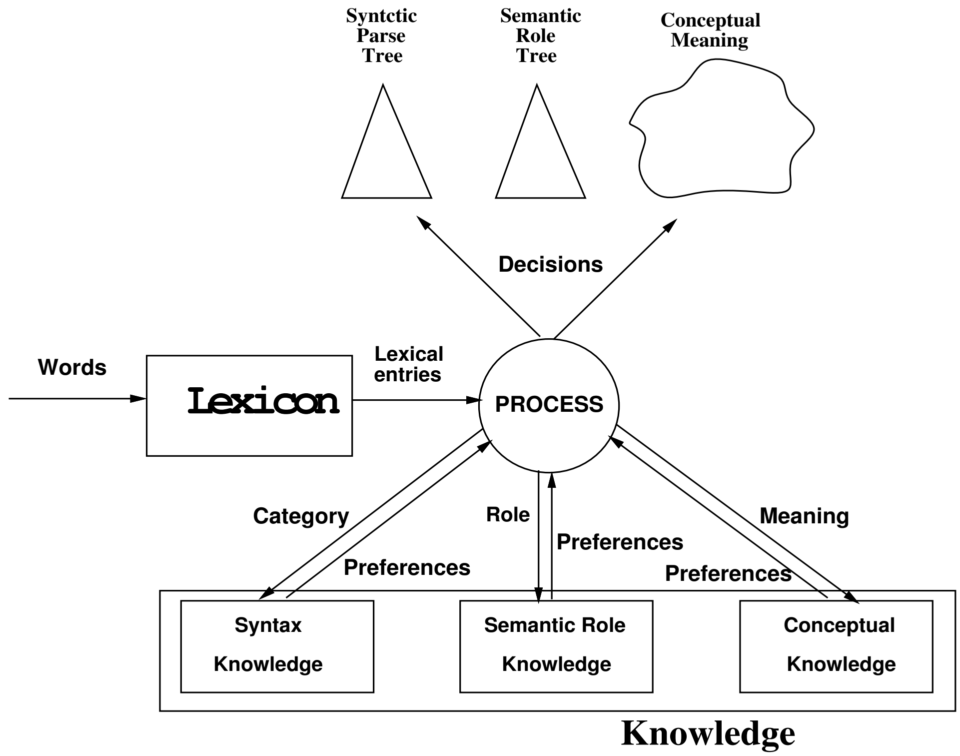

Figure 13: Architecture of COMPERE

5.4.1 Syntactic Knowledge

Syntactic knowledge, a grammar, is represented as a network in which each node holds all the knowledge about a particular syntactic category necessary for parsing a sentence into its surface structure. The representation of the grammar is similar to Categorial Grammars [Steedman, 1987; Steedman, 1989]. Since a rule in a grammar has to be able to express a relationship between three or more constituents, grammatical relationships can not be expressed as binary relationships. As such, syntactic knowledge is not amenable to semantic network representations without significant enhancements to them. For instance, a phrase-structure rule such as

```
S --> NP VP
```

represents a relationship between S, NP, and VP. This can not be broken down to binary relationships since syntactic relationships aren't mere set memberships; other relationships such as word order are important in syntax.

A small subset of COMPERE's syntactic network is shown in Figure 14.

Each link in the syntax network can be binary, ternary, or higher order. For illustration, we will use only ternary links. There are two kinds of these links. The first kind is an *Expectation Link.* It connects three nodes N1, N2, and N3, and says that a constituent of category N1 can be part of a constituent of category N2 iff it is followed by one of category N3. For instance, consider the node

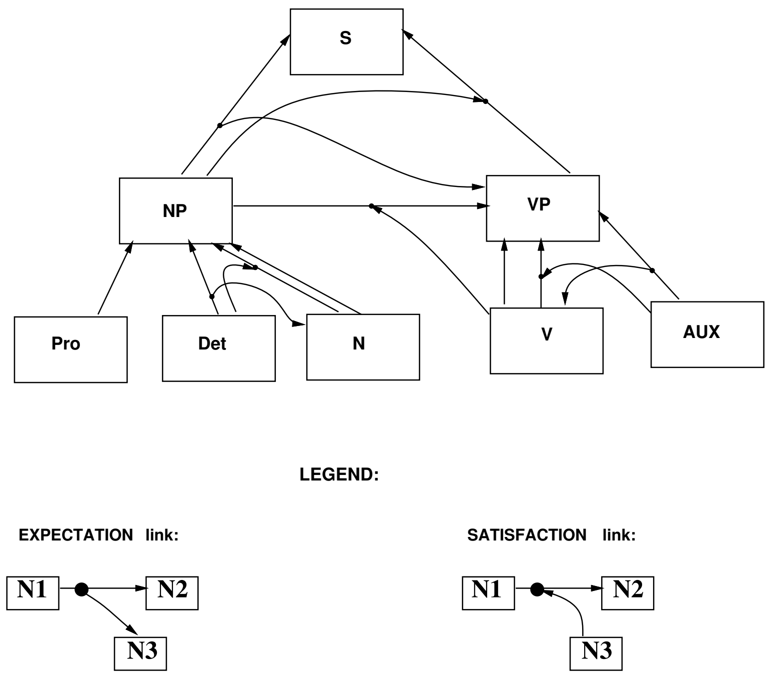

Figure 14: Syntactic Knowledge of COMPERE

in the network representing a part of COMPERE's knowledge of the way a noun phrase (NP) can combine with other syntactic structures. Such a node (in a highly simplified grammar) looks like:

```
NP:
  S:/VP
 VP:\V
 PP:\Prep
```

This NP node says among other things that a NP can be part of a sentence (S) if it is followed by a VP. In this case the language processor *expects* a verb phrase (VP) to follow the NP since it is required to complete the S structure. Thus, we can define syntactic expectation as follows.

> **Syntactic Expectation:** A syntactic constituent is expected at a particular point if and only if the constituent is necessary to complete the currently incomplete structure.

The above NP node also says that an NP can be part of a VP is preceded by a verb (V), and part of a prepositional phrase (PP) if preceded by a preposition (Prep). Optional constituents are not *expected.* For instance, since we can have rules such as

```
S --> NP VP
S --> NP VP PP
```

a VP is expected after seeing an NP in a S but a PP is not. The VP is essential; the PP is optional.

Satisfaction Link: The second kind of link is a Satisfaction Link, so called because when it is processed successfully, a previously made expectation may be satisfied by the addition of a new constituent. A ternary satisfaction link connects three nodes N1, N2, and N3, and says that an N1 can be a part of an N2 iff it is preceded by an N3. For instance, in the representation of the NP shown above, the NP can be a part of a VP if it is preceded by a V, or part of a PP if preceded by a Prep (preposition). The Prep must be an immediately preceding sibling of the NP in the PP. We can thus define contiguity as follows.

> **Syntactic Contiguity:** Two nodes N1 and N2 are said to have a contiguous span if the first word in the part of the sentence spanned by N2 follows the last word spanned by N1 immediately to the right.

If N1 is a child of N2 in a parse tree, then a new node N3 can be added as the next child of N2 only if N1 and N2 are contiguous in the sentence. A higher-order link would specify either more than one expected item or more than one item required to precede the current one.

Bottom-Up Representation of Syntactic Knowledge: This representation of grammar is suitable for a bottom-up method of building the syntactic structure of a sentence. However, the notion of expectation defined above results in augmenting the bottom-up process with top-down preferences. A syntactic structure is expected only when it is required to complete the current construct (and therefore the sentence structure S.) Optional constituents of a construct are not expected; they are represented instead in the optional construct itself. Hence we call these representations **bottom-up representations.** For instance, our knowledge that a PP can be added to an NP is represented in the node for PP (not the one for NP) thus:

```
PP:
 NP:\NP
```

This means that a PP can be a part of an NP when it is preceded by the NP, that is, when the NP is already a complete NP.

A grammar for a natural language can be represented in different ways. When represented simply as a set of rewrite rules, a grammar is suitable for a top-down analysis as done by an augmented transition network (ATN) [Wanner and Maratsos, 1978; Woods, 1970; Woods, 1973]. However, top-down processing forces a sentence processor to make unnecessary commitments along the way with a consequent problem of frequent and wasteful backtracking that does not happen in human sentence processing behavior. One can perform a purely bottom-up processing with a grammar represented as a set of rules. However, this is not efficient since it would make commitments much later than it could; a bottom-up parser such as a chart parser [Kay, 1973] does not commit to a structure until it has seen every part of it. Such a parser would not be an online model of parsing. One could exploit the requirements specified by a grammar to identify the necessary parts of a structure which can be expected to be seen in following text. This would enable the sentence processor to make early commitments when possible as seen in human sentence processing behavior.

In order to make early commitments in bottom-up parsing, the language processor has to access all the pieces of syntactic knowledge that specify structures of which the current category can be a part of. In phrase-structure rules, one has to search through all the rules to extract such information. Phrase-structure rules are inherently top-down representations based on the left-to-right ordering of the constituents of a structure. In order to extract the same information efficiently without making unwieldy sequential searches through the set of rules repeatedly, the grammar has to index the rules so that the rules that are applicable in a context are available right away to the processor. Moreover, since we want to avoid incorrect early commitments and at the same time make correct commitments (based on all the knowledge available at that point) as soon as we can, processing has to make use of a mix of bottom-up and top-down constraints. In order to do that, the representation of the grammar itself has to be bottom up and not just top down. That is, a higher-level structure should not specify its optional children (or at least distinguish optional children from those necessary for completing the structure); rather, a lower-level structure should say what higher-level structures it can be a part of, either as a requirement of the higher-level structure or as an optional part of it.

Such a representation is a set of higher-order relationships between the syntactic categories in a language. However, it differs from phrase-structure rules in that it makes bottom-up information explicit instead of top-down information as in rules. We call such a representation a bottom-up representation. It is different from phrase-structure rules in that it has all the necessary grammatical information about a category in one node unlike phrase-structure rules where such information is spread over several rules which also have information about other categories. It is the addition of more structure and indexing that makes a set of rules a network representation of the same knowledge.

