

Using Multiple Knowledge Sources for Word Sense Discrimination

Susan W. McRoy*

Artificial Intelligence Program
GE Research and Development Center

This paper addresses the problem of how to identify the intended meaning of individual words in unrestricted texts, without necessarily having access to complete representations of sentences. To discriminate senses, an understander can consider a diversity of information, including syntactic tags, word frequencies, collocations, semantic context, role-related expectations, and syntactic restrictions. However, current approaches make use of only small subsets of this information. Here we will describe how to use the whole range of information. Our discussion will include how the preference cues relate to general lexical and conceptual knowledge and to more specialized knowledge of collocations and contexts. We will describe a method of combining cues on the basis of their individual specificity, rather than a fixed ranking among cue-types. We will also discuss an application of the approach in a system that computes sense tags for arbitrary texts, even when it is unable to determine a single syntactic or semantic representation for some sentences.

1. Introduction

Many problems in applied natural language processing — including information retrieval, database generation from text, and machine translation — hinge on relating words to other words that are similar in meaning. Current approaches to these applications are often *word-based* — that is, they treat words in the input as strings, mapping them directly to other words. However, the fact that many words have multiple senses and different words often have similar meanings limits the accuracy of such systems. An alternative is to use a knowledge representation, or interlingua, to reflect text content, thereby separating text representation from the individual words. These approaches can, in principle, be more accurate than word-based approaches, but have not been sufficiently robust to perform any practical text processing task. Their lack of robustness is generally due to the difficulty in building knowledge bases that are sufficient for broad-scale processing.

But a synthesis is possible. Applications can achieve greater accuracy by working at the level of word senses instead of word strings. That is, they would operate on text in which each word has been *tagged* with its sense. Robustness need not be sacrificed, however, because this tagging does not require a full-blown semantic analysis. Demonstrating this claim is one of the goals of this paper.

Here is an example of the level of analysis a sense tagger would provide to an application program. Suppose that the input is (1):

* Correspondence should be addressed to the author at Department of Computer Science, University of Toronto, Toronto, Canada M5S 1A4 or mcroy@ai.toronto.edu.

Example 1

The agreement reached by the state and the EPA provides for the safe storage of the waste.

The analysis would provide an application with the following information.

- *agreement* refers to a state resulting from concurrence, rather than an act, object, or state of being equivalent.
- *reach* is intended to mean ‘achieve,’ rather than ‘extend an arm.’
- *state* refers to a government body, rather than an abstract state of existence.
- *safe* in this context is an adjective corresponding to ‘secure,’ rather than a noun corresponding to a container for valuables.
- The EPA and the state were co-agents in completing some agreement that is instrumental in supplying a secure place to keep garbage, rather than there was some equivalence that extended its arm around the state while the EPA was busy filling safes with trash.

Preliminary evidence suggests that having access to a sense tagging of the text improves the performance of information retrieval systems (Krovetz 1989).

The primary goal of this paper, then, is to describe in detail methods and knowledge that will enable a language analyzer to tag each word with its sense. To demonstrate that the approach is sufficiently robust for practical tasks, the article will also discuss the incorporation of the approach into an existing system, TRUMP (Jacobs 1986, 1987, 1989), and the application of it to unrestricted texts. The principles that make up the approach are completely general, however, and not just specific to TRUMP.

An analyzer whose tasks include word-sense tagging must be able to take an input text, determine the concept that each word or phrase denotes, and identify the role relationships that link these concepts. Because determining this information accurately is knowledge-intensive, the analyzer should be as flexible as possible, requiring a minimum amount of customization for different domains. One way to gain such flexibility is give the system enough generic information about word senses and semantic relations so that it will be able to handle texts spanning more than a single domain.

While having an extensive grammar and lexicon is essential for any system’s domain independence, this increased flexibility also introduces degrees of ambiguity not frequently addressed by current NLP work. Typically, the system will have to choose from several senses for each word. For example, we found that TRUMP’s base of nearly 10,000 root senses and 10,000 derivations provides an average of approximately four senses for each word of a sentence taken from the *Wall Street Journal*. The potential for combinatoric explosion resulting from such ambiguity makes it critical to resolve ambiguities quickly and reliably. It is unrealistic to assume that word sense discrimination can be left until parsing is complete, as suggested, for example, by Dahlgren, McDowell, and Stabler (1989) and Janssen (1990).

No simple recipe can resolve the general problem of lexical ambiguity. Although semantic context and selectional restrictions provide good cues to disambiguation, they are neither reliable enough, nor available quickly enough, to be used alone. The approach to disambiguation that we will take below combines many different, strong

sources of information: syntactic tags, word frequencies, collocations, semantic context (clusters), selectional restrictions, and syntactic cues. The approach incorporates a number of innovations, including:

- a hybridization of several lexicons to help control which senses are considered:
 - a static generic lexicon
 - a lexicon linked to collocations
 - a lexicon linked to *concretions* (i.e., specializations of abstract senses of words)
 - lexicons linked to specialized conceptual domains;
- a separate processing phase, prior to parsing, that eliminates some ambiguities and identifies baseline semantic preferences;
- a preference combination mechanism, applied during parsing and semantic interpretation, that uses dynamic measures of strength based on *specificity*, instead of a fixed, ordered set of rules.

Although improvements to our system are ongoing, it already interprets arbitrary text and makes coarse word sense selections reasonably well. (Section 6 will give some quantitative assessments.) No other system, to our knowledge, has been as successful.

We will now review word sense discrimination and the determination of role relations. In Section 3, we discuss some sources of knowledge relevant to solving these problems, and, in Section 4, how TRUMP's semantic interpreter uses this knowledge to identify sense preferences. Section 5 describes how it combines the preference information to select senses. Afterward, we will discuss the results of our methods and the avenues for improvement that remain.

2. Cues to Word Sense Discrimination

The problem of word sense discrimination is to choose, for a particular word in a particular context, which of its possible senses is the "correct" one for the context. Information about senses can come from a wide variety of sources:

- the analysis of each word into its root and affixes, that is, its **morphology**;
- the contextually appropriate part or parts of speech of each word, that is, its **syntactic tag** or **tags**;
- for each sense of the word, whether the sense is preferred or deprecated — either in general, because of its frequency, or in the context, because it is the expected one for a domain;
- whether a word is part of a common expression, or **collocation**, such as a nominal compound (e.g., *soda cracker*) or a predicative relation (e.g., *take action*);
- whether a word sense is supported by the semantic context — for example, by its association with other senses in the context sharing a semantic category, a situation, or a topic;
- whether the input satisfies the expectations created by syntactic cues (e.g., some senses only take arguments of a particular syntactic type);

- whether it satisfies role-related expectations (i.e., expectations regarding the semantic relations that link syntactically attached objects);
- whether the input refers to something already active in the discourse focus.

Of course, not all these cues will be equally useful.

We have found that, in general, the most important sources of information for word sense discrimination are syntactic tags, morphology, collocations, and word associations. Role-related expectations are also important, but to a slightly lesser degree. Syntactic tags are very important, because knowing the intended part of speech is often enough to identify the correct sense. For example, according to our lexicon, when *safe* is used as an adjective (as in Example 1), it always denotes the sense related to security, whereas *safe* used as a noun always denotes a type of container for storing valuables.

Morphology is also a strong cue to discrimination because certain sense–affix combinations are preferred, deprecated, or forbidden. Consider the word *agreement*. The verb *agree* can mean either ‘concur,’ ‘benefit,’ or ‘be equivalent’ and, in general, adding the affix *-ment* to a verb creates a noun corresponding either to an act, or to its result, its object, or its associated state. However, of the twelve possible combinations of root sense and affix sense, in practice only four occur: *agreement* can refer only to the act, object, or result in the case of the ‘concur’ sense of *agree* or the state in the case of the ‘equivalence’ sense of *agree*. Furthermore, the last of these combinations is deprecated.

Collocations and word associations are also important sources of information because they are usually “dead giveaways,” that is, they make immediate and obvious sense selections. For example, when paired with *increase*, the preposition *in* clearly denotes a patient rather than a temporal or spatial location, or a direction. Word associations such as *bank/money* similarly create a bias for the related senses. Despite their apparent strength, however, the preferences created by these cues are not absolute, as other cues may defeat them. For example, although normally the collocation *wait on* means ‘serve’ (*Mary waited on John*), the failure of a role-related expectation, such as that the BENEFICIARY be animate, can override this preference (*Mary waited on the steps*). Thus, collocations and word associations are strong sources of information that an understander must weigh against other cues, and not just treat as rules for sense-filtering (as in Hirst 1987 or Dahlgren, McDowell, and Stabler 1989).

The selection of a role relationship can both influence and be influenced by the selection of word senses, because preferences partially constrain the various combinations of a role, its holder, and the filler. For example, the preposition *from* prefers referring to the SOURCE role; transfers, such as *give*, prefer to have a DESTINATION role; and instances of colors, such as *red*, prefer to fill a COLOR role. Approaches based on the word disambiguation model tend to apply constraint satisfaction techniques to combine these role preferences (Hirst 1987). Preferences based on role-related expectations are often only a weak cue because they are primarily for verbs and not normally very restrictive.

Although generally a weak cue, role-related preferences are quite valuable for the disambiguation of prepositions. In our view, prepositions should be treated essentially the same as other words in the lexicon. The meaning of a preposition either names a relation directly, as one of its core senses (Hirst [1987] also allows this), or indirectly, as a specialized sense triggered, for example, by a collocation or concretion. Because the meaning of a preposition actually names a relation, relation-based cues are a good source of information for disambiguating them. (References to objects in the discourse

focus can also be a strong cue for disambiguating prepositions, but this cue appears fairly infrequently [Whittemore, Ferrara, and Brunner 1990].)

