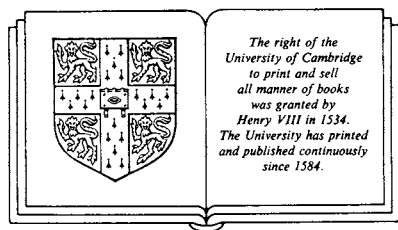


Semantic interpretation and the resolution of ambiguity

GRAEME HIRST

Department of Computer Science, University of Toronto



CAMBRIDGE UNIVERSITY PRESS

CAMBRIDGE

NEW YORK PORT CHESTER

MELBOURNE SYDNEY

Published by the Press Syndicate of the University of Cambridge
The Pitt Building, Trumpington Street, Cambridge CB2 1RP
40 West 20th Street, New York, NY 10011-4211, USA
10 Stamford Road, Oakleigh, Victoria 3166, Australia

© Cambridge University Press 1987

First published 1987

First paperback edition 1992

British Library cataloguing in publication data

Hirst, Graeme

Semantic interpretation and the resolution of ambiguity.–

(Studies in natural language processing)

1. Semantics–Data processing

I. Title II. Series

412 P325.5D38

Library of Congress cataloguing in publication data

Hirst, Graeme

Semantic interpretation and the resolution of ambiguity.

(Studies in natural language processing)

Bibliography.

Includes indexes.

1. Semantics–Data processing. 2. Ambiguity–Data processing. 3. Psycholinguistics–Data processing. I. Title II. Series

P325.5.D38H57 1986 415'.028'5 85-18978

ISBN 0 521 32203 0 hardback

ISBN 0 521 42898 X paperback

Transferred to digital printing 2003

Contents

Preface xi

Organization of the book xi

Notation xii

Acknowledgements xiii

1. Introduction 1

1.1. The problems 2

1.1.1. Semantic interpretation 2

1.1.2. Word sense and case slot disambiguation 4

1.1.3. Syntactic disambiguation 9

1.2. Frames 12

1.3. Artificial intelligence research at Brown University 15

1.3.1. The Frail frame language 16

1.3.2. The Paragram parser 18

1.4. Policy statements 20

Part I SEMANTIC INTERPRETATION

2. Semantic interpretation 25

2.1. Introduction 25

2.2. Semantic interpretation and linguistic theory 28

2.2.1. Decompositional semantics 28

2.2.2. Model-theoretic and truth-conditional semantics 29

2.3. Semantic interpretation and artificial intelligence 33

2.3.1. Procedural semantics and Woods's semantic interpreter 34

2.3.2. Knowledge semantics 39

2.3.3. Object-oriented semantics 41

2.4. Psycholinguistic research on semantic interpretation 42

2.5. Qualities desirable in a semantic interpreter 43

- 3. The Absity semantic interpreter 44**
 - 3.1. Introduction 44
 - 3.2. Two strategies: Strong typing and tandem processing 44
 - 3.3. The typing system of Absity 46
 - 3.4. Another strategy: Turning meaningful syntax into words 50
 - 3.5. The semantic rules of Absity 52
 - 3.6. Implementation details 56
 - 3.7. Absity as the fulfillment of our dreams 58
 - 3.8. What Absity can't do yet 59
 - 3.9. Conclusion 72

Part II LEXICAL DISAMBIGUATION

- 4. Lexical disambiguation 77**
 - 4.1. What is necessary for lexical disambiguation? 77
 - 4.2. Lexical disambiguation research in AI 80
 - 4.2.1. Wilks: Preference semantics 80
 - 4.2.2. Boguraev: Semantic judgments 81
 - 4.2.3. Hayes: Disambiguation with frames and associations 82
 - 4.2.4. Small: Word expert parsing 83
 - 4.3. Psycholinguistic research on lexical disambiguation 84
 - 4.3.1. Semantic priming and spreading activation 86
 - 4.3.2. Phoneme monitoring and prior choice activation models 87
 - 4.3.3. The ordered search hypothesis 90
 - 4.3.4. Semantic priming and the all-readings hypothesis 92
 - 4.3.5. Conclusion 94
- 5. Polaroid Words 96**
 - 5.1. Introduction 96
 - 5.2. Marker passing 97
 - 5.2.1. Marker passing in Frail 98
 - 5.2.2. Lexical disambiguation with marker passing 100
 - 5.2.3. Constraining marker passing 101
 - 5.3. Polaroid Words 102
 - 5.3.1. What Polaroid Words look like 102
 - 5.3.2. How Polaroid Words operate 106
 - 5.3.3. An example of Polaroid Words in action 108
 - 5.3.4. Recovery from doubt 110
 - 5.3.5. Polaroid Words for bound constituents 112
 - 5.3.6. Cues unused 113
 - 5.4. What Polaroid Words can't do 115
 - 5.5. Slot restriction predicates and selectional restrictions 117