Syntactic Preferences: Syntactic preferences have been considered to result from structural criteria such as high or low attachment or minimality of the number of nodes in a structure [Frazier, 1987]. Thus the sentence processor is said to choose the minimal structure or one with right association and so on. Many recent psycholinguistic studies [Crain and Steedman, 1985; Pearlmutter and MacDonald, 1992; Spivey-Knowlton, 1992; Taraban and McClelland, 1988; Trueswell and Tanenhaus, 1992] have used various forms of contextual bias to demonstrate that the sentence processor does not have any pervasive preference for minimality of the size of a structure or for high or low attachment. They have been able to induce and avoid processing difficulties at the will of semantic context. This is not to say that the sentence processor does not have any kind of syntactic preference. Of course it does have preferences for one or another structure which is evident especially when semantics does not have any preference for either of the interpretations, as in (1). Officers might just as well teach or be taught.

(1) The officers taught at the academy were very demanding.

Stowe [Stowe, 1991] has proposed an alternative explanation for such syntactic preferences which I am attempting to generalize to language processing as a whole. The sentence processor has a pervasive goal to complete an incomplete item at any level of processing, as stated in the "Completion Goal" assumption earlier in this section. In syntax, it has a goal to complete the syntactic structure of a unit such as a phrase, clause, or a sentence. The sentence processor prefers the alternative which helps complete the current structure (called the Syntactic Default) over one that adds an optional constituent leaving the incompleteness intact. Such behavior can be described sometimes as a minimal attachment. For instance, in (1), a VP is required to complete the sentence after seeing *The officers*. Since the main-clause interpretation helps complete this requirement and the relative-clause interpretation does not, the main-clause structure gets selected as semantics did not have a bias towards either interpretation anyway at that point.

5.4.2 Role Knowledge

Role knowledge is a bridge filling the gap between structural syntactic knowledge and conceptual knowledge. There is a network of thematic roles which specify how roles combine to form other roles just as the syntactic network specified how syntactic categories combine to form other higher-level categories. A part of COMPERE's role knowledge is shown in Figure 15.

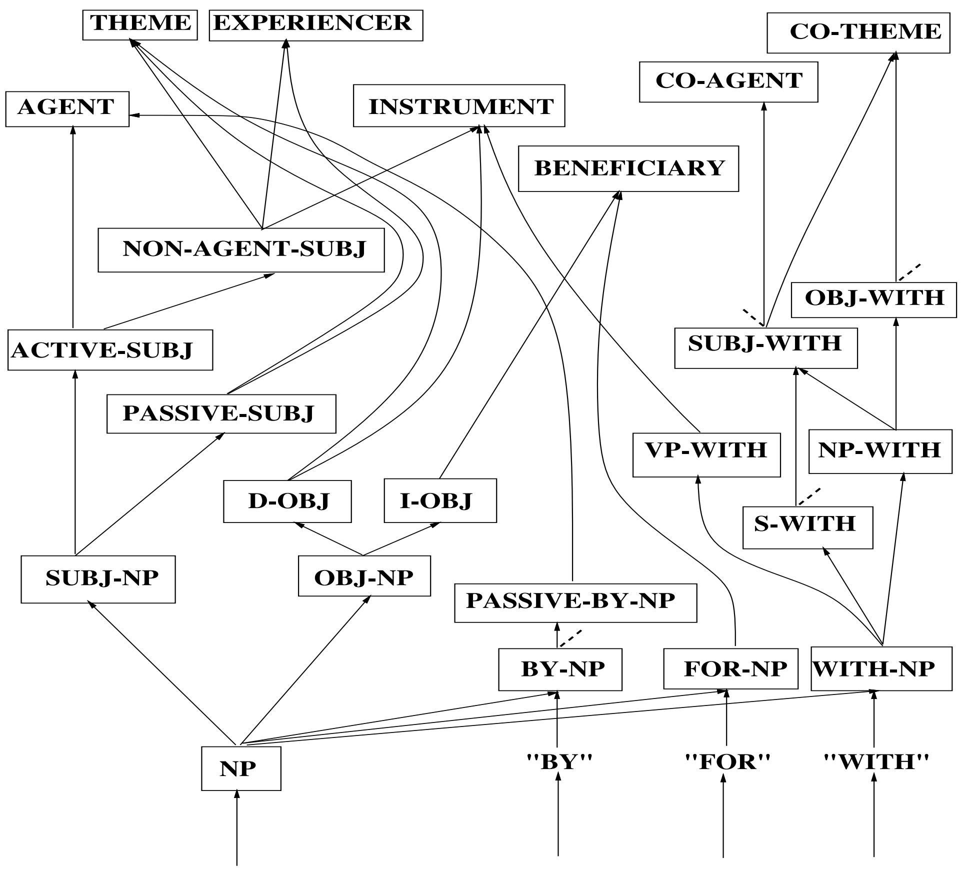

Figure 15: Role Knowledge of COMPERE

There are *primitive roles* which arise directly from syntactic categories or from closed class elements of the language such as prepositions. Primitive roles then evolve into bigger and more specialized roles as further processing of the sentence yields more and more information. For instance, a noun phrase has a primitive role called the **NP** role which can evolve *in context* to thematic roles such as the **AGENT, THEME,** or **EXPERIENCER** roles. These, in turn, become part of an event role.[14]

Between the primitive roles and the thematic roles, there are intermediate roles such as the ones shown in Figure 15. These intermediate roles are the results of a partial semantic analysis of a part of a sentence based on all the information available to the processor at that point. Further information provided by the rest of the sentence might further specialize these intermediate roles

[14]We treat events also as a kind of role since then we could extend the notion of thematic roles to conceptual relationships. An event can take a particular causal role, for instance, in a conceptual structure. However, such "conceptual roles" have not been implemented yet.

to arrive at a thematic role assignment.

Each link in this role network specifies not only that the role at its lower end can be part of the node at the upper end, but also a set of constraints to be satisfied for this role specialization to occur. These constraints employ other cues such as the voice of a sentence, or the argument structure of the verb (whether it is transitive or intransitive, etc.).[15] Semantic and conceptual knowledge join hands with syntactic knowledge to determine the course a role takes in its emergence through the role network. The different kinds of knowledge are used via the tests on different cues that must be performed before specializing a role on any link, just as the syntactic precedence and contiguity tests before making a syntactic binding between a child and a parent node. For instance, conceptual knowledge comes into play while testing whether an event affords a particular thematic role, and if so, whether the object that stands for the noun-phrase satisfies any conceptual constraints on that role. However, at this point COMPERE does not have an equivalent of syntactic expectations in semantic or conceptual knowledge.

Predicate Argument Structures We saw above that roles emerge from primitive roles which arise directly from syntactic information. However, there is an additional kind of knowledge involved in determining roles and also in determining permissible syntactic structures. It is the knowledge of what kinds of heads (usually verbs) take what arguments. It has been called variously as predicate argument structure or subcategorization structure [e.g., Tanenhaus and Carlson, 1989; Tanenhaus, Garnsey, and Boland, 1991]. It is this knowledge that helps the sentence processor make distinctions between, say, intransitive, transitive, bitransitive verbs, and verbs that are combinations of those.

At this time we do not have a clear idea of how to represent predicate argument structures. However, we believe that this kind of knowledge is also a part of the bridge between purely syntactic and purely conceptual knowledge. Roles emerge in context not just from syntactic categories but from a combination of category information with argument structure information in addition to other semantic and conceptual constraints.

Many people have proposed that this information is lexical [Abney, 1989; Fodor, 1978; Ford, Bresnan, and Kaplan, 1983; Holmes, 1987; Stowe, in press; Tanenhaus and Carlson, 1989]. That is, the lexical entry of a verb specifies what (syntactic) arguments it takes. There are good reasons however, for not making such knowledge a part of the lexicon. It is not parsimonious to begin with since some information is duplicated for many verbs of the same kind. Also, one can capture the ambiguities that verbs embody with respect to the arguments they accept and require, by representing such information in a more flexible and general form as we have been doing for other kinds of knowledge. More than all, there is evidence against such lexical generation of syntax [Frazier, 1989]. The sentence processor seems to be delaying decisions as we saw earlier when there are major categorial ambiguities such as a noun-verb ambiguity. It however makes immediate decisions without waiting for disambiguating information in subcategory ambiguities such as a transitive-intransitive ambiguity. Such distinctions between major category ambiguities and subcategory ambiguities show that differences exist in the availability of information required for resolving the two kinds of ambiguities. Major categorial ambiguities use information about the categories of a word which is stored in the lexicon. Subcategory ambiguity resolution must use

[15] English seems to make use of eight types of cues to determine the thematic roles of the noun phrases in a sentence. These are: word order, prepositions, voice, syntactic structural relationships (such as, does the PP modify a VP or an NP), argument structure, role structure of an event (what roles an event affords), role-filler preferences of an event (such as animacy), and semantic role-filler constraints for the role. In particular, agreement information such as agreement in tense, number, gender, and possessive and reflexive agreements are not important cues as far as thematic role assignment goes. Agreement cues might, on the other hand, play major roles in the phenomena of reference and anaphora.

knowledge of the subcategorization or argument structure of the verb which may not be lexically accessible.

Categorial Grammars The discussion of representing argument structures enables us to address the question of how our representation of syntactic knowledge differs from categorial grammars [Steedman, 1987; Steedman, 1989]. The most radical proposals of categorial grammars purport completely bottom-up processing mentioned earlier and hence can not account for a host of evidence from human sentence processing behavior. Other approaches to categorial grammars combine semantic information to various degrees in their categories. This not only takes away the independence between syntax and semantics, it also makes processing more lexically based. Such an account will have difficulties capturing the generalization observable in syntactic processing [Frazier, 1989] apart from not being parsimonious. Our representation tries to capture generalizations as far as possible by representing argument structures separately from the major syntactic categories and by hierarchicalizing the categorial information. For instance, our representation of a verb only tells us how it can be part of a VP but not how that VP can combine with an NP to form an S. By not precompiling such information we restore each information to that level in the hierarchy which enables us to capture the generalizations the most.