The problem of determining role relationships entangles word sense discrimination with the problem of syntactic attachment. The attachment problem is a direct result of the ambiguity in determining whether a concept is related to an adjacent object, or to some enveloping structure that incorporates the adjacent object. Most proposed solutions to this problem specify a fixed set of ordered rules that a system applies until a unique, satisfactory attachment is found (Fodor and Frazier 1980; Wilks, Huang, and Fass 1985; Shieber 1983; Hirst 1987; Dahlgren, McDowell, and Stabler 1989). Such rules can be either syntactic, semantic, or pragmatic. Syntactic rules attempt to solve the attachment problem independent of the sense discrimination problem. For example, a rule for Right Association (also known as Late Closure) says to prefer attaching a new word to the lowest nonterminal node on the rightmost branch of the current structure (i.e., in the same structure as the last word processed) (Kimball 1973). Semantic rules, by contrast, intertwine the problems of discrimination and attachment; one must examine all combinations of senses and attachments to locate the semantically best one. Such rules normally also collapse the attachment problem into the conceptual role filling problem. For example, a *lexical preference* rule specifies that the preference for a particular attachment depends on how strongly or weakly the verb of the clause prefers its possible arguments (Fodor 1978; Ford, Bresnan, and Kaplan 1982). Pragmatic rules also intermingle sense discrimination and attachment, but consider the context of the utterance. For example, one suggested rule says to prefer to build structures describing objects just mentioned (Crain and Steedman 1985; Altmann and Steedman 1988).

The accuracy of systems with fixed-order rules is limited by the fact that it is not always possible to strictly order a set of rules independent of the context. For example, Dahlgren, McDowell, and Stabler (1989) propose the rule “If the object of the preposition is an expression of time, then S-attach the PP” to explain the preference for assuming that “in the afternoon” modifies *adjourn* in Example 2:

Example 2

The judge adjourned the hearing in the afternoon.

Although they admit this rule would fail for a sentence like *John described the meeting on January 20th*, where the NP has a lexical preference for a time modifier, lexical preferences are not always the determining factor either. The existence of a conceptually similar object in the context (such as “the morning trial”) can also create an expectation for the grouping “hearing in the afternoon,” as in Example 3 below.

Example 3

The judge had to leave town for the day. He found a replacement to take over his morning trial, but couldn’t find anyone else that was available. He called the courthouse and cancelled the hearing in the afternoon.

Moreover, pragmatic effects are not always the determining factor either, leading many people to judge the following sentence as silly (Hirst 1987).

Example 4

The landlord painted all the walls with cracks (Rayner, Carlson, and Frazier 1983).

The presence of different lexical items or different objects in the discourse focus may strengthen or weaken the information provided by an individual rule. Another possibility we will discuss in Section 5 is to weigh all preference information dynamically (cf. Schubert 1986; McRoy and Hirst 1990).

The system we will be describing in Section 4 will use many of the cues described above, including syntactic tags, morphology, word associations, and role-related expectations. But first, we need to discuss the sources of knowledge that enable a system to identify these cues.

3. Sources of Knowledge

To identify preference cues such as morphology, word frequency, collocations, semantic contexts, syntactic expectations, and conceptual relations in unrestricted texts, a system needs a large amount of knowledge in each category. In most cases, this just means that the understander's lexicon and conceptual hierarchy must include preference information, although processing concerns suggest moving some information out of these structures and into data modules specific to a particular process, such as identifying collocations. TRUMP obtains the necessary knowledge from a moderately sized lexicon (8,775 unique roots), specifically designed for use in language understanding, and a hierarchy of nearly 1,000 higher-level concepts, overlaid with approximately 40 concept-cluster definitions. It also uses a library of over 1,400 collocational patterns. We will consider each in turn.

3.1 The Lexicon

Development of TRUMP's current lexicon followed an experiment with a moderately-sized, commercially available lexicon (10,000 unique roots), which demonstrated many substantive problems in applying lexical resources to text processing. Although the lexicon had good morphological and grammatical coverage, as well as a thesaurus-based semantic representation of word meanings, it lacked reasonable information for discriminating senses. The current lexicon, although roughly the same size as the earlier one, has been designed to better meet the needs of producing semantic representations of text. The lexicon features a hierarchy of 1,000 parent concepts for encoding semantic preferences and restrictions, sense-based morphology and subcategorization, a distinction between primary and secondary senses and senses that require particular "triggers" or appear only in specific contexts, and a broad range of collocational information. (An alternative would have been to give up discriminating senses that the lexicon does not distinguish; cf. Janssen [1990].) At this time, the lexicon contains about 13,000 senses and 10,000 explicit derivations.

Each lexical entry provides information about the morphological preferences, sense preferences, and syntactic cues associated with a root, its senses, and their possible derivations. An entry also links words to the conceptual hierarchy by naming the conceptual parent of each sense. If necessary, an entry can also specify the composition of common phrases, such as collocations, that have the root as their head.

TRUMP's lexicon combines a core lexicon with *dynamic* lexicons linked to specialized conceptual domains, collocations, and concretions. The core lexicon contains the generic, or context-independent, senses of each word. The system considers these senses whenever a word appears in the input. The dynamic lexicons contain word senses that normally appear only within a particular context; these senses are considered only when that context is active. This distinction is a product of experience; it is conceivable that a formerly dynamic sense may become static, as when military terms creep into everyday language. The partitioning of the lexicon into static and

dynamic components reduces the number of senses the system must consider in situations where the context does not trigger some dynamic sense. Although the idea of using dynamic lexicons is not new (see Schank and Abelson [1977], for example), our approach is much more flexible than previous ones because TRUMP's lexicon does not link *all* senses to a domain. As a result, the lexical retrieval mechanism never forces the system to use a sense just because the domain has preselected it.

3.1.1 The Core Lexicon. The core lexicon, by design, includes only coarse distinctions between word senses. This means that, for a task such as generating databases from text, task-specific processing or inference must augment the core lexical knowledge, but problems of considering many nuances of meaning or low-frequency senses are avoided. For example, the financial sense of *issue* (e.g., a new security) falls under the same core sense as the latest issue of a magazine. The 'progeny' and 'exit' senses of *issue* are omitted from the lexicon. The idea is to preserve in the core lexicon only the common, coarse distinctions among senses (cf. Frazier and Rayner 1990).

Figure 1 shows the lexical entries for the word *issue*. Each entry has a part of speech, :POS, and a set of core senses, :SENSES. Each sense has a :TYPE field that indicates *primary* for a preferred (primary) sense and *secondary* for a deprecated (secondary) sense. The general rule for determining the :TYPE of a sense is that secondary senses are those that the semantic interpreter should not select without specific contextual information, such as the failure of some selectional restriction pertaining to the primary sense. For example, the word *yard* can mean an enclosed area, a workplace, or a unit of measure, but in the empty context, the enclosed-area sense is assumed. This classification makes clear the relative frequency of the senses. This is in contrast to just listing them in historical order, the approach of many lexicons (such as the *Longman Dictionary of Contemporary English* [Procter 1978]) that have been used in computational applications.

The :PAR field links each word sense to its immediate parent in the semantic hierarchy. (See Section 3.2.) The parents and siblings of the two noun senses of *issue*, which are listed in Figure 2, give an idea of the coverage of the lexicon. In the figure, word senses are given as a root followed by a sense number; conceptual categories are designated by atoms beginning with c-. Explicit derivations, such as "period-ic-al-x," are indicated by roots followed by endings and additional type specifiers. These derivative lexical entries do "double duty" in the lexicon: an application program can use the derivation as well as the semantics of the derivative form.

The :ASSOC field, not currently used in processing, includes the lexicographer's choice of synonym or closely related words for each sense.

The :SYNTAX field encodes syntactic constraints and subcategorizations for each sense. When senses share constraints (not the case in this example), they can be encoded at the level of the word entry. When the syntactic constraints (such as *io-rec*, *one-obj*, and *no-obj*) influence semantic preferences, they are attached to the sense entry. For example, in this case, *issue* used as an intransitive verb (*no-obj*) would favor 'passive moving' even though it is a secondary sense. The *io-rec* subcategorization in the first two senses means indirect object as recipient: the ditransitive form will fill the RECIPIENT role. The grammatical knowledge base of the system relates these subcategories to semantic roles.

The :G-DERIV and :S-DERIV fields mark morphological derivations. The former, which is NIL in the case of *issue* to indicate no derivations, encodes the derivations at the word root level, while the latter encodes them at the sense preference level. For example, the :S-DERIV constraint allows *issuance* to derive from either of the first two senses of the verb, with *issuer* and *issuable* deriving only from the 'giving' sense.

```

( issue
  :POS noun
  :SENSES
  (( issue1
    :EXAMPLE (address important issues)
    :TYPE *primary*
    :PAR (c-concern)
    :ASSOC (subject)  )
   ( issue2
    :EXAMPLE (is that the october issue?)
    :TYPE *secondary*
    :PAR (c-published-document)
    :ASSOC (edition) )))

( issue
  :POS verb
  :G-DERIV nil
  :SENSES
  (( issue1
    :SYNTAX (one-obj io-rec)
    :EXAMPLE (the stockroom issues supplies)
    :TYPE *primary*
    :PAR (c-giving)
    :ASSOC (supply)
    :S-DERIV ((-able adj tr_ability)
              (-ance noun tr_act)
              (-er noun tr_actor))  )
   ( issue2
    :SYNTAX (one-obj io-rec)
    :EXAMPLE (I issued instructions)
    :TYPE *primary*
    :PAR (c-informing)
    :ASSOC (produce)
    :S-DERIV ((-ance noun tr_act))  )
   ( issue3
    :SYNTAX (one-obj no-obj)
    :EXAMPLE (good smells issue from the cake)
    :TYPE *secondary*
    :PAR (c-passive-moving)  )))

```

Figure 1

The lexical entries for *issue*.