- 5.6. Psychological reality 118
 - 5.6.1. Polaroid Words; marker passing, and psycholinguistic models 118
 - 5.6.2. Psychological non-reality 121
 - 5.6.3. Marker passing, path strength, and magic numbers 124
- 5.7. Conclusion 125

Part III STRUCTURAL DISAMBIGUATION

- 6. Structural disambiguation 131**
 - 6.1. Introduction 131
 - 6.2. Types of structural ambiguity 131
 - 6.2.1. Attachment problems 131
 - 6.2.2. Gap finding and filling 136
 - 6.2.3. Analytical ambiguities 137
 - 6.2.4. The interaction between categorial and structural ambiguity 149
 - 6.2.5. Structural ambiguity as a closure problem 150
 - 6.3. Current theories of structural disambiguation 152
 - 6.3.1. Structural disambiguation in ATN parsers 152
 - 6.3.2. Structural disambiguation in Marcus parsers 153
 - 6.3.3. Ford, Bresnan, and Kaplan's theory of closure 155
 - 6.3.4. Structural disambiguation as presupposition minimization 158
 - 6.4. Conclusion 162
- 7. The Semantic Enquiry Desk 164**
 - 7.1. Introduction 164
 - 7.2. Prepositional phrase attachment 165
 - 7.2.1. Where prepositional phrases may be attached 165
 - 7.2.2. Determinants of prepositional phrase attachment 166
 - 7.2.3. Verb annotations 168
 - 7.2.4. Plausibility 168
 - 7.2.5. Testing for presupposition satisfaction 170
 - 7.2.6. Making the attachment decision 172
 - 7.2.7. Muffling combinatorial explosions 175
 - 7.3. Gap finding in relative clauses 177
 - 7.3.1. Why gap finding is difficult 177
 - 7.3.2. How to find gaps 178
 - 7.4. Methods for other structural ambiguities 182
 - 7.4.1. Particle detection 182
 - 7.4.2. Relative clause attachment 184
 - 7.4.3. Adverb attachment 184
 - 7.5. Conclusion 186

Part IV CONCLUSION

- 8. Conclusion** 189
 - 8.1. Towards a conclusion 189
 - 8.2. The work in review 189
 - 8.2.1. What has been achieved 189
 - 8.2.2. Implicit claims 190
 - 8.2.3. Applications of the work 192
 - 8.3. Related work of others 192
 - 8.3.1. Semantic interpretation and disambiguation 193
 - 8.3.2. Knowledge representation for language understanding 199
 - 8.3.3. Spreading activation in natural language systems 199
 - 8.3.4. Lexical disambiguation in text databases 200

- 9. Speculations, Partially Baked Ideas, and Exercises for the Reader** 202
 - 9.1. The representation of knowledge 202
 - 9.2. Semantic formalisms 207
 - 9.3. Semantic interpretation and discourse pragmatics 209
 - 9.4. Lexical ambiguity 212
 - 9.5. Structural ambiguity 215
 - 9.6. None of the above 220

- References** 225

- Index of names** 251

- Index of subjects** 256

1 Introduction

Like all other scientists, linguists wish they were physicists. They dream of performing classic feats like dropping grapefruits off the Leaning Tower of Pisa, of stunning the world with pithy truths like $F = ma$. . . [But instead,] the modern linguist spends his or her time starring or unstarring terse unlikely sentences like “John, Bill, and Tom killed each other”, which seethe with repressed frustration and are difficult to work into a conversation.

—Joseph D Becker¹

WHEN I TOOK my first linguistics course, freshman transformational syntax, in 1974, we were taught that syntax was now basically under control. Sure, people still argued over particular transformations, and this was still all new and exciting stuff, but there was general agreement on the approach. Semantics, on the other hand, was a difficult and tenuous territory; no one yet understood what a semantic was. Semantics was said to have the same qualities as God or Mind—fun to argue about, but inherently unknowable. The study of semantics was left, therefore, until junior year.

Given linguists with attitudes like those toward semantics, it is not surprising that consumers of linguistic theory, such as researchers in natural language understanding, took semantic matters into their own hands. The result was approaches to semantics that were exemplary in their own terms but lacked a firm theoretical basis and hence were inadequate in their relationship to other aspects of language and to wider issues of meaning and representation of meaning. The best example of this is the dissertation of Woods (1967), which I will discuss in some detail in section 2.3.1.

But times are changing. Linguists are much braver than they used to be, and exciting new things are happening in the study of linguistic semantics. Probably the most important is Montague semantics (Montague 1973), which remained for several years a small and arcane area, but which has now attracted a large amount of interest. It is therefore time to start importing modern semantic theories into NLU and examining them to see how they can be applied, how they need to be adapted, and what their limitations are.