5.4.3 Conceptual Knowledge

COMPERE has a network of structured nodes representing its knowledge of objects and events in the world and relationships between them. A node in the conceptual network stands for a concept.[16] The structure of the node (i.e., its slots) represents the relationships between the node and other concepts. These relationships can specify thematic roles that the concept affords as well as world knowledge in the form of selectional preferences. For example, the event of seeing something is represented as:

```
SEE:
 role: EVENT
 agent: (must-be ANIMAL)
 theme: (must-be VISIBLE-OBJECT)
 instrument: (must-be OPTICAL-INSTRUMENT)
```

Objects are distinguished from one another by a simple object hierarchy such as the one shown in Figure 16. Nodes representing objects have the necessary information to constitute this object hierarchy as a part of the conceptual network. At this time, this is very incomplete and rather ad hoc.

5.4.4 Lexical Knowledge

In addition to structural and conceptual knowledge, the system has a lexicon which provides the syntactic categories and subcategories of words as well as the meanings of words represented as pointers to conceptual nodes in the network. An example of a lexical entry would be:

```
Saw:
 word: see
 category: V
```

[16] While we do not make any rigorous attempt to define commonly used terms such as object, event, and concept, here we intend concept to simply mean an object, an event, or a relationship.

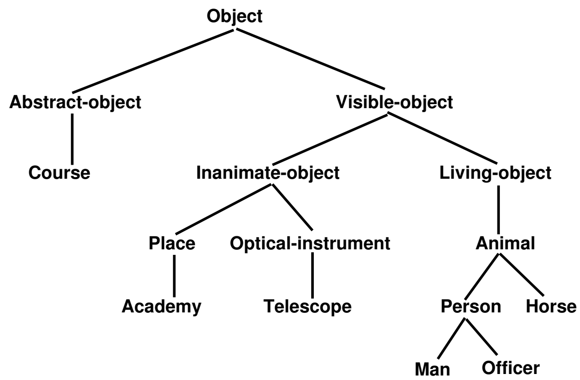

Figure 16: A Simple Object Hierarchy.

```
subcategory: ...
meaning: SEE
```

5.5 The Unified Process

COMPERE's unified process has a single control structure that simulates the parallelism involved in processing the input at the different levels of sentence processing using the several kinds of knowledge we saw above. Though we are constrained by the single processor architectures of today's computers, we can visualize COMPERE running on a parallel machine in which each node in the representation does some processing based on the knowledge available at that node and perhaps its neighbors. Most decisions are made locally in a node in COMPERE without having to search large networks of nodes.

The unified process can be described as having the following control structure for its operations. This is just an overview of the process with many important details such as error recovery mechanisms left out to be explained later. The control structure is also shown in Figure 17.

- **Lexical Access:** The process reads the words in a sentence from left to right and accesses their lexical entries. It then passes the syntactic categories of the word to the syntactic knowledge representation. At the same time, it also passes pointers to the meaning(s) of the word to the conceptual representation. This is a multiple-access model which is supported by a large body of experimental evidence [Seidenberg et al., 1982].

- **Instantiation:** The processor builds instances of nodes for the syntactic category of the node, the meanings of the word, and the primitive roles if any suggested by the syntactic categories. If the word is lexically ambiguous, multiple instances of any or all of the above may be instantiated depending on the kind of ambiguity.

- **Proposing Feasible Attachments:** The process next proposes ways of attaching the current constituent to the structures built so far for the sentence that are feasible in the current syntactic and semantic context. Such feasibilities are proposed independently from each kind of knowledge at the corresponding level. These proposed attachments come with degrees of preference expressed in numerical (integer) values.

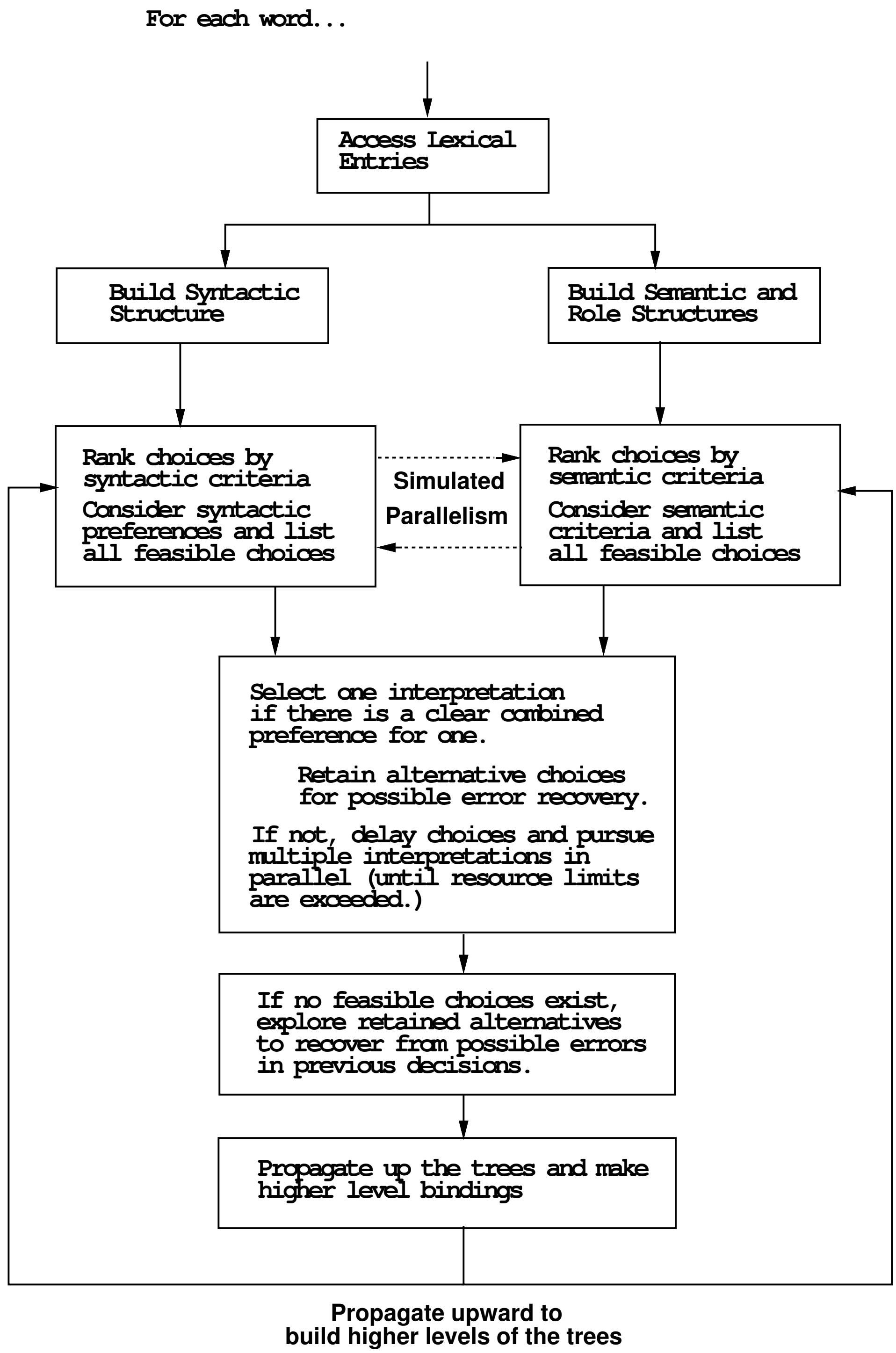

Figure 17: Control Flow in COMPERE.

- **Selecting the best:** The process then selects the best alternatives by taking all the proposals and their preferences into account. It might choose just one or several attachments based on several factors determined by the kinds of ambiguity present at that point. We will come back to those details after seeing the complete picture of the process.
- **Making Attachments:** The process then binds the current structures to the proposed higher-ups and continues cyclically to propose and select ways of attaching the higher-ups to other structures until there is no more attachment possible to higher structures.

Bottom-up expectation-driven parsing with early commitment: In order for syntax to be functionally independent of semantics and to interact with it to carry out integrated processing in a unified process, syntactic processing has to go on hand-in-hand with semantics without unwarranted backtracking or unnecessary commitments or search. Such a method for syntactic parsing is one that is bottom-up, that nevertheless exploits top-down predictions, and makes early commitments when possible. Since it is bottom-up, it would not make unnecessary commitments and pay for their consequences by backtracking. On the other hand, it has to make commitments when there is information that allows one to make those commitments. Otherwise, it would search fruitless paths even when there is disambiguating semantic information thereby violating the integrated processing hypothesis. Syntactic expectations created by such information, as well as semantic and conceptual feedback enable the parser to make such commitments early on in processing a sentence. In spite of all this, since natural language does not always provide information monotonically, the parser could go wrong. In such a case, the parser must be able to recover from its errors using information it had retained about alternatives at previous decision points. The unified process in COMPERE is such a bottom-up, early-commitment parsing mechanism integrated with top-down guidance through expectations. The parsing method being developed is a variant of left-corner parsing methods which have distinct advantages in explaining human parsing behaviors [Abney and Johnson, 1991].

The operators and the control structure that constitute the unified process are shown briefly in the following algorithm:

For each word in a sentence, in left to right order,

1. Access lexical entries (word).
2. Create instance node (syntactic category);
   Create instance node (meanings);
   Create instance node (primitive thematic role).
3. Compute feasible bindings to parents (syntactic instance node);
   Compute feasible bindings to parents (role instance node).
   [These operations check any conditions to be satisfied to make the binding feasible; they also take existing expectations into account.]
4. Rank feasible bindings by syntactic preference criteria (syntactic bindings);
   Rank feasible bindings by semantic preference criteria (role bindings).
   Combine feasible bindings and select the most preferred binding.
5. Make the binding by creating parent node instances and appropriate links, and making any expectations. Create links between correspondings instances in syntax and their thematic roles and meanings.
6. Retain alternative bindings for possible error recovery.
7. If there is no feasible binding for a node, explore previously retained alternatives to recover from errors.
8. Continue to bind the parent nodes to nodes further up as far as possible (such as until the S node in syntax or the Event node in semantics).

Interaction Mechanisms: The description of the operations above does not explain how structural and conceptual information interact with each other. As shown in the control structure in Figure 17, processing of the different kinds of knowledge happens in a coroutine fashion alternating between syntactic and semantic processing. In particular, whenever a syntactic attachment is proposed or actually made between a pair of nodes, the process looks at the semantics of the link between the two nodes to find corresponding semantic feasibilities and preferences for attachments. This interaction helps semantic processing use syntactic information. Since more than one syntactic attachment at different levels in the parse tree may be made in processing a single word, this interaction may take place more than once in processing a word. At this "sub-word level," COMPERE's process is syntax-first. If it were not, then semantic processing would be ignoring syntactic information that is available. This could result in incorrect semantic decisions and greater needs to recover from errors.