The derivation triples encode the form of each affix, the resulting syntactic category (usually redundant), and the “semantic transformation” that applies between the core sense and the resulting sense. For example, the triple *(-er noun tr_actor)* in the entry for *issue* says that an *issuer* plays the ACTOR role of the first sense of the verb *issue*. Because derivations often apply to multiple senses and often result in different semantic transformations (for example, the ending *-ion* can indicate the act of performing some action, the object of the action, or the result of the action), a lexical entry can mark certain interpretations of a morphological derivation as primary or secondary.

NOUN_ISSUE1:

PARENT CHAIN: c-concern c-mental-obj c-obj
 c-entity something

SIBLINGS (all nouns):

regard1 realm2 puzzle1 province2
 premonition1 pity1 pet2 parameter1
 ground3 goodwill1 feeling2 enigma1
 draw2 department2 concern1 cause2
 care1 business3 baby2 apprehend-ion-x

NOUN_ISSUE2:

PARENT CHAIN: c-published-document c-document
 c-phys-obj c-obj c-entity something

SIBLINGS (all nouns):

week-ly-x volume1 transcript1 tragedy2
 tome1 thesaurus1 supplement2 strip4
 source2 software1 serial1 scripture1
 romance2 publication2 profile2 period-ic-al-x
 paperback1 paper3 paper2 pamphlet1
 omnibus1 obituary1 novel1 notice2
 month-ly-x memoir1 map1 manual1
 magazine1 library1 journal1 handbook1
 guide1 grammar1 gazette1 fiction1
 feature4 facsimile1 epic1 encyclopedia1
 dissertation1 directory1 digest1 dictionary1
 copy2 constitute-ion-x1 comic1 column2
 column1 catalogue1 calendar1 bulletin1
 brochure1 book1 blurb1 biography1
 bibliography1 bible1 atlas1 article1
 anthology1

Figure 2

The parents and siblings of two senses of *issue*.

3.1.2 The Dynamic Lexicons. Unlike the core lexicon, which lists senses active in all situations, the dynamic lexicons contain senses that are active only in a particular context. Although these senses require triggers, a sense and its trigger may occur just as frequently as a core sense. Thus, the dynamic-static distinction is orthogonal to the distinction between primary and secondary senses made in the core lexicon.

Currently, TRUMP has lexicons linked to domains, collocations, and concretions. For example, TRUMP's military lexicon contains a sense of *engage* that means 'attack.' However, the system does not consider this sense unless the military domain is active. Similarly, the collocational lexicon contains senses triggered by well-known patterns of words; for example, the sequence *take effect* activates a sense of *take* meaning 'transpire.' (Section 3.3 discusses collocations and their representation in more detail.) Concretions activate specializations of the abstract sense of a word when it occurs with an object of a specific type. For example, in the core lexicon, the verb *project* has the abstract sense 'transfer'; however, if its object is a sound, the system activates a sense corresponding

to a ‘communication event,’ as in *She projected her voice*. Encoding these specializations in the core lexicon would be problematic, because then a system would be forced to resolve such nuances of meaning even when there was not enough information to do so. Dynamic lexicons can provide much finer distinctions among senses than the core lexicon, because they do not increase the amount of ambiguity when their triggering context is inactive.

Together, the core and dynamic lexicons provide the information necessary to recognize morphological preferences, sense preferences, and syntactic cues. They also provide some of the information required to verify and interpret collocations. Sections 3.2, 3.3, and 3.4, below, describe sources of information that enable a system to recognize role-based preferences, collocations, and the semantic context.

3.2 The Concept Hierarchy

The concept hierarchy serves several purposes. First, it associates word senses that are siblings or otherwise closely related in the hierarchy, thus providing a thesaurus for information retrieval and other tasks (cf. Fox et al. 1988). In a sense tagging system, these associations can help determine the semantic context. Second, it supplies the basic ontology to which domain knowledge can be associated, so that each new domain requires only incremental knowledge engineering. Third, it allows role-based preferences, wherever possible, to apply to groups of word senses rather than just individual lexical entries.

To see how the hierarchy’s concept definitions establish the basic ontology, consider Figure 3, the definition of the concept *c-recording*. *c-recording* is the parent concept for activities involving the storage of information, namely, the following verb senses:

```
book2 catalogue1 clock1 compile1
date3 document1 enter3 index1
input1 key1          log1 record1
```

In a concept definition, the `:PAR` fields link the concept to its immediate parents in the hierarchy. The `:ASSOC` field links the derived instances of the given concept to their places in the hierarchy. For example, according to Figure 3, the object form derived

```
(c-ent c-recording
 :DESC (the storing of information)
 :PAR (c-action)
 :PAR (c-simple-occurrence
       :ROLE-PLAY (r-object r-patient))
 :ASSOC ((r-object c-information))
 :PREF ((r-patient c-information)))
```

Figure 3

The conceptual definition of *c-recording*.

```
(c-ent c-clothing
 :DESC (cloth materials for wearing)
 :PAR (c-phys-obj)
 :RELS ((*modified-by*
        (c-fabric-material c-made-of-rel))))
```

Figure 4

The conceptual definition of *c-clothing*.

```
(c-ent c-color-qual
  :DESC (qualities of the color of an entity)
  :PAR (c-phys-prop-qual)
  :RELS ((*transform* (c-state c-color-rel))
         (*modifier-of*
          (c-phys-obj c-color-rel))))
```

Figure 5

The conceptual definition of *c-color-qual*.

```
(c-ent c-made-of-rel
  :DESC (a relationship between an object and what it is made of)
  :PAR (c-phys-prop-rel)
  :PREF ((r-statevalue c-phys-obj)
         (r-stateholder (or c-phys-obj c-whole)))
  :RELS ((*held-by* c-phys-obj)))
```

Figure 6

The conceptual definition of *c-made-of-rel*.

from *enter3* (i.e., *entry*) has the parent *c-information*.

The *:ROLE-PLAY* fields mark specializations of a parent's roles (or introduce new roles). Each *:ROLE-PLAY* indicates the parent's name for a role along with the concept's specialization of it. For example, *c-recording* specializes its inherited *OBJECT* role as *PATIENT*.

The *:RELS* and *:PREF* fields identify which combinations of concept, role, and filler an understander should expect (and hence prefer). For example, the definition in Figure 4 expresses that fabric materials are common modifiers of clothing (e.g., *wool suit*) and fill the clothing's *MADE-OF* role. *TRUMP*'s hierarchy also allows the specification of such preferences from the perspective of the filler, where they can be made more general. For example, although colors are also common modifiers of clothing (e.g., *blue suit*), it is better to associate this preference with the filler (*c-color-qual*) because colors prefer to fill the *COLOR* role of any physical object. (Figure 5 shows an encoding of this preference.) The hierarchy also permits the specification of such preferences from the perspective of the relation underlying a role. For example, the relation *c-made-of* in Figure 6 indicates (in its *:RELS*) that physical objects normally have a *MADE-OF* role and (in its *:PREF*) that the role is normally filled by some physical object. Figure 7 gives a complete account of the use of the *:RELS* and *:PREF* fields and how they permit the expression of role-related preferences from any perspective.

3.3 Collocational Patterns

Collocation is the relationship among any group of words that tend to co-occur in a predictable configuration. Although collocations seem to have a semantic basis, many collocations are best recognized by their syntactic form. Thus, for current purposes, we limit the use of the term "collocation" to sense preferences that result from these well-defined syntactic constructions.¹ For example, the particle combination *pick up*

¹ Traditionally many of these expressions have been categorized as idioms (see Cowie and Mackin 1975; Cowie, Mackin, and McCraig 1983), but as most are at least partly compositional and can be processed by normal parsing methods, we prefer to use the more general term "collocation." This categorization thus happily encompasses both the obvious idioms and the compositional expressions whose status as idioms is highly debatable. Our use of the term is thus similar to that of Smadja and McKeown, who partition collocations into open compounds, predicative relations, and idiomatic expressions (Smadja and McKeown 1990).

Perspective	Preference		
	holder	filler	relation
holder	NA	:RELS ((*modified-by* (filler) (relation))) :PREF ((role) (filler)))	:RELS ((*holder-of* (role))) :PREF ((role) (filler)))
filler	:RELS ((*modifier-of* (holder) (relation)))	NA	:RELS ((*modifier-of* (holder) (relation)))
relation	:RELS ((*held-by* (holder))) :PREF (r-stateholder (filler)))	:PREF (r-statevalue (holder)))	NA

Figure 7

The use of :PREF and :RELS.

1. 249 profit take
2. 205 take place
3. 157 take act
4. 113 say take
5. 113 act take
6. 99 take advantage
7. 94 take effect
8. 88 take profit
9. 77 take step
10. 76 take account

Figure 8

The top ten co-occurrences with *take*.

and the verb-complement combination *make the team* are both collocation-inducing expressions. Excluded from this classification are unstructured associations among senses that establish the general semantic context, for example, *courtroom/defendant*. (We will discuss this type of association in the next section.)

Collocations often introduce dynamic word senses, i.e., ones that behave compositionally, but occur only in the context of the expression, making it inappropriate for the system to consider them outside that context. For example, the collocation *hang from* triggers a sense of *from* that marks an INSTRUMENT. In other cases, a collocation simply creates preferences for selected core senses, as in the pairing of the 'opportunity' sense of *break* with the 'cause-to-have' sense of *give* in *give her a break*. There is also a class of collocations that introduce a noncompositional sense for the entire expression, for example, the collocation *take place* invokes a sense 'transpire.'