¹Becker 1975: 70.

It is the goal of this work to do just this, with a particular emphasis on semantic interpretation. I will be using new approaches to semantics to help put NLU semantic interpretation onto a firmer and more theoretical foundation, and, in the framework thereby set up, to look at issues of lexical and structural disambiguation.

1.1 The problems

The problems discussed in this monograph can be divided into three distinct (but, of course, related) areas: semantic interpretation, word sense disambiguation, and structural disambiguation. In this section I explain each of these problem areas.

1.1.1 Semantic interpretation

By SEMANTIC INTERPRETATION we mean the process of mapping a syntactically analyzed text of natural language to a representation of its meaning. The input to a semantic interpreter is a parse tree, but we do not require that it represent a complete sentence; we allow well-formed subtrees such as noun phrases and even single words (labeled with their part of speech and syntactic features) as input. The output of a semantic interpreter is the meaning of the input text, or a suitable representation thereof. I will discuss in chapters 2 and 3 what such a representation might be; for the present, we will just observe that the natural language input text itself is a representation of its meaning, but not a “suitable” one.

I exclude from semantic interpretation all aspects of syntactic analysis; rather, I assume the existence of a parser that performs morphological and syntactic analysis upon an input text before it is passed to the semantic interpreter. This is not in any sense a logical necessity; systems have been built in which syntactic analysis and semantic interpretation have been completely integrated—*e.g.*, Riesbeck 1974, 1975; Riesbeck and Schank 1978; Hendrix 1977a, 1977b; Cater 1981, 1982. However, this approach becomes very messy when complex syntactic constructions are considered. Keeping syntactic and semantic analysis separate is well motivated merely by basic computer science principles of modularity. Moreover, it is my observation that those who argue for the integration of syntactic and semantic processing are usually disparaging the role of syntax,² a position that I reject (see

²“The theory of syntax is an artifact that cannot be used as a foundation for parsing; stereotypic patterns of lexical interactions cannot account for the highly particular nature of context and usage” (Small 1980: 12).

“Syntactic and semantic processing is [*sic*] done at the same time, with the primacy of semantics over syntax . . . Syntax is used only when it helps in semantic analysis” (Gershman (1979: 11), describing Riesbeck’s parser).

See also Schank and Birbaum 1980.

also Charniak 1983b), and one which has been found to be unworkable³ and, probably, psychologically unreal (Marslen-Wilson and Tyler 1980: 62–66; Tanenhaus and Donnanwerth-Nolan 1984) (*see section 2.4*). This is not to say that parsing is possible without semantic help; in chapters 6 and 7 we will see many situations, such as prepositional phrase and relative clause attachment, in which the parser must call on semantic knowledge. In this book, I will show that syntax and semantics may work together well and yet remain distinct modules.⁴

Lytinen (1984) argues for a “compromise” position. It seems to me, however, that his position is much closer to the separation camp, in which he places the present work, than the integration camp. He states five principles regarding the interaction of syntax and semantics (1984: 4–5), and I am in agreement with about 4.3 of them:

1. Syntactic and semantic processing of text should proceed at the same time.
2. Syntactic decisions must be made with full access to semantic processing; that is, communication between syntax and semantics is high.
3. [Only] a limited amount of syntactic representation [need] be built during text understanding.
4. Knowledge about syntax and semantics is largely separate. Syntactic knowledge should be expressed in the parser’s knowledge base as a largely separate body of knowledge, but this knowledge should have references to semantics, telling the system how semantic representations are built from these syntactic rules.
5. Semantics guides the parsing process, but relies on syntactic rules to make sure that it is making the right decisions.

³“The conclusion that must be drawn from [these experiments] is that if a semantic parser operates without a complete syntactic parse of its input, it is difficult, if not impossible, to prevent it finding readings which do not in fact exist” (Tait 1982, comparing the parsers of Boguraev 1979 and Cater 1981).

“It is fair to say that none of these ‘semantically based’ approaches has succeeded in producing anything like the clear, communicable framework that seems to be offered by a syntactic parser sitting together with a semantic interpreter. As a result, people are continuing to write new and better syntactic parsers, and more and more complete grammars to be used with them in two-part natural language processing systems. The advantages of modularity and portability to new application areas seem to outweigh any other arguments that there may be” (Mellish 1982a).

“[The] basic assumption [of Riesbeck’s parser] that every part of the input sentence has to be specifically expected by some previously built structure does not always work” (Gershman 1979: 11). (Gershman adapted Riesbeck’s parser by adding two new modes of processing, with much more syntactic knowledge.)

“The [not-much-syntax] models that have been presented to date . . . are and will remain fundamentally inadequate to handle the range of grammatical phenomena well known and understood within the linguistics community for the last ten years” (Marcus 1984: 254).

See also chapter 4 of Lytinen 1984 for a critique of integrating syntax and semantics.