Independence in Processing: This, however, does not mean that semantic processing is dependent on syntactic processing. It is not, since it comes up with its attachment proposals and preferences independently. In particular, if syntax fails to come up with any attachments, then semantics is still called on to propose its role attachments and preferences. Since the process finally makes the selection among alternative attachments, it would make them based only on semantic preferences if syntax did not have any.

From this description of the unified process, it is clear that COMPERE employs the orthogonal decomposition of the sentence understanding task into access, proposal, and selection as discussed earlier.

Output of COMPERE Revisited: We have seen that the input to COMPERE is a sentence in left to right order. COMPERE produces a set of interconnected structures at different levels as output. They are:

- **Parse Trees:** A parse tree at the syntax level showing the surface structure of the sentence. This tree might have disjunctive links between nodes representing ambiguous structures without having to duplicate the common parts. There may be more than one subtree if the input sentence does not parse to a single tree according to the grammar.
- **Role Trees:** A role tree showing which syntactic structures play which roles in the semantics of the sentence. There is a link from a node in the parse tree to a node in the role tree and vice versa. Again, there may be more than one disjoint role subtrees.
- **Conceptual Graph:** A network of nodes which are instances of nodes representing the meaning of the content words in the sentence. These nodes are linked to nodes in the parse and role trees. Links between the conceptual instances show the conceptual relationships described by the sentence.

Upward Propagation: Since the three kinds of output structures built by COMPERE are all interconnected, we can visualize the unified process of COMPERE as propagating the meanings of parts of the sentence up along its syntactic structure until at the end the meaning of the entire sentence is available at the root of the structure such as an S node. As syntactic links are made between children nodes and their parent, the meanings corresponding to the children are propagated up to the parent; they are combined at the parent node in semantically meaningful ways based on the semantic attachments preferred at that level. Thus, at each node in the syntactic structure of a sentence, the meaning of the portion of the sentence spanned by that structure is available, making

COMPERE an online model which always represents the meaning of any part of a sentence it processes. Just as syntactic analysis has the goal of producing a single tree structure for a sentence by merging many subtrees, conceptual structures have an urge to move up along the syntactic structure and to combine with other conceptual structures according to the roles which themselves move up along the role trees. Thus the syntactic structure of a sentence represented in a parse tree acts as a set of pathways for parts of meaning to flow through and merge with other parts at the junctions so as to finally merge into a coherent meaning for the entire sentence at the root of the pathways.

5.5.1 Example

To illustrate the unified process in action, consider the simple sentence in (19).

(19) The man saw the horse.

COMPERE reads the first two words and builds the noun-phrase structure and the corresponding meaning of “the man” with a SUBJ-NP role assigned to it. On reading the word “saw,” syntax builds a verb node V to be added to the parse tree of the current sentence. Semantics builds nodes for an EVENT role and an instance of the SEE structure shown above. These structures have to be connected to other role and meaning structures so far built for the sentence. The structures that exist after reading “The man saw” are shown in Figure 18.

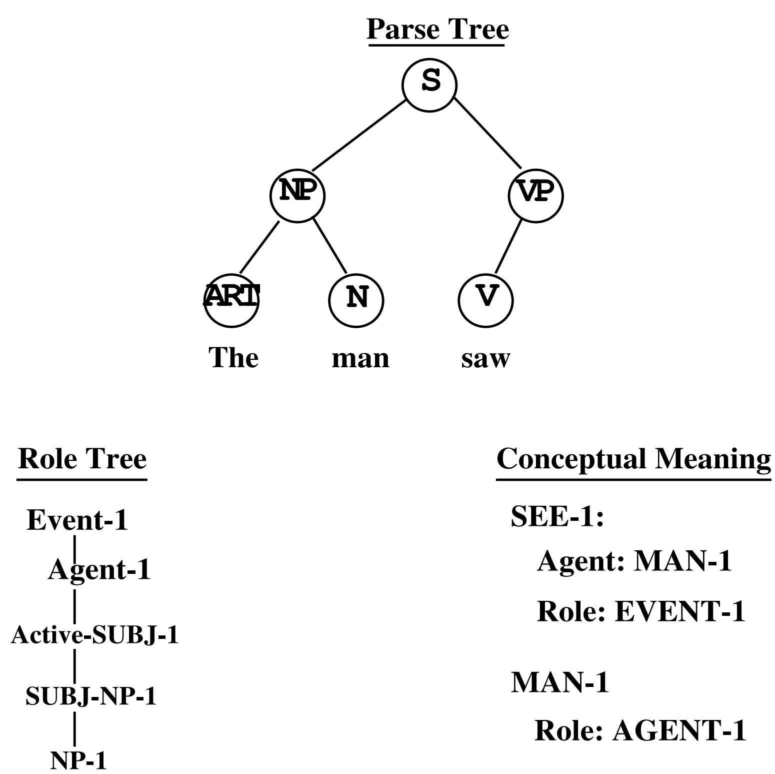

Figure 18: Partial Output from COMPERE.

Now, after reading “the horse,” the system creates an NP node to be connected to the above parse tree, an NP role to be connected to the above role tree, and a HORSE1 structure to be connected to the meaning structures above. Syntax processing could propose a connection from the new NP to the VP in the tree making "the horse" the syntactic object (OBJ-NP role). Semantics finds corresponding links between the HORSE1 node and the SEE1 node through its THEME slot. This results from specializing the OBJ-NP role of "the horse" first to a D-OBJ role and finally to a THEME role which can now be connected to the EVENT1 role. This process can be viewed as the meaning of 'horse' propagating up the parse tree to meet the meaning of 'see' at the VP node where the corresponding semantic connections are found. The structures built at the end of the sentence are shown in Figure 19.

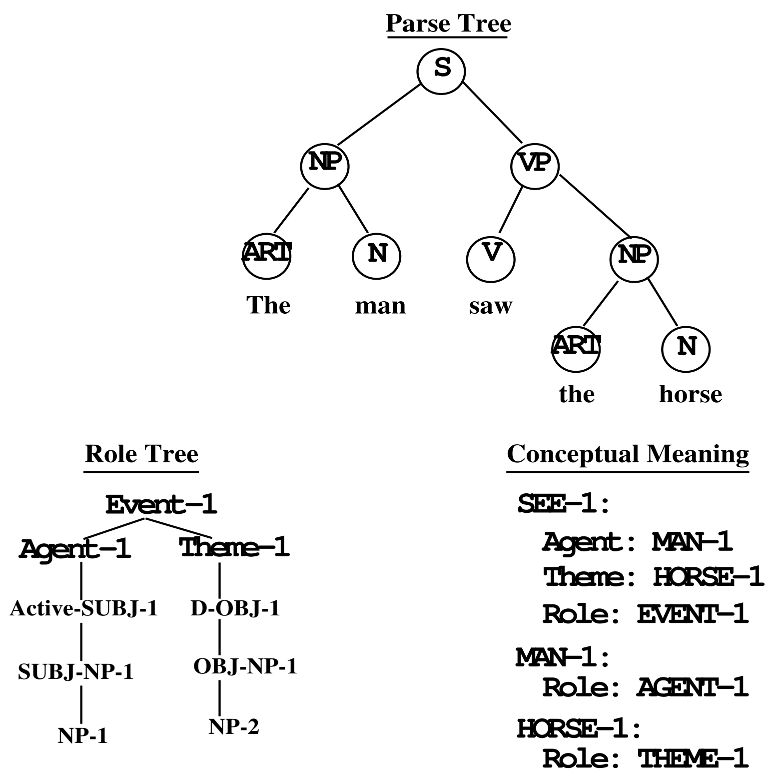

Figure 19: Output from COMPERE.

5.6 Ambiguity Resolution

COMPERE can apply its unified process to resolve both structural and lexical ambiguities. Structural ambiguities in a sentence can be resolved in two ways, using syntactic or semantic preferences. For instance, if (19) were changed to:

(20) The man saw the woman with the horse.

there would be at least two possible interpretations from a syntactic point of view—attaching the prepositional phrase (PP) to the VP or to the object NP—but only one of them is supported by semantics. The NP-attachment interpretation with its "woman together with the horse" meaning is acceptable whereas the VP-attachment interpretation with its "saw using the horse as an instrument" is not acceptable since it violates the constraint that the **INSTRUMENT** slot of the event **SEE** must be filled by an optical instrument. The latter interpretation would get the least preference from semantics.

On the other hand, consider sentence (1) we saw earlier:

(1) The officers taught at the academy were very demanding.

The verb "taught" is interpreted as the main verb of the sentence since that would satisfy the expectation of a VP at that point in processing. In other words, we would rather use the verb to begin the VP that is required to complete the sentence structure, instead of treating it as the verb in a reduced relative clause which would have left the expectation of a VP unsatisfied. This behavior is the same as the one explained by the "first analysis" models of Frazier (1987) and colleagues using a minimal-attachment preference.

Lexical Ambiguities: COMPERE can handle lexical categorial ambiguities such as noun-verb ambiguities at this time. To see how the unified process works when faced with such ambiguities, consider sentence (21) which is locally very ambiguous but globally unambiguous.

(21) The large can can hold the water.