To recognize collocations during preprocessing, TRUMP uses a set of patterns, each of which lists the root words or syntactic categories that make up the collocation. For example, the pattern (TAKE (A) (ADJ) BATH) matches the clauses *take a hot bath* and *takes hot baths*. In a pattern, parentheses indicate optionality; the system encodes the repeatability of a category, such as adjectives, procedurally. Currently, there are patterns for verb-particle, verb-preposition, and verb-object collocations, as well as compound nouns.

Initially, we acquired patterns for verb-object collocations by analyzing lists of root word pairs that were weighted for relative co-occurrence in a corpus of articles

from the *Dow Jones News Service* (cf. Church and Hanks 1990; Smadja and McKeown 1990). As an example of the kind of data that we derived, Figure 8 shows the ten most frequent co-occurrences involving the root “take.” Note that the collocation “take action” appears both in its active form (third in the list), as well as its passive, *actions were taken* (fifth in the list).

From an examination of these lists and the contexts in which the pairs appeared in the corpus, we constructed the patterns used by TRUMP to identify collocations. Then, using the patterns as a guide, we added lexical entries for each collocation. (Figure 9 lists some of the entries for the compositional collocations associated with the verb *take*; the entries pair a dynamic sense of *take* with a sense occurring as its complement.)

These entries link the collocations to the semantic hierarchy, and, where appropriate, provide syntactic constraints that the parser can use to verify the presence of a collocation. For example, Figure 10 shows the entry for the noncompositional collocation *take place*, which requires that the object (*r-*tail**) be singular and determinerless.

These entries differ from similar representations of collocations or idioms in Smadja and McKeown (1990) and Stock (1989), in that they are sense-based rather than word-based. That is, instead of expressing collocations as word-templates, the lexicon groups together collocations that combine the same sense of the head verb with particular senses or higher-level concepts (cf. Dyer and Zernik 1986). This approach better addresses the fact that collocations do have a semantic basis, capturing general forms such as *give him or her* (some temporal object), which underlies the collocations *give month*, *give minute*, and *give time*. Currently, the system has entries for over 1700 such collocations.

3.4 Cluster Definitions

The last source of sense preferences we need to consider is the semantic context. Work on lexical cohesion suggests that people use words that repeat a conceptual category or that have a semantic association to each other to create unity in text (Morris 1988; Morris and Hirst 1991; Halliday and Hasan 1976). These associations can be thought of as a class of collocations that lack the predictable syntactic structure of, say, collocations arising from verb-particle or compound noun constructions. Since language producers select senses that group together semantically, a language analyzer should prefer senses that share a semantic association. However, it is unclear whether the benefit of knowing the exact nature of an association would justify the cost of determining it. Thus, our system provides a cluster mechanism for representing and identifying groups of senses that are associated in some unspecified way.

A *cluster* is a set of the senses associated with some central concept. The definition of a cluster includes a name suggesting the central concept and a list of the cluster’s members, as in Figure 11. A cluster may contain concepts or other clusters.

TRUMP’s knowledge base contains three types of clusters: categorial, functional, and situational. The simplest type of cluster is the *categorial* cluster. These clusters consist of the sets of all senses sharing a particular conceptual parent. Since the conceptual hierarchy already encodes these clusters implicitly, we need not write formal cluster definitions for them. Obviously, a sense will belong to a number of categorial clusters, one for each element of its parent chain.

The second type of cluster is the *functional* cluster. These consist of the sets of all senses sharing a specified functional relationship. For example, our system has a small number of part-whole clusters that list the parts associated with the object named by the cluster. Figure 12 shows the part-whole cluster *c1-egg* for parts of an egg.

The third type of cluster, the *situational* cluster, encodes general relationships among senses on the basis of their being associated with a common setting, event,

```

( take
  :POS verb
  :SPECIAL
  (( take50
    :S-COMPOUNDS
    ((vc (or (member c-verb_advise2-obj
                    c-act-of-verb_blame1
                    c-act-of-verb_lose1 noun_profit2)
              c-giving)))
    :EXAMPLE (take delivery)
    :PAR (c-receiving)  )
    ( take51
    :S-COMPOUNDS ((vc (or (member noun_effort1
                              c-temporal-obj c-energy)))
    :EXAMPLE (the job takes up time))
    :PAR (c-require-rel)  )
    ( take52
    :S-COMPOUNDS ((vc (member noun_news1
                              noun_burden1 noun_load2 noun_pressure3
                              noun_pressure2 noun_stress1 noun_stress2
                              c-act-of-verb_strain1)))
    :EXAMPLE (he couldn't take the presssure)
    :PAR (c-managing)  )
    .
    .
    .
    ( take58
    :S-COMPOUNDS ((vc (or (member noun_office2
                              noun_advantage1 noun_charge1
                              c-act-of-verb_control1 noun_command2
                              noun_responsibility1) c-structure-rel
                              c-shape-rel)))
    :EXAMPLE (they took advantage of the situation)
    :PAR (c-contracting)  )
    ( take59
    :S-COMPOUNDS ((vc (member noun_effect1)))
    :EXAMPLE (the new rules take effect today)
    :PAR (c-transpire)  )
    ( take60
    :S-COMPOUNDS ((vc (or c-task)))
    :EXAMPLE (he took the assignment)
    :PAR (c-deciding)  ))

```

Figure 9
Some compositional collocations involving *take*.

```

( take-place1
  :CTYPE vc
  :TAIL noun_place
  :PREF ((r-*tail* (and (fillerp number singular)
                        (fillerp limit null))))
  :PAR (c-transpire)
)

```

Figure 10
The entry for the noncompositional phrase *take place* from the collocational entry for *take*.

```
(c-ent cl-business
  :PAR c-cluster-obj
  :CLUSTERS (c-business-group c-business-manager-human
             c-business-org c-business-qual c-business-human
             c-profession c-employment-action cl-financial)
)
```

Figure 11

The definition of the cluster *cl-business*.

```
(c-ent cl-egg
  :PAR c-cluster-obj
  :CLUSTERS (noun_albumin1 noun_white4 noun_egg1 noun_yolk1))
)
```

Figure 12

The definition of the cluster *cl-egg*.

```
(c-ent cl-courtroom
  :PAR c-cluster-obj
  :CLUSTERS (c-law-action c-law-obj verb_judge1 noun_jury1
             verb_defend1 noun_lawyer1 noun_attorney1
             noun_crime1 noun_plaintiff1 noun_justice1
             noun_justice2 verb_prosecute1 noun_bail1
             noun_plea1 verb_object1 noun_fine1 noun_jail1
             noun_prison1 noun_court1 noun_testimony1
             verb_testify1 verb_try3 verb_swear2 noun_oath1
             noun_truth1 noun_bench2 verb_perjure1))
```

Figure 13

The definition of the cluster *cl-courtroom*.

or purpose. Since a cluster's usefulness is inversely proportional to its size, these clusters normally include only senses that do not occur outside the clustered context or that strongly suggest the clustered context when they occur with some other member of the cluster. Thus, situational clusters are centered upon fairly specific ideas and may correspondingly be very specific with respect to their elements. It is not unusual for a word to be contained in a cluster while its synonyms are not. For example, the cluster *cl-courtroom* shown in Figure 13 contains sense *verb.testify1*, but not *verb.assert1*. Situational clusters capture the associations found in generic descriptions (cf. Dahlgren, McDowell, and Stabler 1989) or dictionary examples (cf. Janssen 1990), but are more compact because clusters may include whole categories of objects (such as *c-law-action*) as members and need not specify relationships between the members. (As mentioned above, the conceptual hierarchy is the best place for encoding known role-related expectations.)

The use of clusters for sense discrimination is also comparable to approaches that favor senses linked by marked paths in a semantic network (Hirst 1987). In fact, clusters capture most of the useful associations found in scripts or semantic networks, but lack many of the disadvantages of using networks. For example, because clusters do not specify what the exact nature of any association is, learning new clusters from previously processed sentences would be fairly straightforward, in contrast to learning new fragments of network. Using clusters also avoids the major problem associated with marker-passing approaches, namely how to prevent the production of stupid paths (or remove them from consideration after they have been produced) (Charniak

1983). The relevant difference is that a cluster is cautious because it must explicitly specify all its elements. A marker passer takes the opposite stance, however, considering all paths up, down, and across the network unless it is explicitly constrained. Thus a marker passer might find the following dubious path from the 'written object' sense of book to the 'part-of-a-plant' sense of leaf:

```
[book made-of paper]
[paper made-from wood]
[tree made-of wood]
[tr e has-part leaf]
```

whereas no cluster would link these entities, unless there had been some prior evidence of a connection. (The recommended solution to the production of such paths by a marker passer is to prevent the passing of marks through certain kinds of nodes [Hirst 1987; Hendler 1987].)

From the lexical entries, the underlying concept hierarchy, and the specialized entries for collocation and clusters just described, a language analyzer can extract the information that establishes preferences among senses. In the next section, we will describe how a semantic interpreter can apply knowledge from such a wide variety of sources.

4. Using Knowledge to Identify Sense Preferences

There is a wide variety of information about which sense is the correct one, and the challenge is to decide when and how to use this information. The danger of a combinatorial explosion of possibilities makes it advantageous to try to resolve ambiguities as early as possible. Indeed, efficient preprocessing of texts can elicit a number of cues for word senses, set up preferences, and help control the parse. Then, the parse and semantic interpretation of the text will provide the cues necessary to complete the task of resolution.