⁴Mellish (1982a) suggests the possibility of a system in which separate syntactic and semantic modules are automatically compiled into an efficient unified system. Hendler and Phillips (1981) suggest object-oriented computing to achieve this goal.

The system I will develop in this book is in full accord with principles 1, 2, and 5. I will not quite meet principle 4; rather, we shall see that the relationship between syntactic and semantic rules need not be quite so explicit. Principle 3 is usually true; the internal structure of a syntactic constituent at any level is almost never used once it and its semantic representation have been built, and it may be immediately discarded. There are, however, two good reasons for constructing a full parse tree anyway:

- **Programming:** It is easier to discard the whole tree at the end of the parse than do it piece by piece during the parse, and it is useful to retain the tree for purposes of debugging the program.
- **Theoretical:** Sentences in which apparently-closed constituents are re-opened are widely acceptable (*see section 6.2.5*); the internal structure must be retained just in case.⁵

I also exclude from semantic interpretation any consideration of discourse pragmatics; rather, discourse pragmatics operates upon the output of the semantic interpreter. Thus, semantic interpretation does not include the resolution in context of anaphors or definite reference, or of deictic or indexical expressions, or the recognition and comprehension of speech acts, irony and sarcasm, metaphor, or other non-literal meanings.⁶ These exclusions should not be thought of as uncontroversial; while few would advocate making speech-act interpretation part of semantic interpretation, Moore (1981) argues that definite reference resolution, as well as certain “local” pragmatic matters, must be resolved during semantic interpretation, and Plantinga (1986) argues that metaphor comprehension cannot be divorced from other aspects of language comprehension.

1.1.2 Word sense and case slot disambiguation

Many words of English have more than one meaning, and many quite common words have a very large number of meanings. Table 1.1 lists the words of English

⁵Moreover, there are sentences that require the surface form to be retained—for example those with SURFACE COUNT ANAPHORS (Hirst 1981a):

- (i) When connecting the toe pin to the ankle rod, make sure that the latter goes underneath the former.
- (ii) When the ankle rod is connected to the toe pin, make sure that the former goes underneath the latter.

and it may be that there are sentences that, similarly, refer into their structure. (I have not been able to find any examples.)

⁶For a discussion of anaphors, definite reference, and their resolution in context, see Hirst 1981a, 1981b. Deictics and indexicals are discussed by Fillmore 1975, Kaplan 1978, 1979, and Levinson 1983. Some useful starting points for reading about the role of speech acts in language are Searle 1969, Cole and Morgan 1975, Boyd and Ferrara 1979, and Levinson 1983. For work in AI on recognizing and understanding speech acts, see Allen 1979, 1983a, 1983b, Allen and Perrault 1980, Perrault and Allen 1980, Brown 1979, 1980.

Table 1.1. Words with the greatest number of senses in the Merriam-Webster Pocket Dictionary (data from Amsler 1980:55–57)

WORD	CATEGORY	NO. OF SENSES	WORD	CATEGORY	NO. OF SENSES
go	verb	63	take	verb	24
fall	verb	35	dead	adj	21
run	verb	35	good	adj	21
turn	verb	31	have	verb	21
way	noun	31	line	noun	21
work	verb	31	pass	verb	21
do	verb	30	touch	verb	21
draw	verb	30	dry	adj	20
play	verb	29	wing	noun	20
get	verb	26	draft	noun	19
form	noun	24	give	verb	19
make	verb	24	turn	noun	19
strike	verb	24			

that Amsler (1980, 1981, 1982a) found to have the greatest number of senses listed in *The Merriam-Webster pocket dictionary*. Any practical NLU system must be able to disambiguate words with multiple meanings, and the method used to do this must necessarily work with the methods of semantic interpretation and knowledge representation used in the system.

There are three types of lexical ambiguity: POLYSEMY, HOMONYMY, and CATEGORIAL AMBIGUITY. Polysemous words are those whose several meanings are related to one another. For example, the verb *open* has many senses concerning unfolding, expanding, revealing, moving to an open position, making openings in, and so on. Conversely, homonymous words have meanings with no relationship one to another.⁷ For example, *bark* means both **the noise a dog makes** and **the stuff on the outside of a tree**. A word may be both polysemous and homonymous; the adjective *right* has several senses concerning correctness and righteousness, but also senses concerning the right-hand side.⁸ There is no clear line between

⁷The terminology in this area can be a little confusing. Strictly speaking, since we are interested in written language, the homonymous words we are concerned with are HOMOGRAHS, that is words where many meanings are associated with the same lexeme, though different meanings may have different pronunciations. For example, the vowel varies in *row* depending on whether it means **a line of objects** or **a commotion**, but this fact is of no consequence when dealing with written language. If we were concerned with speech recognition, the type of homonyms we would worry about would be HOMOPHONES—words that are pronounced the same but possibly spelled differently, such as *four* and *fore*. A HETERONYM is a non-homophonic homograph (Drury 1983).