This sentence is a real test for COMPERE's ability to build and maintain multiple interpretations in parallel. The ambiguous structures built by COMPERE for this sentence are shown in Figure 20. COMPERE goes through a series of ambiguous structures to arrive at the correct interpretation of this sentence.[17]

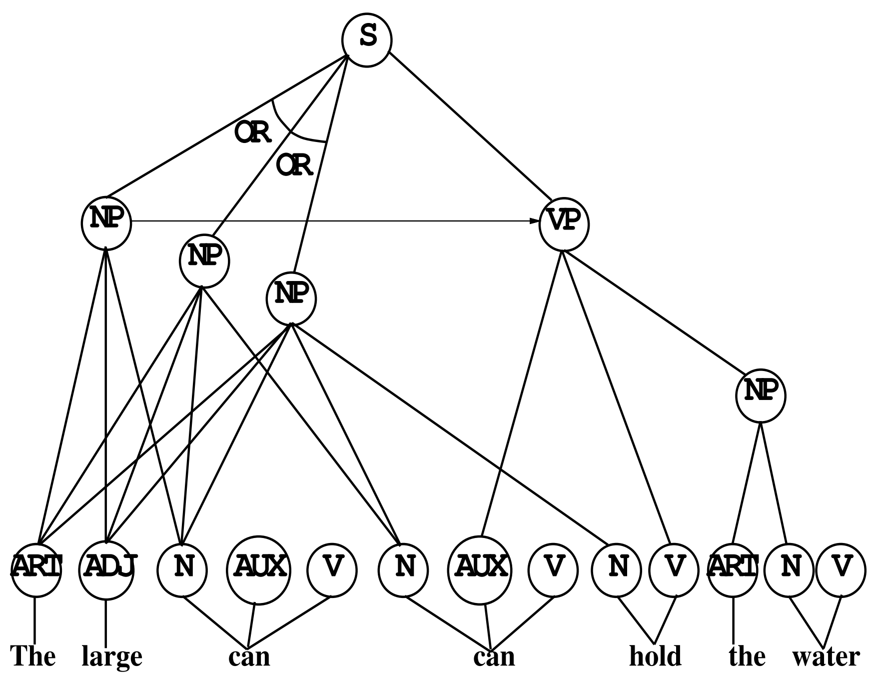

Figure 20: Lexical Category Ambiguities.

[17] In this example, we ignore the possibility that "large" is a noun, the head of the subject NP, for the sake of simplicity.

COMPERE can not deal with lexical semantic and pragmatic ambiguities at present. However, we believe that extending the ambiguity resolution capabilities of the unified process to the domain of semantics amounts to simple and straightforward modifications to the program. COMPERE's ancestor program, ATLAST [Eiselt, 1989], has demonstrated such ambiguity resolution capabilities and we plan to incorporate them in COMPERE in the near future.

5.7 Error Recovery

COMPERE is able to make the kinds of errors that people do and to recover from them automatically, as people do. When choices are made to resolve structural ambiguities, the alternatives that were not selected are retained for possible recovery from erroneous decisions. When it is not possible to attach a structure to the existing tree(s), the previously retained alternatives are examined to see if choosing another alternative at an earlier point provides a way to attach the current structure. If so, the tree is repaired accordingly to recover from the error. Since the subtree that was originally misplaced is merely attached at a different point, error recovery does not amount to reprocessing the structure of the phrase that corresponds to the subtree.

In (1),

(1) The officers taught at the academy were very demanding.

until seeing the word "were," the verb "taught" is treated as the main verb since it satisfies the expectation of a VP that is required to complete the sentence. However, at this point, the structure is incompatible with the remaining input. The processor now tries the other way of attaching the VP as a reduced relative clause so that there will still be a place for a main verb. In doing so, it did not have to reprocess the PP that was part of the VP for the verb "taught."

In resolving the structural ambiguity in (1), semantic preferences did not play a significant role. In other situations, semantic preferences could influence the decisions that the processor makes in resolving syntactic ambiguities. Such behavior would be the same as the ones explained by models which argue for the early effects of semantic and contextual information in syntactic processing [e.g., Crain and Steedman, 1985; Tyler and Marslen-Wilson, 1977]. COMPERE is intended to demonstrate that the range of behaviors that these models account for, and the behaviors that the "first analysis" models [e.g., Frazier, 1987] account for, can be explained by a unified model with a single processor operating on multiple independent sources of knowledge.

5.8 COMPERE, the Program

COMPERE has been implemented in a Common Lisp computer program. We have adopted the method of incremental design in building this program. That is, we have used versions of the program to define and understand the task better and to refine the program into its next version, rather than formally design the entire program beforehand. At this time, COMPERE can process a sentence at the syntactic, semantic, and conceptual levels fairly quickly.[18] Its unified process can resolve structural and lexical syntactic ambiguities. It can deal with fairly complex syntax with relative clauses and complements. However, its role knowledge is fairly limited at this time and its conceptual knowledge is even more so.

The COMPERE program follows closely the spirit of our theoretical model described earlier [Holbrook et al., 1992], but diverges slightly in actual implementation. The divergence appears in

[18] To give a very rough idea of its speed of processing, COMPERE can process the syntax (only) of a ten word sentence in about two seconds on a Symbolics Lisp Machine.

the process itself: the theoretical model has a single unified process, while the prototype computational model has two nearly-identical processes—one for syntax and one for semantics—which share identical control structures but are duplicated for convenience because each process must work with information encoded in slightly different formats. This was an unimportant distinction initially since we set out to explore only syntactic ambiguity resolution and error recovery in the computational model. Now that we are expanding the scope of the program to lexical ambiguities as well, we are in the process of merging the two components of the program into a single unified process.

6 Discussion and Evaluation

Having seen why we need the architecture that we proposed for the sentence processor COMPERE and how it works in understanding sentences with different kinds of ambiguities, let us now look at how we can validate the claims we made at the beginning, and also discuss several related issues regarding the "why"s and "why not"s of the model.

6.1 Validation

We are building our prototype computational model of sentence processing to demonstrate the feasibility of our claims. The current implementation of COMPERE is more complete in syntax than in semantics. The reason for this is it has separate pieces of code for the two. Once we unify the two programs, semantics will also be complete and, perhaps, it will reveal what more it takes for COMPERE to achieve a semantic competence comparable to its syntactic competence.

The model will show the following things:

- The unified process can analyze the syntax and semantics of sentences such as (1) with relative clauses and more than one event described in the sentence.

  (1) The officers taught at the academy were very demanding.

  The program can already do a significant portion of this except that it has two separate pieces of code for syntax and semantics which are similar to one another. They need to be unified into a single program. On the way, we might identify any operations on one kind of knowledge that may be different from the ones on the other kinds of knowledge.

- Demonstrate integrated processing by showing the immediate effect of semantic constraints on assigning syntactic structures. This has been shown in the case of prepositional phrase attachment for instance. However, it should be generalized to show the differences between (1) and (2) for instance.

  (2) The courses taught at the academy were very demanding.

- Demonstrate functional independence. One way to do this is by means of lesion studies [Cottrell, 1985; Small, 1991]. We can turn off syntactic or semantic processing by making the corresponding type of knowledge unavailable to show that the other can still function normally. This was attempted with ungrammatical sentences. However, there is a need to find good examples.

- Show error recovery happening in both syntax and semantics.[19] We have already demonstrated error recovery in resolving structural ambiguities in syntax.

- Show the emergence of roles from syntactic, lexical, and semantic information.[20] We need to show that each of syntactic, lexical, and semantic information can contribute to the determination of roles. We also need to find good examples for this purpose.

[19] It is not yet clear to me what kinds of semantic errors COMPERE can deal with; this should be apparent once the two programs for syntax and semantics are unified.

[20] It may be beyond the scope of this work to show how the overall context in which the sentence is set, or such other factors as rhetorics or genre, influence the assignment of semantic roles.

Since this is an architectural theory, the program will not only produce the same final output in processing a sentence as predicted by the theory, but it will also follow the same course of intermediate decisions and results as observed in human behavior that is being modeled by the theory. The program might also bring to light details that we have failed to realize at this time and thereby suggest modifications to the claims of the theory. Below we discuss how the program will demonstrate the computational feasibility of the claims. We also discuss what changes to the theory we expect the program to make and how we propose to modify the theory in order to incorporate such suggestions.

6.1.1 Example Sentences

Before seeing how the claims will be shown to be feasible by the working of the program, let us look at a set of sentences and the expected output from the program for those sentences. This set of sentences will then be used to show how the claims are met by the model in performing the task (specified earlier in Section 5.2). The sentences (showing the sentence itself which is the input, its parse tree, its meaning representation, and some significant features of the sentence) are:

22 **Ex1:** The officers taught at the academy.

```
(S (NP (ART the) (N officers))                    teach:
   (VP (V taught)                                   AGENT: officer
       (PP (PREP at)                                AT-LOC: academy
           (NP (ART the) (N academy)))))
```

23 **Ex2:** The officers were at the academy.

```
(S (NP (ART the) (N officers))                    be:
   (VP (V were)                                   AGENT: officer
       (PP (PREP at)                              AT-LOC: academy
           (NP (ART the) (N academy)))))
```

[Lexical category ambiguity (AUX/V ambiguity).]

24 **Ex3:** The officers were taught at the academy.

```
(S (NP (ART the) (N officers))                    teach:
   (VP (AUX were)                                   EXPERIENCER: officer
       (V taught)                                   AT-LOC: academy
       (PP (PREP at)
           (NP (ART the) (N academy)))))
```

[Passive voice.]

1 **Ex4:** The officers taught at the academy were very demanding.

```
(S (NP (ART the) (N officers)
       (Rel-Cl (Rel-S (VP (V taught)
                          (PP (PREP at)
                              (NP (ART the) (N academy)))))))
   (VP (V were)
       (ADV very)
       (ADJ demanding)))

be:
 AGENT: officer
 THEME: demanding

teach:
 EXPERIENCER: officer
 AT-LOC: academy
```

[Syntactic garden path, reduced relative clause, passive voice, subcategory ambiguity (taught: past/past participle), structural ambiguity, error recovery.]

25 **Ex5:** The officers who taught at the academy were very demanding.

```
(S (NP (ART the) (N officers)
       (Rel-Cl (Rel-Pro who)
               (Rel-S (VP (V taught)
                          (PP (PREP at)
                              (NP (ART the) (N academy)))))))
   (VP (V were)
       (ADV very)
       (ADJ demanding)))

be:
 AGENT: officer
 THEME: demanding

teach:
 AGENT: officer
 AT-LOC: academy
```

[Relative clause, no garden path.]

26 **Ex6:** The officers who were taught at the academy were very demanding.

```
(S (NP (ART the) (N officers)
       (Rel-Cl (Rel-Pro who)
               (Rel-S (VP (AUX were)
                          (V taught)
                          (PP (PREP at)
                              (NP (ART the) (N academy)))))))
```

```
   (VP (V were)
       (ADV very)
       (ADJ demanding)))

be:
 AGENT: officer
 THEME: demanding

teach:
 EXPERIENCER: officer
 AT-LOC: academy
```

[Relative clause, passive voice, no garden path.]