Without actually parsing a text, a preprocessor can identify for each word its morphology,² its syntactic tag or tags,³ and whether it is part of a collocation; for each sense, it can identify whether the sense is preferred or deprecated and whether it is supported by a cluster. These properties are all either retrievable directly from a knowledge base or computable from short sequences of words. To identify whether the input satisfies the expectations created by syntactic cues or whether it satisfies role-related expectations, the system must first perform some syntactic analysis of the input. Identifying these properties must come after parsing, because recognizing them requires both the structural cues provided by parsing and a semantic analysis of the text.

In our system, processing occurs in three phases: morphology, preprocessing, and parsing and semantic interpretation. (See Figure 14.) Analysis of a text begins with the identification of the morphological features of each word and the retrieval of the (core) senses of each word. Then, the input passes through a special preprocessor that identifies parse-independent semantic preferences (i.e., syntactic tags, collocations, and clusters) and makes a preliminary selection of word senses. This selection process eliminates those core senses that are obviously inappropriate and triggers certain

2 This is at least true for English, although whether it is possible for morphologically complex or agglutinative languages such as Finnish remains to be seen.

3 A similar caveat applies here. (See Church [1988] or Zernik [1990] for statistical approaches to tagging English words.)

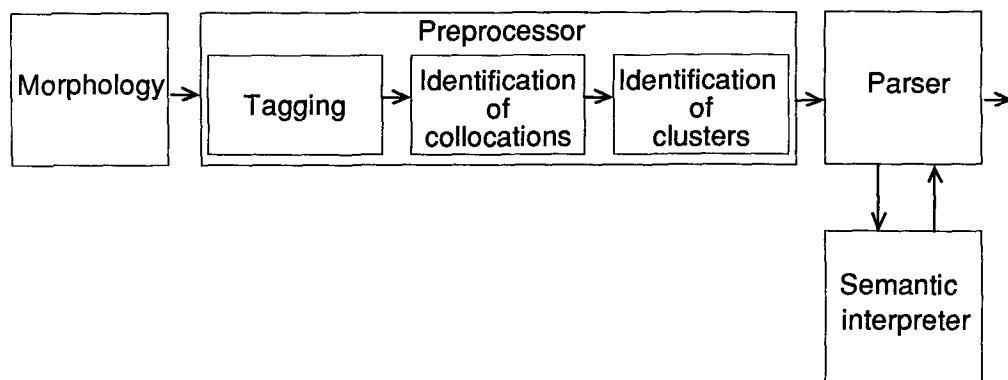


Figure 14
The system architecture.

specialized senses. In the third phase, TRUMP attempts to parse the input and at the same time produce a “preferred” semantic interpretation for it. Since the preferred interpretation also fixes the preferred sense of each word, it is at this point that the text can be given semantic tags, thus allowing sense-based information retrieval.

In the next few subsections we will describe in greater detail the processes that enable the system to identify semantic preferences: morphological analysis, tagging, collocation identification, cluster matching, and semantic interpretation. Afterward we will discuss how the system combines the preferences it identifies.

4.1 Morphological Analysis and Lexical Retrieval

The first step in processing an input text is to determine the root, syntactic features, and affixes of each word. This information is necessary both for retrieving the word’s lexical entries and for the syntactic tagging of the text during preprocessing. Morphological analysis not only reduces the number of words and senses that must be in the lexicon, but it also enables a system to make reasonable guesses about the syntactic and semantic identity of unknown words so that they do not prevent parsing (see Rau, Jacobs, and Zernik 1989). Once morphological analysis of a word is complete, the system retrieves (or derives) the corresponding senses and establishes initial semantic preferences for the primary senses. For example, by default, the sense of *agree* meaning ‘to concur’ (*agree1*) is preferred over its other senses. The lexical entry for *agree* marks this preference by giving it :TYPE *primary* (see Figure 15). The entry also says that derivations (listed in the :S-DERIV field) *agree1+ment* and *agree2+able* are preferred, derivations *agree1+able* and *agree3+ment* are deprecated, and all other sense-affix combinations (excepting inflections) have been disallowed.

During morphological analysis, the system retrieves only the most general senses. It waits until the preprocessor or the parser identifies supporting evidence before it retrieves word senses specific to a context, such as a domain, a situation, or a collocation. In most cases this approach helps reduce the amount of ambiguity. The approach is compatible with evidence discussed by Simpson and Burgess (1988) that

```

( agree
  :POS verb
  :G-DERIV nil
  :SENSES
  (( agree1
    :SYNTAX (one-obj no-obj thatcomp comp subj-equi)
    :EXAMPLE (she agrees with me * they agreed to use force *
              they agreed on 3 percent * they agreed that he was right *
              I agree it is true)
    :TYPE *primary*
    :PAR (c-agreeing)
    :ASSOC (concur believe)
    :S-DERIV ((-ment preferred noun tr_act tr_object tr_result)
              (-able secondary adj tr_ability))
  )
  ( agree2
    :SYNTAX (one-obj)
    :EXAMPLE (winter agrees with me)
    :TYPE *secondary*
    :PAR (c-abstract-relation)
    :ASSOC (benefit)
    :S-DERIV ((-able preferred adj tr_ability))
  )
  ( agree3
    :SYNTAX (no-obj)
    :EXAMPLE (the two accounts do not agree)
    :TYPE *secondary*
    :PAR (c-equivalence-rel)
    :ASSOC (correspond)
    :S-DERIV ((-ment secondary noun tr_state))
  )
)
)
)

```

Figure 15
The lexical entry for the verb *agree*.

“multiple meanings are activated in frequency-coded order” and that low-frequency senses are handled by a second retrieval process that accumulates evidence for those senses and activates them as necessary.

4.2 Tagging

Once the system determines the morphological analysis of each word, the next step in preprocessing is to try to determine the correct part of speech for the word. Our system uses a tagging program, written by Uri Zernik (1990), that takes information about the root, affix, and possible syntactic category for each word and applies stochastic techniques to select a syntactic tag for each word. Stochastic taggers look at small groups of words and pick the most likely assignment of tags, determined by the frequency of alternative syntactic patterns in similar texts. Although it may not be possible to completely disambiguate all words prior to parsing, approaches based on

stochastic information have been quite successful (Church 1988; Garside, Leech, and Sampson 1987; de Marcken 1990).⁴

To allow for the fact that the tagger may err, as part of the tagging process the system makes a second pass through the text to remove some systematic errors that result from biases common to statistical approaches. For example, they tend to prefer modifiers over nouns and nouns over verbs; for instance, in Example 5, the tagger erroneously marks the word *need* as a noun.

Example 5

You really need the Campbell Soups of the world to be interested in your magazine.

In this second pass, the system applies a few rules derived from our grammar and resets the tags where necessary. For example, to correct for the noun versus verb overgeneralization, whenever a word that can be either a noun or a verb gets tagged as just a noun, the corrector lets it remain ambiguous unless it is immediately preceded by a determiner (a good clue for nouns), or it is immediately preceded by a plural noun or a preposition, or is immediately followed by a determiner (three clues that suggest a word may be a verb). The system is able to correct for all the systematic errors we have identified thus far using just nine rules of this sort.

After tagging, the preprocessor eliminates all senses corresponding to unselected parts of speech.

4.3 Identification of Collocations

Following the syntactic filtering of senses, TRUMP's preprocessor identifies collocations and establishes semantic preferences for the senses associated with them. In this stage of preprocessing, the system recognizes the following types of collocations:

- verb+particle pairs such as *take on*;
- verb+preposition pairs such as *invest in*;
- verb+particle+preposition combinations such as *break in on*;
- verb+complement clauses such as *take a bath*, their passives, as in *actions were taken*, and hyphenated nominals, such as *profit-taking*;
- compound noun phrases such as *investment bank*.

To recognize a collocation, the preprocessor relies on a set of simple patterns, which match the general syntactic context in which the collocation occurs. For example, the system recognizes the collocation "take profit" found in Example 6 with the pattern (TAKE (DET) PROFIT).

Example 6

A number of stocks that have spearheaded the market's recent rally bore the brunt of isolated profit-taking Tuesday.

The preprocessor's strategy for locating a collocation is to first scan the text for trigger words, and if it finds the necessary triggers, then to try to match the complete pattern. (Triggers typically correspond to the phrasal head of a collocation, but for

⁴ Magerman and Marcus (1990) do complete stochastic N-gram parsing.

more complex patterns, such as verb-complement clauses, both parts of the collocation must be present.) The system's matching procedures allow for punctuation and verb-complement inversion.

If the triggers are found and the match is successful, the preprocessor has a choice of subsequent actions, depending on how cautious it is supposed to be. In its *aggressive mode*, it updates the representations of the matched words, adding any triggered senses and preferences for the collocated senses. It also deletes any unsupported, deprecated senses. In its *cautious mode*, it just adds the word senses associated with the pattern to a dynamic store. Once stored, these senses are then available for the parser to use after it verifies the syntactic constraints of the collocation; if it is successful, it will add preferences for the appropriate senses. Early identification of triggered senses enables the system to use them for cluster matching in the next stage.

4.4 Identification of Clusters

After the syntactic filtering of senses and the activation of senses triggered by collocations, the next step of preprocessing identifies preferences for senses that invoke currently active clusters (see Section 3.4). A cluster is *active* if it contains any of the senses under consideration for other words in the current paragraph. The system may also activate certain clusters to represent the general topic of the text.

The preprocessor's strategy for assessing cluster-based preferences is to take the set of cluster names invoked by each sense of each content word in the sentence and locate all intersections between it and the names of other active clusters. (For purposes of cluster matching, the sense list for each word will include all the special and noncompositional senses activated during the previous stage of preprocessing, as well as any domain-specific senses that are not yet active.) For each intersection the preprocessor finds, it adds preferences for the senses that are supported by the cluster match. Then, the preprocessor activates any previously inactive senses it found to be supported by a cluster match. This triggering of senses on the basis of conceptual context forms the final step of the preprocessing phase.