⁸A common etymology does not preclude the senses being distinct enough to be considered homony-

polysemy, homonymy, and metaphor; today's metaphor may be tomorrow's polysemy or homonymy. For example, there is an obvious relationship between *mouth* in the sense of **a person's mouth** and in the sense of **the mouth of a river**, but for practical purposes they are quite separate concepts, and it is not clear into which category *mouth* should therefore be placed.

Categorially ambiguous words are those whose syntactic category may vary. For example, *sink* can be a noun describing a **plumbing fixture** or a verb meaning **become submerged**. Clearly, categorial ambiguity is orthogonal to the other types: the ambiguity of *respect* is categorial and polysemous, as its noun and verb meanings are related, but that of *sink* is categorial and homonymous, as its noun and verb meanings are not related. Categorial ambiguity is mainly a problem in parsing, and I will say no more about it in this monograph, except where it interacts with other types of ambiguity. (See Milne 1980, 1986 for a discussion of handling categorial ambiguity in a deterministic parser.)

Generally, verbs tend to polysemy while nouns tend to homonymy, though of course there are many homonymous verbs and polysemous nouns.⁹ This is consistent with the suggestion of Gentner (1981a, 1981b) that verbs are more "adjustable" than nouns; that is, nouns tend to refer to fixed entities, while verb meanings are easily adjusted to fit the context, with frequent adjustments becoming lexicalized as new but related senses of the original verb.

Panman (1982) argues that although experiments, including his own, have shown that people's intuitions do distinguish between polysemy and homonymy, it is difficult and probably unnecessary to maintain the distinction at the level of linguistic theory. While it seems strange that a cognitively real linguistic phenomenon should have no place in linguistic theory, I too will make little use of it in this work. The semantic objects we will be using are discrete entities,¹⁰ and if a word maps to more than one such entity, it will generally (but not always) be a matter of indifference how closely related those two entities are.

For an NLU system to be able to disambiguate words,¹¹ it is necessary that it use both the discourse context in which the word occurs and local cues within the sentence itself. In this book, I discuss how this may best be done in conjunction with my approach to semantic interpretation, although the techniques will not be limited to use solely within my approach.

mous in everyday modern usage.

⁹In general, adjectives show less ambiguity than nouns and verbs, and this is reflected in Table 1.1.

¹⁰There are those who would argue that this fact immediately damns the whole approach. But there are no well-developed models yet for any form of non-discrete semantics in AI, though current research in fine-grained connectionist systems may change this.

¹¹It is not always the case that an NLU system need worry about disambiguation at all; in some applications, such as machine translation, it is acceptable to ask the user for help (Tomita 1984) or simply preserve the ambiguity in the system's output (Hirst 1981a[1]:68, fn 11; 1981b:90, fn 10; Pericliev 1984).

Ideally, an NLU system should be able to go beyond polysemy and into metaphor. In the most general case, this is an extremely difficult task, and I do not attempt it in this research. For a discussion of metaphor in NLU, see Russell 1976, Wilks 1977, Browse 1978, Hobbs 1979, Carbonell 1981, and Plantinga 1986.¹²

A problem closely related to lexical disambiguation is case slot disambiguation. Case theories of language are generally associated with the work of Fillmore (1968, 1977). In its most basic form, case theory views a sentence as an assertion whose predicate is denoted by the verb of the sentence and whose arguments are denoted by the noun phrases. For example:

(1-1) Nadia tickled Ross with a feather.

Here, *Nadia* is the AGENT of the verb *tickle*, and we say that *Nadia* FILLS THE SLOT of the AGENT CASE. Similarly, *Ross* fills the the PATIENT slot, and *a feather* is in the INSTRUMENT case. We say that the INSTRUMENT case is FLAGGED by the preposition *with*; the AGENT and PATIENT cases are flagged by subject and object position respectively.

There is no rigid one-to-one mapping between flags and cases, however; that is, case flags are not unambiguous. For example, *with* can also flag the cases MANNER¹³ and ACCOMPANIER:

(1-2) Nadia tickled Ross with glee.

(1-3) Ross flew to Casablanca with Nadia.

Also, a case may have more than one flag, often varying with different verbs. For example, some verbs allow the INSTRUMENT in the subject position when no AGENT is specified:

(1-4) The feather tickled Ross.

Thus, different verbs take different cases and different flag-to-case mappings; however, there is still a great degree of regularity in case systems that we will be able to use.

This explanation of cases is greatly simplified, and a few extra points should be made. First, not all prepositional phrases are case-flags and fillers; PPs can qualify nouns as well as verbs. Second, an adverb can act as a combined case-flag and filler:

(1-5) Nadia tickled Ross gleefully.