27 **Ex7:** The officers who taught the men at the academy were very demanding.

```
(S (NP (ART the) (N officers)
       (Rel-Cl (Rel-Pro who)
               (Rel-S (VP (V taught)
                          (NP (ART the)
                              (N men))
                          (PP (PREP at)
                              (NP (ART the) (N academy)))))))
   (VP (V were)
       (ADV very)
       (ADJ demanding)))

be:
 AGENT: officer
 THEME: demanding

teach:
 AGENT: officer
 EXPERIENCER: man
 AT-LOC: academy
```

[Relative clause with transitive verb.]

2 **Ex8:** The courses taught at the academy were very demanding.

```
(S (NP (ART the) (N courses)
       (Rel-Cl (Rel-S (VP (V taught)
                          (PP (PREP at)
                              (NP (ART the) (N academy)))))))
   (VP (V were)
       (ADV very)
       (ADJ demanding)))
```

```
be:
 AGENT: course
 THEME: demanding

teach:
 THEME: course
 AT-LOC: academy
```

[Reduced relative clause, no garden path, (in)animacy effect.]

Several things may be noted from these examples. The program will not deal with such issues as time and tense, number, gender, and so on since we believe that those issues are not central to our theory and are not affected by it very much either. These sentences are mere examples of the kinds of inputs the program can handle; it is expected to be able to process many other sentences of the kind shown in these examples. We plan to test COMPERE on a much larger set of texts in a domain yet to be selected.

Moreover, the output shown above may not be the best or may not capture certain useful distinctions. However, the semantic representations produced by the program are merely illustrative; we are not proposing a theory of semantics here. It might also seem that the events in the main and relative clauses are represented completely separate from one another in the output while they are connected through the relative-clause mechanism in the input sentence. However, the event representations and the thematic roles are all connected to the parse tree shown; they are not disjoint as shown in the simplified pictures above.

We will now consider the claims in the light of the above examples. Though there is no reason to believe that our unified process theory and the architecture we are proposing are applicable only to the restricted task of thematic role assignment, the program can only perform the restricted task due to many unsolved problems in the representation and processing of semantics.

6.1.2 Bottom-up, early-commitment syntax

The sentences shown above show a variety of syntactic features such as relative clauses(Ex4-Ex8), prepositional phrases(Ex1-Ex8), passive forms(Ex3,4,6,8), category ambiguities such as AUX/V ambiguities(Ex2-Ex8), structural ambiguities such as main-V/reduced-relative clause ambiguities (Ex1,4,8), and transitive and intransitive verb structures(Ex1-Ex8). If it is possible for the program to analyze the syntax of these eight sentences and produce the parse trees shown above, we will have shown that the representation and processing method we have proposed work for syntactic analysis of a number of English sentences. This has already been shown in the COMPERE program except for the proper handling of argument structures (transitive/intransitive structures).

The program makes errors and recovers from them in resolving the syntactic ambiguities (in Ex4) in the above sentences. However, as stated earlier, the program only analyzes the surface structure of the sentences without doing any structural transformations. It can not for instance, produce the deep structure of a question through WH-transformations. We assume that question words can be dealt with in semantics, as others have shown in their models [Jurafsky, 1991]. We make no attempt to process question words in this work.

The above analysis does not consider certain grammatical features such as agreement in gender, number, person, tense, reflexives, tag questions, and so on. It appears that one needs to make the relevant information globally available to regions of the parse tree (such as the subtree for a phrase or clause). Then, agreement along various dimensions can be maintained by verifying that the features match one another whenever two structures are combined into one during syntactic analysis.

6.1.3 Role assignment

The sentences above show the effect of syntactic structure(Ex1-Ex8), word meanings and semantic constraints such as animacy(Ex4,8), argument structures(Ex7), passive voice(Ex3,4,8), and relative clauses(Ex4-8) on the assignment of thematic roles. If the program can produce the role assignments shown for the example sentences, it will have shown the part played by each of these kinds of information in role assignment. Roles can then be said to emerge from all these kinds of information which form a structure-to-meaning spectrum starting from surface syntactic form through argument role structures to conceptual representations. Roles played by the parts of the sentence in the events described by the sentence start in the form of primitive roles (such as the NP role) originating from the syntactic categories and then emerge as each new kind of information brings some evidence, to more specialized roles resulting finally in the thematic roles shown above.

We are yet to work out the representations of argument structure information. Moreover, the role hierarchies currently known to the program are incomplete and need to be enhanced and revised. One principle that will be applied throughout is to keep lexical information as minimal as possible. In other words lexical information simply identifies categories which are represented in separate knowledge sources thereby eliminating duplications in representation.

Though we said there was no need for our purposes to perform structural transformations, the variety of relative clauses possible in English might lead to difficulties in role assignment due to structural dependencies between the noun phrases in main and relative clauses. We have not looked at relative clauses headed by prepositions for instance. An example of such a construct would be:

28 The academy at which the officers were taught was very old.

It would be beyond the scope of this work to demonstrate the effects of the global context in which the sentence is used (or the genre of the text and so on) on the assignment of roles.

6.1.4 Unified process claim

The above sentences show how syntax helps in semantic processing(Ex1-8), how semantic constraints help resolve syntactic ambiguities(Ex8), and how knowledge of argument structures and roles bridge syntax and semantics. If the program can process them (i.e., if it can perform the above task on them by producing the output shown above) using a unified process for syntactic and semantic analysis, then it will have demonstrated the computational feasibility of the unified process claim. The current program uses two separate pieces of code for syntactic and semantic analyses. We have however devised similar forms of representation for syntactic and semantic knowledge and we are currently in the process of unifying the two pieces of code.

One area where we might encounter problems in unifying the two programs arises from a basic distinction between syntax and semantics: that syntax cares about the left-to-right contiguity of its constituents in the input while thematic roles do not. This would affect the unified process in its parts where it comes up with one or more interpretations for a part of the input and proposes preferences for each based on the different kinds of information. We believe the effect to amount merely to an additional constraint to be satisfied only while using syntactic knowledge. Such a difference can be viewed as resulting in a unified process that resorts to a few specialized subprocesses at the lowest levels of processing different kinds of knowledge. An example of such a subprocess (originating from the use of word order as a cue in English,) is one that checks to see if the current piece of input is preceded immediately by a unit of a particular syntactic category, or one that checks if a previous unit is contiguous with the current one in the surface form of the input. The unified parts of the process are the ones that combine the results of these different subprocesses and integrate them to make decisions in ambiguity resolution and error recovery.

6.1.5 Functional independence claim

We can show functional independence by artificial lesions in the program. That is, we can temporarily make one or more kinds of knowledge unavailable to see if the other kinds of knowledge can still be applied to make the best decisions with available knowledge. We have already tried this on a few examples and will do it more extensively once the program has been unified.

Functional independence is not a new claim to our theory. We have simply maintained it in our model. It has been shown to work by others and we do not anticipate any major problems in showing its feasibility.

To show the integration of knowledge from different sources during processing we can show the immediate effects of syntactic decisions on semantic role assignment and the immediate effects of semantic constraints on syntactic ambiguity resolution. This can be shown in sentences such as Ex8 and also in prepositional attachment ambiguities which have already been dealt with by the program.

6.1.6 Error recovery claim

Syntactic error recovery in structural ambiguity resolution happens in processing sentence Ex 4. This has already been shown by the COMPERE program both in syntax and in its consequences on the corresponding role assignments in the semantics of the sentence. However, we need to identify the kind or kinds of error recovery possible in semantics and select appropriate examples to show error recovery happening in semantics through the same unified process. We could adapt the examples demonstrated earlier by the ATLAST model. However, they used examples spanning more than a single sentence. We are not sure if we want to extend our theory or any of its claims to beyond individual sentences at this point. Error recovery in semantics will be more easy to address once the unified process in the program is able to process the semantics of the above sentences.

It may be emphasized here that these claims and tasks have always been in our minds when we proposed the model in the first place. We have designed the network representation of syntactic knowledge, for instance, to be able to show the feasibility of a unified process doing integrated processing without sacrificing functional independence. We believe that the groundwork we have laid will enable us to build the computational model further and support the claims we are making.

7 Conclusions

So far, our model of sentence understanding, COMPERE, has shown that its unified-process, independent knowledge-source architecture can process a sentence at the lexical, syntactic, and semantic levels, resolve syntactic and lexical category ambiguities and make and recover from errors in deciding between alternative interpretations just as people do. In particular, it has shown that the error recovery methods originally proposed for lexical and pragmatic ambiguity resolution in the ATLAST model are applicable to syntactic disambiguation as well.

7.1 Issues and Contributions

We have addressed several issues of significance to the study of language processing in this work.

- How does the language processor integrate different kinds of knowledge to bring out the interpretation of a sentence that is best supported by information arising from all the sources of knowledge? We have shown how the application of a single unified process to the multiple independent sources of knowledge can result in a smooth integration of the preferences from the different sources. We have also appealed to the integrated processing hypothesis and shown that such interaction can happen at the earliest opportunities in sentence processing.
- How does the sentence processor integrate information from multiple sources as often and as soon as possible and yet retain the ability to use them independently? This, in essence, is the modularity debate in the study of human language processing. We have shown that by keeping the knowledge sources independent and unifying the process, we can support early and frequent interaction between the use of different knowledge sources at different levels without sacrificing the independence between the use of each.
- How does the language understander cope with the variety of ambiguities in natural languages? We have shown that the same unified process can resolve different kinds of ambiguities without having to do any exhaustive search or much wasteful backtracking.
- How does the language understander recover from errors it makes along the way due to the lack complete information at different points in sentence processing? We have a model that maintains an online interpretation of any portion of the input that is the best with respect to all the information available at that point in processing. Our model also retains information about alternative interpretations so that it can switch to one of the retained interpretations if later information proves the current interpretation wrong. We have shown that this method, proposed originally for semantic and pragmatic error recovery [Eiselt, 1989], is applicable to syntactic error recovery as well.

7.2 Further Work

Though we have presented this work at times as though it has been completed, there is a lot of it that is still to be modeled or not completely understood. A few important issues that we will be focusing on in the near future are listed below.