4.5 Semantic Interpretation

Once preprocessing is complete, the parsing phase begins. In this phase, TRUMP attempts to build syntactic structures, while calling on the semantic interpreter to build and rate alternative interpretations for each structure proposed. These semantic evaluations then guide the parser's evaluation of syntactic structures. They may also influence the actual progression of the parse. For example, if a structure is found to have incoherent semantics, the parser immediately eliminates it (and all structures that might contain it) from further consideration. Also, whenever the semantics of a parse becomes sufficiently better than that of its competitors, the system prunes the semantically inferior parses, reducing the number of ambiguities even further.⁵

As suggested above, the system builds semantic interpretations incrementally. For each proposed combination of syntactic structures, there is a corresponding combination of semantic structures. It is the job of the semantic interpreter to identify the possible relations that link the structures being combined, identify the preferences associated with each possible combination of head, role (relation), and filler (the argument or modifier), and then rank competing semantic interpretations.

⁵ A similar approach has been taken by Gibson (1990) and is supported by the psychological experiments of Kurtzman (1984).

For each proposed combination, knowledge sources may contribute the following preferences:

- preferences directly associated with the head or the filler, determined recursively from their components, beginning with preferences identified during preprocessing.
- preferences associated with syntactic cues, such as the satisfaction of restrictions listed in the lexicon. For example, a word may allow only modifiers of a particular syntactic form, or a modifier may modify only a certain syntactic form. (For example, the sense meaning ‘to care for,’ in *She tends plants* or *She tends to plants* occurs with an NP or PP object, whereas the sense of *tend* meaning ‘to have a tendency’ as in *She tends to lose things* requires a clausal object.)
- preferences associated with the semantic “fit” between any two of the head, the role, and the filler, for example:

filler and role e.g., foods make good fillers for the PATIENT role of eating activities;

filler and head e.g., colors make good modifiers of physical objects;

head and role e.g., monetary objects expect to be qualified by some QUANTITY.

The conceptual hierarchy and the lexicon contain the information that encodes these preferences.

- preferences triggered by reference resolution. (Currently, our system does not make use of these preferences, but see Crain and Steedman [1985]; Altmann and Steedman [1988]; Hirst [1987].)

How the semantic interpreter combines these preferences is the subject of the next section.

5. Combining Preferences to Select Senses

Given the number of preference cues available for discriminating word senses, an understander must face the question of what to do if they conflict. For example, in the sentence *Mary took a picture to Bob*, the fact that photography does not normally have a destination (negative role-related information) should override the support for the ‘photograph’ interpretation of *took a picture* given by collocation analysis. A particular source of information may also support more than one possible interpretation, but to different degrees. For example, *cigarette filter* may correspond either to something that filters out cigarettes or to something that is part of a cigarette, but the latter relation is more likely. Our strategy for combining the preferences described in the preceding sections is to rate most highly the sense with the strongest combination of supporting cues. The system assigns each preference cue a strength, an integer value between +10 and -10, and then sums these strengths to find the sense with the highest rating.

The strength of a particular cue depends on its type and on the degree to which the expectations underlying it are satisfied. For cues that are polar — for example, a sense is either low or high frequency — a value must be chosen experimentally, depending on the strength of the cue compared with others. For example, the system assigns frequency information (the primary–secondary distinction) a score close to

zero because this information tends to be significant only when other preferences are inconclusive. For cues that have an inherent extent — for example, the conceptual category specified by a role preference subsumes a set of elements that can be counted — the cue strength is a function of the magnitude of the extent, that is, its *specificity*.

TRUMP's specificity function maps the number of elements subsumed by the concept onto the range 0 to +10. The function assigns concepts with few members a high value and concepts with many members a low value. For example, the concept *c-object*, which subsumes roughly half the knowledge base, has a low specificity value (1). In contrast, the concept *noun_hammer1*, which subsumes only a single entity, has a high specificity value (10). Concept strength is inversely proportional to concept size because a preference for a very general (large) concept often indicates that either there is no strong expectation at all or there is a gap in the system's knowledge. In either case, a concept that subsumes only a few senses is stronger information than a concept that subsumes more. The preference score for a complex concept, formed by combining simpler concepts with the connectives AND, OR, and NOT, is a function of the number of senses subsumed by both, either, or neither concept, respectively. Similarly, the score for a cluster is the specificity of that cluster (as defined in Section 3.4). (If a sense belongs to more than one active cluster, then only the most specific one is considered.) The exact details of the function (i.e., the range of magnitudes corresponding to each specificity class) necessarily depend on the size and organization of one's concept hierarchy. For example, one would assign specificity value 1 to any concept with more members than any immediate specialization of the most abstract concept.

When a preference cue matches the input, the cue strength is its specificity value; when a concept fails to match the input, the strength is a negative value whose magnitude is usually the specificity of the concept, but it is not always this straightforward. Rating the evidence associated with a preference failure is a subtle problem, because there are different types of preference failure to take into account. Failure to meet a general preference is always significant, whereas failure to meet a very specific preference is only strong information when a slight relaxation of the preference does not eliminate the failure. This presents a bit of a paradox: the greater the specificity of a concept, the more information there is about it, but the less information there may be about a corresponding preference. The paradox arises because the failure of a very specific preference introduces significant uncertainty as to why the preference failed.

Failing to meet a very general preference is always strong information because, in practice, the purpose of such preferences is to eliminate the grossly inappropriate — such as trying to use a relation with a physical object when it should only be applied to events. The specificity function in this case returns a value whose magnitude is the same as the specificity of the *complement* of the concept (i.e., the positive specificity less the maximum specificity, 10.) The result is a negative number whose absolute value is greater than it would be by default. For example, if a preference is for the concept *c-object*, which has a positive specificity of 1, and this concept fails to match the input, then the preference value for the cue will be -9 .

On the other hand, a very specific preference usually pinpoints the expected entity, i.e., the dead giveaway pairings of role and filler. Thus, it is quite common for these preferences to *overspecify* the underlying constraint; for example, *cut* may expect a tool as an INSTRUMENT, but almost any physical object will suffice. When a slight relaxation of the preference is satisfiable, a system should take the cautious route, and assume it has a case of overspecification and is at worst a weak failure. Again, the specificity function returns a negative value with magnitude equivalent to the specificity of the complement of the concept, but this time the result will be a negative number whose

absolute value is less than it would be by default. When this approach fails, a system can safely assume that the entity under consideration is “obviously inappropriate” for a relatively strong expectation, and return the default value. The default value for a concept that is neither especially general nor specific and that fails to match the input is just -1 times the positive specificity of the concept.

The strategy of favoring the most specific information has several advantages. This approach best addresses the concerns of an expanding knowledge base where one must be concerned not only with competition between preferences but also with the inevitable gaps in knowledge. Generally, the more specific information there is, the more complete, and hence more trustworthy, the information is. Thus, when there is a clear semantic distinction between the senses and the system has the information necessary to identify it, a clear distinction usually emerges in the ratings. When there is no strong semantic distinction, or there is very little information, preference scores are usually very close, so that the parser must fall back on syntactic preferences, such as Right Association. This result provides a simple, sensible means of balancing syntactic and semantic preferences.

To see how the cue strengths of frequency information, morphological preferences, collocations, clusters, syntactic preferences, and role-related preferences interact with one another to produce the final ranking of senses, consider the problem of deciding the correct sense of *reached* in Example 1 (repeated below):

Example 1

The agreement reached by the state and the EPA provides for the safe storage of the waste.

According to the system’s lexicon, *reached* has four possible verb senses:

- *reach1*, as in *reach a destination*, which has conceptual parents *c-dest-occur* (“destination occurrence”) and *c-arriving*;
- *reach2*, as in *reach for a cookie*, which has conceptual parent *c-bodypart-action*;
- *reach3*, as in *reach her by telephone*, which has conceptual parent *c-comm-event* (“communication event”); and
- *reach4*, as in *reach a conclusion*, which has conceptual parent *c-cause-to-event-change*.

Figure 16 shows a tabulation of cue strengths for each of these interpretations of *reach* in Example 1, when just information in the VP *reached by the state and the EPA* is considered. The sense *reach3* has the highest total score. From the table, we see that, at this point in the parse, the only strong source of preferences is the role information (line 6 of Figure 16). The derivation of these numbers is shown in Figures 17, 18, and 19, which list the role preferences associated with the possible interpretations of the preposition *by* for *reach3*, and its two nearest competitors, *reach1* and *reach4*. Together, the data in the tables reveal the following sources of preference strength:

- The ‘arrival’ sense (*reach1*) gains support from the fact that there is a sense of *by* meaning AGENT, which is a role that arrivals expect (line 3 of column 3 of Figure 17), and *the state* and *the EPA* make reasonably good agents (line 5 of column 3 of Figure 17).

Cue Type	Cue Strength			
	reach1 c-dest-occur	reach2 c-bodypart-action	reach3 c-comm-event	reach4 c-cause-to-event-change
Frequency	1	1	-1	1
Morphology	0	0	0	0
Collocation	0	0	0	0
Cluster	0	0	0	0
Syntax	0	0	0	0
Roles	41	38	46	41
Total	42	39	45	42

Figure 16
Score tabulations for *reached* in the VP.

Preference Type	reach1 Role Preference Strength			
	by1	by3	by4	by5
	SPATIAL-PROXIMITY	DIRECTION	AGENT	INSTRUMENT
Relation-Filler	0	0	0	-2
Filler-Holder	0	0	0	0
Relation-Holder	0	0	5	0
Syntax	1	1	1	1
Strength of PP	37	35	35	35
Total	38	36	41	34

Figure 17
Role-related preferences of reach1 for the preposition *by*.