¹²Also of interest here is the work of Granger (1977; Granger, Staros, Taylor and Yoshii 1983) on determining the meaning of an unknown word from context. If it has been determined that a particular instance of a word does not accord with its normal usage, techniques such as Granger's may be applied. Metaphor, of course, provides more semantic constraints than are available in the general case of a hapax legomenon.

¹³Strictly speaking, MANNER is not a case at all but a verb modifier; see footnote 15.

In this example, *gleefully* behaves exactly as *with glee* does in (1-2).¹⁴ Third, subordinate clauses also exhibit case behavior, with the conjunction as the flag and the sentence as the filler:

- (1-6) Because Ross couldn't bring himself to touch the geranium, Nadia put it in an old shoe box for him.

The word *because* flags the REASON case here. Fourth, there are good linguistic reasons (e.g., Bresnan 1982b) for distinguishing between cases and certain VERB-MODIFYING PPs that describe such things as the time or place at which an action occurs:

- (1-7) Nadia tickled Ross on Friday at the Art Museum.

In the present research, this distinction will not in general be necessary, and we will usually be able to treat all verb-attached PPs in the same way.¹⁵

Clearly, the assignment of a case to a flag depends on the meaning of the potential slot-filler; we know that in (1-4) *the feather* is not in the AGENT case because

¹⁴In English, we can think of the suffix *-ly* as a flag for the MANNER case. However, English morphology is not quite regular enough to permit a general morphological analysis of case-flags in adverbs; rather, we just think of both the flag and the case-filler being bundled up together in the word's meaning.

¹⁵An example of the linguistic distinction between case-fillers and verb modifiers is that PP verb modifiers are sensitive to adverb movement and may be put at the start of the sentence, while PP case-fillers must usually follow the verb unless topicalized. Thus it sounds strange to say (i) instead of (ii):

- (i) *On the boat, Ross put his luggage.
(ii) Ross put his luggage on the boat.

where *on the boat* is in the LOCATION case, but one can say both (iii) and (iv):

- (iii) On the boat, Ross was having fun.
(iv) Ross was having fun on the boat.

where *on the boat* is a PLACE verb qualifier. Also, modifiers may sometimes not come between the verb and true case-fillers. Thus one can say (v) but not (vi):

- (v) Ross put his luggage on the boat on Tuesday.
(vi) *Ross put his luggage on Tuesday on the boat.

But:

- (vii) Ross threatened Nadia with a wrench in the park.
(viii) Ross threatened Nadia in the park with a wrench.

(Barbara Brunson, personal communication). The most important difference, however, is the BIUNIQUENESS CONDITION (Bresnan 1982b): each case may appear at most once in a sentence, while there is no restriction on how many times each modifier type may appear. Thus one may not have two INSTRUMENT cases; if more than one INSTRUMENT has to be specified, conjunction must be used with a single flag; examples from Bresnan 1982b:

- (ix) *Ross escaped from prison with dynamite with a machine gun.
(x) Ross escaped from prison with dynamite and a machine gun.

On the other hand, (xi) (also from Bresnan 1982b) contains three MANNERS, three TIMES and two PLACES:

AGENTS must be conscious animate entities. Thus the problem of determining which case slot a particular preposition or syntactic position flags is very similar to that of lexical disambiguation: in each, semantic information is necessary to decide which one of a set of meanings is to be assigned to a particular token. In the present research, I will show how the two tasks may indeed be accomplished by very similar mechanisms.

1.1.3 Syntactic disambiguation

Although many sentences of English have more than one parse,¹⁶ there is usually a unique preferred parse for a sentence after semantics and discourse context are considered. For example, in (1-8):

(1-8) Nadia left the university on the wrong bus.

we do not take *the university on the wrong bus* as a single noun phrase; rather, we apply the knowledge that universities seldom ride buses. That is, there is a SEMANTIC BIAS to one of the parses. Bias may also come from context; the parse of a sentence such as (1-9):

(1-9) They're cooking apples.

depends on whether it answers the question *What are they doing in the kitchen?* or *What kind of apples are those?*

In addition, the English language often exhibits certain preferences—SYNTACTIC BIASES—in choosing among several possible parses. Thus (1-10) was judged silly by informants:

(1-10) The landlord painted all the walls with cracks.¹⁷

(xi) Ross deftly handed a toy to the baby by reaching behind his back over lunch at noon in a restaurant last Sunday in the Back Bay without interrupting the discussion.

Nevertheless, there are certain restrictions on using the same flag twice in the same way, even for modifiers; thus *with* can't be used twice in the same sentence for MANNER:

(xii) *Ross chaired the meeting with tact with dignity.

(xiii) Ross chaired the meeting with tact and dignity.