- The COMPERE program has two nearly identical processes instead of a single process. Our immediate next step will be in merging the two processes for syntactic and semantic analyses into one unified process. In doing so, we might identify small distinctions between different levels in using different kinds of knowledge to come up with alternative interpretations and preferences for them. In order to unify the processes we might have to convert the representations of the different kinds of knowledge into even more similar forms.

- COMPERE has demonstrated some ability to deal with lexical category ambiguities. One of the things we would like to see soon is to extend it so that its ambiguity resolution methods work in lexical semantic ambiguity resolution as well. This is particularly interesting and tractable since the ATLAST model originally proposed these methods for lexical and pragmatic disambiguation in semantics.

- COMPERE has shown error recovery only in syntactic structural ambiguity resolution. We would like to see this happen in lexical and pragmatic ambiguity resolution as well.

Other interesting issues which are, however, outside the scope of our present efforts include the following.

- At present, COMPERE analyzes a sentence in a null context. We would like to introduce the semantic context of a sentence as a new source of information for COMPERE's decision making so that it can model the effects of the context of a sentence on sentence understanding behavior. It is quite possible though that the context might affect disambiguation in indirect ways, say, by altering the preferences from a particular level.

- At present COMPERE does not model parsing breakdown. For instance, it does not experience much difficulty in processing center-embedded sentences. We have to find a good method to impose resource limitations so that we can model parsing breakdown behavior.

- It may not be necessary to build an explicit representation of a syntactic parse tree for a sentence. It seems to be possible to use syntactic cues to influence thematic role assignment without actually building a parse tree. It is not clear how we can do less work in syntax and yet achieve the same end result. However, figuring out a way of doing less explicit syntactic analysis might result in a smoother unification of syntactic and semantic processing.

- Impose working memory limitations [Carpenter and Just, 1988; Just and Carpenter, 1992; King and Just, 1991; MacDonald, Just, and Carpenter, 1992] on the model and use such a limitation to make decisions such as whether to pursue multiple interpretations in parallel or choose one interpretation [Stowe, 1991].

- At some point in the development of this model, we would like to extend COMPERE to understand more than single sentences.

- We have been using a very limited definition of semantics. We could extend the semantics to capture other aspects of meaning such as spatial and temporal relationships [Frawley, 1992].

- We could enhance conceptual processing capabilities and figure out how to use the output of COMPERE in a reasoning task. Another interesting possibility is to devise methods for a reasoner to exert top-down influence on the sentence processor so that higher-level contextual knowledge can be employed in resolving the ambiguities in a sentence.

- Psychological experiments could be conducted to verify the cognitive plausibility of different aspects of the model.

- The performance of the model could be evaluated by means of standardized English tests.

7.3 Ideas

Apart from the important idea of unifying language processing at different levels into a single process, this work has produced other ideas useful in language processing.

- A method for representing and modeling the use of syntactic expectations in parsing.
- Using a form of categorial grammars and a left-corner parser to achieve a proper mixture top-down and bottom-up influences in parsing.
- Representing multiple parallel structural interpretations of a sentence using disjunctive links in a tree.
- The notion of emergence of roles from primitive roles showing the continuum of kinds of information from the surface syntactic to the conceptual and showing how the information in the surface form of a sentence gradually emerges into the meaning of the sentence.
- The structured network representation in which different kinds of knowledge can be represented and which supports dynamic processes such as instantiation.

7.4 Conclusion

We can conclude from this study that a unified process applied independently to multiple knowledge sources answers the modularity debate by supporting interaction between different faculties while retaining independence so as to enable the sentence processor to produce and explain the variety of behaviors observed in human language processing.

Acknowledgements

I would like to thank my advisor Kurt Eiselt for the many invaluable ideas that he has constantly been providing me in doing this work. I would also like to thank him and Jennifer Holbrook for coming up with the unified process hypothesis and for encouraging me to develop a real model of language processing based on it. I also thank Justin Peterson for all the numerous discussions and suggestions.

References

[Abney and Johnson, 1991] S. P. Abney and M. Johnson. Memory Requirements and Local Ambiguities of Parsing Strategies. *Journal of Psycholinguistic Research*, 20(3):233–250, 1991.

[Abney, 1989] S. P. Abney. A Computational Model of Human Parsing. *Journal of Psycholinguistic Research*, 18(1):129–144, 1989.

[Allen, 1987] J. Allen. *Natural Language Understanding.* The Benjamin/Cummings Publishing Company, Inc., 1987.

[Bates et al., 1991] E. Bates, B. Wulfeck, and B. MacWhinney. Cross-Linguistic Research in Aphasia: An Overview. *Brain and Language*, 41(2):123–148, August 1991.

[Birnbaum and Selfridge, 1981] L. Birnbaum and M. Selfridge. Conceptual analysis of natural language. In R. Schank and C. Riesbeck, editors, *Inside Computer Understanding*, pages 318–353. Lawrence Erlbaum Associates, 1981.

[Birnbaum, 1986] L. Birnbaum. *Integrated Processing in Planning and Understanding.* Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, December 1986. Research Report #489.

[Birnbaum, 1989] L. Birnbaum. A Critical Look at the Foundations of Autonomous Syntactic Analysis. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 99–106. Cognitive Science Society, 1989.

[Caramazza and Berndt, 1978] A. Caramazza and R. S. Berndt. Semantic and syntactic processes in aphasia: A review of the literature. *Psychological Bulletin*, 85:898–918, 1978.

[Carpenter and Daneman, 1981] P. A. Carpenter and M. Daneman. Lexical Retrieval and Error Recovery in Reading: A Model Based on Eye Fixations. *Journal of Verbal Learning and Verbal Behavior*, 20:137–160, 1981.

[Carpenter and Just, 1988] P. A. Carpenter and M. A. Just. The Role of Working Memory in Language Comprehension. In D. Klahr and K. Kotovsky, editors, *Complex information processing: The impact of Herbert A. Simon.* Erlbaum, 1988.

[Charniak, 1983] E. Charniak. Passing Markers: A Theory of Contextual Influence in Language Comprehension. *Cognitive Science*, 7:171–190, 1983.

[Chomsky, 1957] N. Chomsky. *Syntactic structures.* Mouton, 1957.

[Clifton and Ferreira, 1987] C. Clifton and F. Ferreira. Modularity in sentence comprehension. In J. L. Garfield, editor, *Modularity in Knowledge Representation and Natural-Language Understanding.* MIT Press, 1987.

[Cottrell, 1985] G. W. Cottrell. Connectionist Parsing. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society, Irvine, CA*, pages 201–211, August 1985.

[Crain and Steedman, 1985] S. Crain and M. Steedman. On not being led up the garden path: the use of context by the psychological syntax processor. In D. R. Dowty, L. Karttunen, and A. M. Zwicky, editors, *Natural Language Parsing: Psychological, computational, and theoretical perspectives.* Cambridge University Press, 1985.

[Eiselt and Holbrook, 1991] K. P. Eiselt and J. K. Holbrook. Toward a Unified Theory of Lexical Error Recovery. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*. Cognitive Science Society, August 1991.

[Eiselt et al., To appear] K. Eiselt, K. Mahesh, and J. Holbrook. Having Your Cake and Eating It Too: Autonomy and Interaction in a Model of Sentence Processing. To appear in the Proceedings of the Eleventh National Conference on Artificial Intelligence, AAAI-93, July 11-16, 1993, To appear.

[Eiselt, 1985] K. Eiselt. A Parallel-process Model of On-Line Inference Processing. In *Proc. of IJCAI-85*, pages 863–869, Los Angeles, CA, August 1985.

[Eiselt, 1989] K. P. Eiselt. *Inference Processing and Error Recovery in Sentence Understanding*. Ph.D. thesis, University of California, Irvine, CA, 1989. Tech. Report 89-24.

[Ferreira and Clifton, 1985] F. Ferreira and C. Clifton. The Independence of Syntactic Processing. *The Journal of Memory and Language*, 25:348–368, 1985.

[Fodor, 1978] J. D. Fodor. Parsing strategies and constraints on transformations. *Linguistic Inquiry*, 9:427–474, 1978.

[Fodor, 1983] J. A. Fodor. *The Modularity of Mind*. The MIT Press, Cambridge, MA, 1983.

[Fodor, 1987] J. A. Fodor. Modules, Frames, Fridgeons, Sleeping Dogs, and the Music of the Spheres. In J. L. Garfield, editor, *Modularity in Knowledge Representation and Natural-Language Understanding*. MIT Press, 1987.

[Ford et al., 1983] M. Ford, J. Bresnan, and R. Kaplan. A competence-based theory of syntactic closure. In J. Bresnan, editor, *The Mental Representation of Grammatical Relations*. MIT Press, 1983.

[Forster, 1979] K. I. Forster. Levels of Processing and the Structure of the Language Processor. In W. E. Cooper and E. C. T. Walker, editors, *Sentence Processing: Psycholinguistic Studies Presented to Merrill Garrett*. Lawrence Erlbaum Associates, 1979.

[Frawley, 1992] W. Frawley. *Linguistic Semantics*. Lawrence Erlbaum Associates, 1992.

[Frazier and Rayner, 1982] L. Frazier and K. Rayner. Making and Correcting Errors during Sentence Comprehension: Eye Movements in the Analysis of Structurally Ambiguous Sentences. *Cognitive Psychology*, 14:178–210, 1982.

[Frazier and Rayner, 1987] L. Frazier and K. Rayner. Resolution of Syntactic Category Ambiguities: Eye Movements in Parsing Lexically Ambiguous Sentences. *Journal of Memory and Language*, 26:505–526, 1987.

[Frazier et al., 1983] L. Frazier, C. Clifton, and J. Randall. Filling Gaps: Decision Principles and Structure in Sentence Comprehension. *Cognition*, 13:187–222, 1983.

[Frazier, 1987] L. Frazier. Theories of Sentence Processing. In J. L. Garfield, editor, *Modularity in Knowledge Representation and Natural Language Understanding*. MIT Press, 1987.

[Frazier, 1989] L. Frazier. Against Lexical Generation of Syntax. In W. Marslen-Wilson, editor, *Lexical Representation and Process*. MIT Press, 1989.