- The ‘communication’ sense (reach3) gains support from the fact that there is a sense of *by* corresponding to the expected role COMMUNICATOR (line 3 of column 3 of Figure 18) and *the state* and *the EPA* make very good agents of communication events (communicators), in particular (line 1 of column 3 of Figure 18), as well as being good agents in general (line 5 of column 3 of Figure 18); however, reach3 is disfavored by frequency information (line 1 of column 3 of Figure 16).
- The ‘event change’ (conclude) sense (reach4) gains support from the fact that there is a sense of *by* corresponding to the expected role CAUSE (line 3 of column 3 of Figure 19) and from the fact that *the state* and *the EPA* make good agents (line 5 of column 3 of Figure 19).

Although the system favors the ‘communication’ sense of *reach* in the VP, for the final result, it must balance this information with that provided by the relationship between *agreement* and the verb phrase. By the end of the parse, the ‘event-change’ sense comes to take precedence:

- The system completely eliminates the ‘destination’ sense from consideration because it is significantly weaker than all its competitors.

reach3				
Preference Type	Role Preference Strength			
	by1	by3	by4	by5
	SPATIAL-PROXIMITY	DIRECTION	COMMUNICATOR	INSTRUMENT
Relation-Filler	0	0	5	0
Filler-Holder	0	0	0	0
Relation-Holder	0	0	5	0
Syntax	1	1	1	1
Strength of PP	37	35	35	35
Total	38	36	46	36

Figure 18 Role-related preferences of reach3 for the preposition *by*.

reach4				
Preference Type	Role Preference Strength			
	by1	by3	by4	by5
	SPATIAL-PROXIMITY	DIRECTION	CAUSE (AGENT)	INSTRUMENT
Relation-Filler	0	0	0	0
Filler-Holder	0	0	0	0
Relation-Holder	0	0	5	0
Syntax	1	1	1	1
Strength of PP	37	35	35	35
Total	38	36	41	36

Figure 19 Role-related preferences of reach4 for the preposition *by*.

The main cause of this weakness is that (in our system) the role that agreement would fill, DESTINATION, has no special preference for being associated with a *c-dest-event* — many events allow a DESTINATION role.

- The ‘communication’ sense loses favor because it does not gain much support from having agreement as either PATIENT or RECIPIENT. The final score of this sense is 52.
- The ‘event-change’ sense gains support from having *agreement* fill its AFFECTED role, enough that the final strength of the ‘event-change’ sense, 55, ultimately surpasses the final strength of the ‘communication’ sense.

By summing the cue strengths of each possible interpretation in this way and selecting the one with the highest total score, the system decides which sense is the “correct” one for the context. The strengths of individual components of each interpretation contribute to, but do not determine, the strength of the final interpretation, because there are also strengths associated with how well the individual components fit together. No additional weights are necessary, because the specificity values the system uses are a direct measure of strength.

6. Results and Discussion

Our goal has been a natural language system that can effectively analyze an arbitrary input at least to the level of word sense tagging. Although we have not yet fully accomplished this goal, our results are quite encouraging. Using a lexicon of approximately 10,000 roots and 10,000 derivations, the system shows excellent lexical and morphological coverage. When tested on a sample of 25,000 words of text from the *Wall Street Journal*, the system covered 98% of non-proper noun, non-abbreviated word occurrences (and 91% of *all* words). Twelve percent of the senses the system selected were derivatives.

The semantic interpreter is able to discriminate senses even when the parser cannot produce a single correct parse. Figure 20 gives an example of the sense tagging that the system gives to the following segment of *Wall Street Journal* text:

Example 7

The network also is changing its halftime show to include viewer participation, in an attempt to hold on to its audience through halftime and into the second halves of games. One show will ask viewers to vote on their favorite all-time players through telephone polls.

Each word is tagged with its part of speech and sense number along with a parent concept. For example, the tag [changing verb_3 (c-replacing)] shows that the input word is *changing*, the preferred sense is number 3 of the verb, and this sense falls under the concept *c-replacing* in the hierarchy. This tagging was produced even though the parser was unable to construct a complete and correct syntactic representation of the text. In fact, when tested on the *Wall Street Journal* texts (for which there has been no adaptation or customization aside from processing by a company-name recognizer [Rau 1991]), the system rarely produces a single correct parse; however, the partial parses produced generally cover most of the text at the clause level. Since most semantic preferences appear at this level (and those that do not, do not depend on syntactic analysis), the results of this tagging are encouraging.

This example also shows some of the limitations of our system in practice. The system is unable to recognize the collocation "hold on to" in the first sentence, because it lacks a pattern for it. The system also lacks patterns for the collocations "vote on" and "all-time players" that occur in the second sentence, and as a result, mistakenly tags *on* as *c-temporal-proximity-rel* rather than something more appropriate, such as *c-purpose-rel*. These difficulties point out the need for even more knowledge.

It is encouraging to note that, even if our encoding scheme is not entirely "correct" according to human intuition, as long as it is consistent, in theory it should lead to capabilities that are no worse, with zero customization, than word-based methods for information retrieval. However, having access to sense tags allows for easy improvement by more knowledge-intensive methods. Although this theory is still untested, there is some preliminary evidence that word sense tagging can improve information retrieval system performance (Krovetz 1989).

To date we have been unable to get a meaningful quantitative assessment of the accuracy of the system's sense tagging. We made an unsuccessful attempt at evaluating the accuracy of sense-tagging over a corpus. First, we discovered that a human "expert" had great difficulty identifying each sense, and that this task was far more tedious than manual part-of-speech tagging or bracketing. Second, we questioned what we would learn from the evaluation of these partial results, and have since turned our

```

[the det_1 (c-definite-qual) ]
[network noun_2 (c-entertainment-obj c-business-org c-system) ]
[also adv_1 (c-numeric-qual) ]
[is *aux* ]
[changing verb_3 (c-replacing) ]
[its ppnoun_1 (c-obj) ]
[halftime noun_1 (c-entity) ]
[show c-act-of-verb_show1 (c-manifesting) ]
[to *infl* ]
[include verb_2 (c-grouping) ]
[viewer c-verb_view2-er (c-entity) ]
[participation c-result-of-being-verb_participate1 (c-causal-state) ]
[*comma* *punct* ]
[in prep_27 (c-group-part) ]
[an det_1 (c-definite-qual) ]
[attempt c-act-of-verb_attempt1 (c-attempting) ]
[to *infl* ]
[hold verb_4 (c-positioning) ]
[on adv_1 (c-range-qual c-continuity-qual) ]
[to prep_1 (c-destination-rel) ]
[its ppnoun_1 (c-obj) ]
[audience noun_1 (c-human-group) ]
[through prep_1 (c-course-rel) ]
[halftime noun_1 (c-entity) ]
[and coordconj_1 (c-conjunction) ]
[into prep_5 (c-engage-in) ]
[the det_1 (c-definite-qual) ]
[second c-numword_two1-th (c-order-qual) ]
[halves noun_1 (c-portion-part) ]
[of prep_8 (c-stateobject-rel) ]
[games noun_1 (c-recreation-obj) ]
[*period* *punct* ]

[one noun_1 (c-entity) ]
[show c-act-of-verb_show1 (c-manifesting) ]
[will *aux* ]
[ask verb_2 (c-asking) ]
[viewers c-verb_view2-er (c-entity) ]
[to *infl* ]
[vote verb_1 (c-selecting) ]
[on prep_4 (c-temporal-proximity-rel) ]
[their ppnoun_1 (c-obj) ]
[favorite adj_1 (c-importance-qual c-superiority-qual) ]
[all det_1 (c-quantifier) ]
[*hyphen* *punct* ]
[time noun_1 (c-indef-time-period) ]
[play1-er c-verb_play1-er (c-entity) ]
[through prep_1 (c-course-rel) ]
[telephone noun_1 (c-machine) ]
[polls c-act-of-verb_poll1 (c-asking) ]
[*period* *punct* ]

```

Figure 20

A sample sense coding.

attention back to evaluating the system with respect to some task, such as information retrieval.

Improving the quality of our sense tagging requires a fair amount of straightforward but time-consuming work. This needed work includes filling a number of

gaps in our knowledge sources. For example, the system needs much more information about role-related preferences and specialized semantic contexts. At present all this information is collected and coded by hand, although recent work by Ravin (1990) and Dahlgren, McDowell, and Stabler (1989) suggests that the collection of role-related information may be automatable.

Our next step is to evaluate the effect of text coding on an information retrieval task, by applying traditional term-weighted statistical retrieval methods to the recorded text. One intriguing aspect of this approach is that errors in distinguishing sense preferences should not be too costly in this task, so long as the program is fairly consistent in its disambiguation of terms in both the source texts and the input queries.

7. Conclusion

Having access to a large amount of information and being able to use it effectively are essential for understanding unrestricted texts, such as newspaper articles. We have developed a substantial knowledge base for text processing, including a word sense-based lexicon that contains both core senses and dynamically triggered entries. We have also created a number of concept-cluster definitions describing common semantic contexts and a conceptual hierarchy that acts as a sense-disambiguated thesaurus.

Our approach to word sense discrimination uses information drawn from the knowledge base and the structure of the text, combining the strongest, most obvious sense preferences created by syntactic tags, word frequencies, collocations, semantic context (clusters), selectional restrictions, and syntactic cues. To apply this information most efficiently, the approach introduces a preprocessing phase that uses preference information available prior to parsing to eliminate some of the lexical ambiguity and establish baseline preferences. Then, during parsing, the system combines the baseline preferences with preferences created by selectional restrictions and syntactic cues to identify preferred interpretations. The preference combination mechanism of the system uses dynamic measures of strength based on specificity, rather than relying on some fixed, ordered set of rules.