The apparent exception (xiv) may be explained as an elliptical form of (xv):

(xiv) Ross chaired the meeting with tact and with dignity.

(xv) Ross chaired the meeting with tact and Ross chaired the meeting with dignity.

See also Somers 1984 and Brunson 1986a.

¹⁶Church and Patil (1982) point out that some sentences can have parses numbering in the hundreds if semantic constraints are not considered.

¹⁷From Rayner, Carlson, and Frazier (1983), who took it to be semantically well-formed.

who generally said that though the intent was clear, it sounded like either the walls were painted with a crack-like pattern or that cracks were being used to paint the walls, readings that both have the prepositional phrase attached to the verb phrase of the sentence instead of to the object noun phrase. Similarly, (1-11):

(1-11) Ross baked the cake in the freezer.

was taken by informants to mean that the baking in some bizarre way took place in the freezer, rather than that the particular cake known to have once been in the freezer was baked in a conventional manner.¹⁸ PPs starting with *by* often sound like passives even when a locative reading makes more sense:

(1-12) SHOESHINE BY ESCALATOR¹⁹

(1-13) Ross was told what to do by the river.

Sentence (1-12) seems to be saying that the escalator shines one's shoes, and (1-13) sounds like Ross heard voices in the running water; even though the meaning in which Ross received instructions from an unspecified person on the river bank makes more sense, the parse that treats *the river* as the deep-structure subject is still preferred.²⁰ The following are reported by Cutler (1982b) as "slips of the ear":

(1-14) You never actually see a forklift truck, let alone person.

(Perceiver attempted to access a compound noun, forklift person, as if a second occurrence of forklift had been deleted.)

(1-15) The result was recently replicated by someone at the University of Minnesota in children.

(Perceiver assigned NP status to the University of Minnesota in children, cf. the University of California in Berkeley.)

Cutler attributes such errors to the hearer; I am inclined to say, rather, that the error was the speaker's in creating a sentence whose structure misled the speaker. The main point, however, is that the speaker WAS misled into an anomalous interpretation consistent with the sentence structure, despite the availability of a sensible interpretation "close by".

The source of syntactic bias is disputed. Frazier (1978; Frazier and JD Fodor 1978) has suggested two principles for the preferred placement of a constituent whose role is ambiguous:

¹⁸Sentence (1-11) and its test on informants is due to Barbara Brunson.

¹⁹Sign at American Airlines terminal, LaGuardia airport, New York, November 1984.

²⁰Marcus (1980: 228–234) argues that, at least in some cases, when syntactic and semantic biases conflict and neither is strong enough to override the other, the sentence is judged ill-formed. The argument is based on subtleties of well-formedness that vary widely over idiolects, and I am not convinced of the generality of the hypothesis.

- Right Association (also called Low Right Attachment, Late Closure, or Local Association): A new constituent is attached as low and as far to the right in the parse tree as possible.²¹
- Minimal Attachment: A new constituent is attached to the parse tree using as few non-terminal nodes as possible.

These principles predict many of the syntactic biases of English; Frazier (1978:115; Frazier and Fodor 1978) shows that they are inherent consequences of a two-stage parsing model she presents, and Milne (1982a) has shown them to be a natural consequence of Marcus parsing (*see section 1.3.2*). However, the principles sometimes conflict, or interact in complex ways. In cases such as prepositional phrase attachment, when both a noun phrase and its dominating verb phrase could receive the PP, Low Right Attachment suggests that the NP (the lowest, right-most node) should take it, while Minimal Attachment prefers the VP because NP attachment allegedly requires an extra NP node above the resulting complex NP.^{22,23} Sentence (1-10) shows that common sense is not always used to resolve the conflict, and Ford, Bresnan, and Kaplan (1982) and Crain and Steedman (1985) have proposed a different set of principles, which I will discuss in detail in sections 6.3.3 and 6.3.4.

Many sentences that are structurally unambiguous are, however, **LOCALLY AMBIGUOUS**: they contain a point at which, in left-to-right parsing, the parser could take one of several paths, and the information that determines which is correct occurs only later in the sentence. In the case of parsers with limited lookahead, such as Marcus parsers (*see section 1.3.2*), the disambiguating information may be out of sight and a choice may have to be made without it. If this choice is wrong, the parser will eventually find itself off in entirely the wrong direction, unable to find any correct parse. A sentence that can do this to a parser is said to be a **SYNTACTIC GARDEN-PATH SENTENCE**,²⁴ in that it leads the parser “down the garden path”; the unfortunate parser is said to have been “GARDEN-PATHED”. Many well-formed sentences are garden paths for the human mental parsing mechanism:

(1-16) The horse raced past the barn fell.²⁵

²¹This principle was first suggested by Kimball (1973), and was modified by Frazier and Fodor; see Fodor and Frazier 1980 for discussion of the differences.