[Goel and Eiselt, 1991] A. K. Goel and K. P. Eiselt. Mental Models, Text Interpretation, and Knowledge Acquisition. AAAI Spring Symposium on Integrated Intelligent Architectures, 1991.

[Holbrook et al., 1988] J. K. Holbrook, K. P. Eiselt, R. H. Granger, and E. H. Matthei. (Almost) Never Letting Go: Inference Retention during Text Understanding. In S. L. Small, G. W. Cottrell, and M. K. Tanenhaus, editors, *Lexical Ambiguity Resolution: Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*, pages 383–409. Morgan Kaufmann Publishers, 1988.

[Holbrook et al., 1992] J. K. Holbrook, K. P. Eiselt, and K. Mahesh. A Unified Process Model of Syntactic and Semantic Error Recovery in Sentence Understanding. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, pages 195–200. Cognitive Science Society, August 1992.

[Holbrook, 1989] J. K. Holbrook. *Studies of inference retention in lexical ambiguity resolution.* Ph.D. thesis, School of Social Sciences, University of California, Irvine, 1989.

[Holmes et al., 1989] V. M. Holmes, L. Stowe, and L. Cupples. Lexical Expectations in Parsing Complement-Verb Sentences. *Journal of Memory and Language*, 28:668–689, 1989.

[Holmes, 1987] V. M. Holmes. Syntactic parsing: In search of the garden path. In M. Coltheart, editor, *Attention and Performance XII.* Erlbaum, 1987.

[Jacobs et al., 1990] P. S. Jacobs, G. R. Krupka, S. W. McRoy, L. F. Rau, N. K. Sondheimer, and U. Zernik. Generic Text Processing: A Progress Report. In *Proceedings DARPA Speech and Natural Language Workshop*, pages 359–364. Morgan Kaufmann Publishers, June 1990.

[Jurafsky, 1991] D. Jurafsky. An On-Line Model of Human Sentence Interpretation. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 449–454. Cognitive Science Society, August 1991.

[Jurafsky, 1992] D. Jurafsky. An On-Line Computational Model of Human Sentence Interpretation. In *Proceedings of the Tenth National Conference on Artificial Intelligence, AAAI 92*, pages 302–308, July 1992.

[Just and Carpenter, 1992] M. A. Just and P. Carpenter. A Capacity Theory of Comprehension: Individual Differences in Working Memory. *Psychological Review*, 99(1):122–149, 1992.

[Kay, 1973] M. Kay. The MIND System. In R. Rustin, editor, *Natural language processing*, pages 155–188. Algorithmics Press, 1973.

[Kimball, 1973] J. Kimball. Seven Principles of Surface Structure Parsing. *Cognition*, 2:15–47, 1973.

[King and Just, 1991] J. King and M. A. Just. Individual Differences in Syntactic Processing: The Role of Working Memory. *Journal of Memory and Language*, 30:580–602, 1991.

[Lebowitz, 1983] M. Lebowitz. Memory-Based Parsing. *Artificial Intelligence*, 21:363–404, 1983.

[Lehman et al., 1991] J. F. Lehman, R. L. Lewis, and A. Newell. Integrating Knowledge Sources in Language Comprehension. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 461–466, 1991.

[Lehnert et al., 1983] W. G. Lehnert, M. G. Dyer, P. N. Johnson, C. J. Yang, and S. Harley. BORIS - An Experiment in In-Depth Understanding of Narratives. *Artificial Intelligence*, 20(1):15–62, January 1983.

[Lewis, 1992] R. L. Lewis. A Computational Theory of Human Sentence Comprehension. PhD Thesis Proposal, School of Computer Science, Carnegie Mellon University, 1992.

[Lytinen, 1986] S. L. Lytinen. Dynamically combining syntax and semantics in natural language processing. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 574–578, 1986.

[MacDonald et al., 1992] M. C. MacDonald, M. A. Just, and P. A. Carpenter. Working Memory Constraints on the Processing of Syntactic Ambiguity. *Cognitive Psychology*, 24:56–98, 1992.

[Marr, 1982] D. Marr. *Vision: A Computational investigation into the Human Representation and Processing of Visual Information.* W. H. Freeman, San Francisco, 1982.

[Marslen-Wilson and Tyler, 1987] W. Marslen-Wilson and L. K. Tyler. Against Modularity. In J. L. Garfield, editor, *Modularity in Knowledge Representation and Natural-Language Understanding.* MIT Press, 1987.

[McRoy and Hirst, 1990] S. W. McRoy and G. Hirst. Race-Based Parsing and Syntactic Disambiguation. *Cognitive Science*, 14:313–353, 1990.

[Pearlmutter and MacDonald, 1992] N. J. Pearlmutter and M. C. MacDonald. Plausibility and Syntactic Ambiguity Resolution. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, pages 498–503. Lawrence Erlbaum Associates, 1992.

[Peterson and Mahesh, Unpublished] J. Peterson and K. Mahesh. Building a natural language understanding system wearing many hats. Unpublished manuscript, Unpublished.

[Ram, 1989] A. Ram. *Question-driven understanding: An integrated theory of story understanding, memory and learning.* Ph.D. thesis, Yale University, New Haven, CT, May 1989. Research Report #710.

[Rayner, 1978] K. Rayner. Eye Movements in Reading and Information Processing. *Psychological Bulletin*, 85:618–660, 1978.

[Riesbeck and Martin, 1986a] C. K. Riesbeck and C. E. Martin. Direct Memory Access Parsing. In J. L. Kolodner and C. K. Riesbeck, editors, *Experience, memory, and reasoning*, pages 209–226. Lawrence Erlbaum, Hillsdale, NJ, 1986.

[Riesbeck and Martin, 1986b] C. K. Riesbeck and C. E. Martin. Towards Completely Integrated Parsing and Inferencing. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, pages 381–387. Cognitive Science Society, August 1986.

[Schank et al., 1980] R. C. Schank, M. Lebowitz, and L. Birnbaum. An Integrated Understander. *American Journal of Computational Linguistics*, 6(1):13–30, 1980.

[Seidenberg et al., 1982] M. S. Seidenberg, M. K. Tanenhaus, J. M. Leiman, and M. Bienkowski. Automatic Access of the Meanings of Ambiguous Words in Context: Some Limitations of Knowledge-Based Processing. *Cognitive Psychology*, 14:489–537, 1982.

[Small, 1991] S. L. Small. Focal and Diffuse Lesions of Cognitive Models. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 85–90. Cognitive Science Society, August 1991.

[Spivey-Knowlton, 1992] M. J. Spivey-Knowlton. Another Context Effect in Sentence Processing: Implications for the Principle of Referential Support. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, pages 486–491. Lawrence Erlbaum Associates, 1992.

[Steedman, 1987] M. Steedman. Combinatory Grammars and Human Sentence Processing. In J. L. Garfield, editor, *Modularity in Knowledge Representation and Natural Language UNnderstanding*. MIT Press, 1987.

[Steedman, 1989] M. J. Steedman. Grammar, Interpretation, and Processing from the Lexicon. In W. Marslen-Wilson, editor, *Lexical Representation and Process*. MIT Press, 1989.

[Stowe, 1991] L. A. Stowe. Ambiguity Resolution: Behavioral Evidence for a Delay. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 257–262. Cognitive Science Society, August 1991.

[Stowe, In press] L. Stowe. Thematic structures and sentence comprehension. In G. Carlson and M. Tanenhaus, editors, *Linguistic Structure in Language Processing*. Dordrecht: Reidel, In press.

[Tanenhaus and Carlson, 1989] M. K. Tanenhaus and G. N. Carlson. Lexical Structure and Language Comprehension. In W. Marslen-Wilson, editor, *Lexical Representation and Process*. MIT Press, 1989.

[Tanenhaus et al., 1987] M. K. Tanenhaus, G. S. Dell, and G. Carlson. Context Effects in Lexical Processing: A Connectionist Approach to Modularity. In J. L. Garfield, editor, *Modularity in Knowledge Representation and Natural-Language Understanding*. MIT Press, 1987.

[Tanenhaus et al., 1991] M. Tanenhaus, S. Garnsey, and J. Boland. Combinatory Lexical Information and Language Comprehension. In G. Altmann, editor, *Cognitive Models of Speech Processing: Psycholinguistic and Computational Perspectives*. MIT Press, 1991.

[Taraban and McClelland, 1988] R. Taraban and J. L. McClelland. Constituent Attachment and Thematic Role Assignment in Sentence Processing: Influences of Content-Based Expectations. *Journal of Memory and Language*, 27:597–632, 1988.

[Trueswell and Tanenhaus, 1992] J. C. Trueswell and M. K. Tanenhaus. Consulting temporal context during sentence comprehension: Evidence from the monitoring of eye movements in reading. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, pages 492–497. Lawrence Erlbaum Associates, 1992.

[Tyler and Marslen-Wilson, 1977] L. K. Tyler and W. D. Marslen-Wilson. The On-Line Effects of Semantic Context on Syntactic Processing. *Journal of Verbal Learning and Verbal Behavior*, 16:683–692, 1977.

[Waltz and Pollack, 1984] D. L. Waltz and J. B. Pollack. Phenomenologically Plausible Parsing. In *Proc. AAAI-84*, pages 335–339, 1984.

[Waltz and Pollack, 1985] D. L. Waltz and J. B. Pollack. Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation. *Cognitive Science*, 9:51–74, 1985.

[Wanner and Maratsos, 1978] E. Wanner and M. Maratsos. An ATN Approach to Comprehension. In M. Halle, J. Bresnan, and G. A. Miller, editors, *Linguistic Theory and Psychological Reality*, pages 119–161. MIT Press, 1978.

[Winograd, 1973] T. Winograd. A Procedural Model of Language Understanding. In R. C. Schank and K. M. Colby, editors, *Computer models of thought and language*, pages 152–186. W. H. Freeman, 1973.

[Woods, 1970] W. A. Woods. Transition network grammars for natural language analysis. *Communications of the ACM*, 13:591–606, 1970. Also reprinted in Readings in Natural Language Processing, Grosz, Jones, and Webber (ed.), Morgan Kaufmann Publishers, 1986.

[Woods, 1973] W. A. Woods. An experimental parsing system for transition network grammars. In R. Rustin, editor, *Natural language processing*. Algorithmics Press, 1973.