There are some encouraging results from applying the system to sense tagging of arbitrary text. We expect to evaluate our approach on tasks in information retrieval, and, later, machine translation, to determine the likelihood of achieving substantive improvements through sense-based semantic analysis.

Acknowledgments

I am grateful to Paul Jacobs for his comments and his encouragement of my work on natural language processing at GE; to George Krupka for helping me integrate my work with TRUMP; and for continuing to improve the system; to Graeme Hirst for his many comments and suggestions on this article; and to Jan Wiebe and Evan Steeg for their comments on earlier drafts. I acknowledge the financial support of the General Electric Company, the University of Toronto, and the Natural Sciences and Engineering Research Council of Canada.

References

Altmann, Gerry, and Steedman, Mark (1988). "Interaction with context during

human sentence processing." *Cognition* 30(3): 191-238.

Charniak, Eugene (1983). "Passing markers: A theory of contextual influence in language comprehension." *Cognitive Science* 7(3): 171-190.

Church, Kenneth W. (1988). "A stochastic parts program and noun phrase parser for unrestricted text." In *Proceedings, Second Conference on Applied Natural Language Processing*. Austin, Texas, 136-143.

Church, Kenneth W., and Hanks, Patrick (1990). "Word association norms, mutual information, and lexicography." *Computational Linguistics* 16(1): 22-29.

Cowie, Anthony P., and Mackin, Ronald (1975). *Verbs with Prepositions and Particles*. Volume 1 of *Oxford Dictionary of Current*

- Idiomatic English*. Oxford: Oxford University Press.
- Cowie, Anthony P.; Mackin, Ronald; and McCraig, Isabel R. (1983). *Clause and Sentence Idioms*. Volume 2 of *Oxford Dictionary of Current Idiomatic English*. Oxford: Oxford University Press.
- Crain, Stephen, and Steedman, Mark (1985). "On not being led up the garden path: The use of context by the psychological syntax processor." In *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*, edited by D. R. Dowty, L. Karttunen, and A. M. Zwicky, 320–358. Cambridge, England: Cambridge University Press.
- Dahlgren, Kathleen; McDowell, Joyce; and Stabler, Edward (1989). "Knowledge representation for commonsense reasoning with text." *Computational Linguistics* 15(3): 149–170.
- De Marcken, Carl G. (1990). "Parsing the LOB corpus." In *Proceedings, 28th Annual Meeting of the Association for Computational Linguistics*. Pittsburgh, PA, 243–251.
- Dyer, Michael, and Zernik, Uri (1986). "Encoding and acquiring meanings for figurative phrases." In *Proceedings, 24th Annual Meeting of the Association for Computational Linguistics*. New York, NY, 106–111.
- Fodor, Janet Dean (1978). "Parsing strategies and constraints on transformations." *Linguistic Inquiry* 9(3): 427–473.
- Fodor, Janet Dean, and Frazier, Lyn (1980). "Is the human sentence parsing mechanism an ATN?" *Cognition* 8: 417–459.
- Ford, Marilyn; Bresnan, Joan; and Kaplan, Ronald (1982). "A competence-based theory of syntactic closure." In *The Mental Representation of Grammatical Relations*, edited by Joan Bresnan, 727–796. Cambridge: The MIT Press.
- Fox, E.; Nutter, T.; Ahlswede, T.; Evens, M.; and Markowitz, J. (1988). "Building a large thesaurus for information retrieval." In *Proceedings, Second Conference on Applied Natural Language Processing*. Austin, Texas, 101–108.
- Frazier, Lyn, and Rayner, Keith (1990). "Taking on semantic commitments: Processing multiple meanings vs. multiple senses." *Journal of Memory and Language* 29(2): 181–200.
- Garside, Roger; Leech, Geoffrey; and Sampson, Geoffrey (1987). *The Computational Analysis of English: A Corpus-Based Approach*. London: Longman.
- Gibson, Edward (1990). "Memory capacity and sentence processing." In *Proceedings, 28th Annual Meeting of the Association for Computational Linguistics*. Pittsburgh, PA, 39–46.
- Halliday, Michael, and Hasan, Ruqaiya (1976). *Cohesion in English*. London: Longman.
- Hendler, James A. (1987). *Integrating Marker-Passing and Problem Solving*. Norwood, NJ: Lawrence Erlbaum Associates.
- Hirst, Graeme (1987). *Semantic Interpretation and the Resolution of Ambiguity*. Cambridge: Cambridge University Press.
- Jacobs, Paul S. (1986). "Language analysis in not-so-limited domains." In *Proceedings, Fall Joint Computer Conference*. Dallas, TX.
- Jacobs, Paul S. (1987). "A knowledge framework for natural language analysis." In *Proceedings, Tenth International Joint Conference on Artificial Intelligence*. Milan, Italy.
- Jacobs, Paul S. (1989). "TRUMP: A transportable language understanding program." Technical Report CRD89/181, GE Research and Development Center, Schenectady, NY.
- Janssen, Sylvia (1990). "Automatic sense disambiguation with LDOCE: Enriching syntactically analyzed corpora with semantic data." In *Theory and Practice in Corpus Linguistics*, edited by Jan Aarts and Willem Meijs, 105–135. Amsterdam: Rodopi.
- Kimball, John P. (1973). "Seven principles of surface structure parsing in natural language." *Cognition* 2: 15–47.
- Krovetz, Robert (1989). "Lexical acquisition and information retrieval." In *First International Lexical Acquisition Workshop*, edited by Uri Zernik.
- Kurtzman, Howard S. (1984). "Studies in syntactic ambiguity resolution." Doctoral dissertation, Department of Psychology, MIT. Bloomington, IN: Indiana University Linguistics Club.
- Magerman, David M., and Marcus, Mitchell P. (1990). "Parsing a natural language using mutual information statistics." In *AAAI-90 Proceedings, Eighth National Conference on Artificial Intelligence*, Menlo Park, CA: AAAI Press/The MIT Press, 984–989.
- McRoy, Susan W., and Hirst, Graeme (1990). "Race-based syntactic attachment." *Cognitive Science* 14(3): 313–354.
- Morris, Jane (1988). "Lexical cohesion, the thesaurus, and the structure of text." Technical Report CSRI-219, Computer Systems Research Institute, University of Toronto, Toronto.
- Morris, Jane, and Hirst, Graeme (1991).

- "Lexical cohesion computed by thesaural relations as an indicator of the structure of text." *Computational Linguistics* 17(1): 21–48.
- Procter, Paul, editor (1978). *Longman Dictionary of Contemporary English*. Harlow: Longman Group Ltd.
- Rau, Lisa F. (1991). "Extracting company names from text." In *IEEE AI Applications Conference (CAIA)*.
- Rau, Lisa F.; Jacobs, Paul S.; and Zernik, Uri (1989). "Information extraction and text summarization using linguistic knowledge acquisition." *Information Processing and Management* 25(4): 419–428.
- Ravin, Yael (1990). "Disambiguating and interpreting verb definitions." In *Proceedings, 28th Annual Meeting of the Association for Computational Linguistics*. Pittsburgh, PA, 260–267.
- Rayner, Keith; Carlson, Marcia; and Frazier, Lyn (1983). "The interaction of syntax and semantics during sentence processing: Eye movements in the analysis of semantically biased sentences." *Journal of Verbal Learning and Verbal Behavior* 22: 358–374.
- Schank, Roger C., and Abelson, Robert P. (1977). *Scripts, Plans, Goals, and Understanding*. Halsted, NJ: Lawrence Erlbaum.
- Schubert, Lenhart (1986). "Are there preference trade-offs in attachment decisions?" In *Proceedings, National Conference on Artificial Intelligence (AAAI-86)*. Philadelphia, PA, 601–605.
- Shieber, Stuart M. (1983). "Sentence disambiguation by a shift-reduce parsing technique." In *Proceedings, 21st Annual Meeting of the Association for Computational Linguistics*. Cambridge, MA, 113–118.
- Simpson, Greg B., and Burgess, Curt (1988). "Implications of lexical ambiguity resolution for word recognition and comprehension." In *Lexical Ambiguity Resolution*, edited by Steven L. Small, Garrison W. Cottrell, and Michael K. Tanenhaus, 271–288. San Mateo, CA: Morgan Kaufmann Publishers.
- Smadja, Frank A., and McKeown, Kathleen R. (1990). "Automatically extracting and representing collocations for language generation." In *Proceedings, 28th Annual Meeting of the Association for Computational Linguistics*. Pittsburgh, PA, 252–259.
- Stock, Oliviero (1989). "Parsing with flexibility, dynamic strategies, and idioms in mind." *Computational Linguistics* 15(1): 1–18.
- Whittemore, Greg; Ferrara, Kathleen; and Brunner, Hans (1990). "Empirical study of predictive powers of simple attachment schemes for post-modifier prepositional phrases." In *Proceedings, 28th Annual Meeting of the Association for Computational Linguistics*. Pittsburgh, PA, 23–30.
- Wilks, Yorick; Huang, Xiuming; and Fass, Dan (1985). "Syntax, preference, and right attachment." In *Proceedings, Ninth International Joint Conference on Artificial Intelligence*. Los Angeles, CA.
- Zernik, Uri (1990). "Tagging word senses in corpus: The needle in the haystack revisited." Technical Report 90CRD198, GE Research and Development Center, Schenectady, NY. In *Proceedings, AAAI Symposium on Text-Based Intelligent Systems*. Stanford, CA, 25–29.