²²This crucially assumes that noun phrases with PP modifiers are parsed as in (i) rather than (ii):

(i) [NP [NP the noun phrase] [PP with [NP the prepositional phrase]]]

(ii) [NP the noun phrase [PP with [NP the prepositional phrase]]]

Analysis (i) strikes me as dubious, and I use (ii) below.

²³In Frazier’s (1978) two-stage model, Minimal Attachment only occurs (in the second stage) if Right Association (a consequence of the first stage) fails to happen.

²⁴In section 4.3.2, we will see that there are also semantic garden-path sentences.

²⁵From Bever 1970:316.

(1-17) The old dog the footsteps of the young.²⁶

Most people have trouble with these sentences the first time they see them.²⁷ Marcus (1980) argues that it is no shame for a computational parser to be garden-pathed by sentences that also trip humans up, and that such behavior is in fact necessary if claims of cognitive modeling are to be made for the parser. I also take this viewpoint.

To find which parse is the one preferred in each particular case, a parser needs help from both world knowledge and discourse context, as well as knowledge about preferred attachments. In this research, I develop a method of providing such semantic information for a parser—a method that works in concert with the semantic interpretation and lexical disambiguation systems that I also develop.

1.2 Frames

The concept that unifies the approaches to the problems described in the previous section is that of the *FRAME* as a semantic object. I am using the word *frame* in the conventional AI sense: a data item that contains a collection of knowledge about a stereotyped topic (Charniak 1976, 1981a), or represents a concept. A frame is usually structured as a set of *SLOTS* or *ROLES* that may contain *VALUES*; often, a *DEFAULT VALUE* obtains for a slot if no other value is specified. A value may be almost any type of object: a number, a boolean value, another frame (or frame instance—see below), or a homogeneous set of such values. A slot may be marked with restrictions on what sort of values it allows.

Here are some examples. The first set, shown in figure 1.1, is based (loosely) on the frames in Wong 1981b for the task of reading a menu in a restaurant. (The formalism is that for the Frail frame system—see section 1.3.1.) The first frame defined is *read*. Its second line says that it is a (“ISA”) particular type of the *task* frame, and as a consequence *INHERITS* all the properties of that frame. Thus, since *tasks* already have a slot for the *agent* performing the task, it is unnecessary to define the slot again in *read*. On the other hand, *object* is defined at this level of the *FRAME HIERARCHY* (or *ISA HIERARCHY*), because it is not true that all *tasks* have this slot. It is also specified that the filler of this slot must be something that can be read, namely *reading-material*. The *facts* clauses, not shown in detail here, describe the actions involved in reading, starting off with taking the item to be read.

²⁶I believe this example is due to Yorick Wilks.

²⁷Their meanings are, respectively:

- (i) The horse—the one that was raced past the barn—fell.
- (ii) The footsteps of the young are dogged by the old people.

```

frame: read
  isa: task
  slots: object (required) (reading-material)
  facts: (take ?agent ?object)
  ...

frame: possibilities-list
  isa: reading-material
  slots: type-of-possibilities

frame: menu
  isa: possibilities-list
  facts: (type-of-possibilities food)

instance: menu34
  isa: menu

```

Figure 1.1. Frail frames describing some of the knowledge necessary to understand the concept of reading a restaurant menu (from Wong 1981b).

The next frames define `possibilities-list`, `menu`, and also a particular INSTANCE of a menu, namely one given the arbitrary name `menu34`. Because `menu34` has `reading-material` as an ancestor in the hierarchy, it will be allowed to fill the `object` slot in any instance of the `read` frame. An instance is necessarily a leaf in the frame hierarchy, and no other frame may be defined in terms of it. Nodes that are not instances are said to be `GENERIC`.

The second example set, in figure 1.2, is from KRL (DG Bobrow and Winograd 1977, 1979). (I have modified it slightly for this exposition.) First the `BusinessTravel` and `Visit` frames are defined, and then an item that is simultaneously an instance of both is given. The `SELF` clause gives the frame's parent in the frame hierarchy, from which properties may be inherited. The following clauses list slots, giving their names and restrictions on their fillers; for example, the `visitor` slot can only be filled by an instance of the `Person` frame. `Event137` is an instance of the `Visit` frame, and particular values for the slots of `Visit` are given. It is also an instance of `BusinessTravel`; note that since a value is not given for the `mode` slot, it will take on the default value `Plane`. Because a KRL frame can be an instance of more than one frame, the frame hierarchy of KRL is a network, not just a tree as in Frail.

Thus we can think of a generic frame as a representation of a concept and an instance as one specific occurrence of the concept. Slots can be thought of as arguments or parameters of the concept, so that the filled slots of an instance describe or qualify that instance.

Many different frame systems have been developed in AI—another important one not mentioned above is `KL-ONE` (Brachman 1978)—and there is still much