Processing Dictionary Definitions with Phrasal Pattern Hierarchies

Hiyan Alshawi

University of Cambridge Computer Laboratory
Corn Exchange Street, Cambridge CB2 3QG, England*

This paper shows how dictionary word sense definitions can be analysed by applying a hierarchy of phrasal patterns. An experimental system embodying this mechanism has been implemented for processing definitions from the *Longman Dictionary of Contemporary English.* A property of this dictionary, exploited by the system, is that it uses a restricted vocabulary in its word sense definitions. The structures generated by the experimental system are intended to be used for the classification of new word senses in terms of the senses of words in the restricted vocabulary. Examples illustrating the output generated are presented, and some qualitative performance results and problems that were encountered are discussed. The analysis process applies successively more specific phrasal analysis rules as determined by a hierarchy of patterns in which less specific patterns dominate more specific ones. This ensures that reasonable incomplete analyses of the definitions are produced when more complete analyses are not possible, resulting in a relatively robust analysis mechanism. Thus the work reported addresses two robustness problems faced by current experimental natural language processing systems: coping with an incomplete lexicon and with incomplete knowledge of phrasal constructions.

Introduction

A major factor contributing to the lack of robustness of experimental natural language understanding systems is the small number of words in the experimental semantic dictionaries used by these systems. For example "missing vocabulary" is cited as the most frequent cause of errors for the FRUMP system (DeJong 1979), a system designed to achieve a high degree of robustness. The problem does not disappear when dealing with limited discourse domains of the type encountered in database query and expert system interfaces. This is because of the large number of synonyms and specialized words that can occur, and because of the difficulty of delimiting discourse domains exactly.

A different problem faced by designers of natural language understanding systems is how to provide for graceful failure of sentence analysis. There is thus the need to produce reasonable incomplete interpretations of sentences when complete analyses are not possible. This situation can occur because of gaps in the grammatical knowledge of the system or because the system is faced with extragrammatical input. This paper shows how a possible solution to this partial analysis problem can be applied to the vocabulary problem in the context of large machine readable dictionaries.

More specifically, we will see how word sense definitions from the *Longman Dictionary of Contemporary English* (Procter, 1978 — henceforth LDOCE) are processed by a phrasal analyser that applies successively more specific phrasal analysis rules. The aim of this analysis is to provide sufficient semantic information to enable a system carrying out a language processing application to cope with occurrences of unknown words.

Both the problem of coping with new words and the problem of robust phrasal analysis can be thought of as instances of a more general natural language interpretation problem. This is the problem of coping with incomplete knowledge of language use; lexical knowledge in the first case and knowledge of phrasal structure in the second. The unavoidable incompleteness of the knowledge of language use available to a language processing system means that trying to achieve robust natural language processing involves developing effective mechanisms for dealing with this problem. The research

*Author's present address: SRI International, Cambridge Computer Science Research Centre, Millers Yard, Mill Lane, Cambridge CB2 1RQ, England.

0362-613X/ 87 /030195-202$03.00

reported in this paper is intended to be a contribution to this development effort.

The next two sections will discuss the kind of output that may be produced from processing dictionary definitions and give examples of the results of processing LDOCE definitions produced by an implemented definition analyser. Some problems that were encountered are then discussed. Later sections motivate and explain the basic analysis algorithm, and then describe and illustrate details of analysis and structure building rules. Finally some remarks are made about the performance of the current implementation and necessary further research.

Definition analysis

There are various possibilities for the kind of structures useful for language understanding that may be derived from dictionary definitions. These include meaning postulates (Carnap, 1952) expressed in some logic; constraints or 'semantic formulae' based on semantic primitives (Katz and Fodor, 1963, Wilks, 1975); and structures carrying information enabling the classification of the new word sense with respect to an existing classification of entities in a discourse domain. The structures produced by the implemented definition analyser belong to this last type, and examples of these structures are given later.

The dictionary being used in this work, LDOCE, has features that make it particularly suitable for definition analysis. Thus many LDOCE word sense entries contain additional semantic information that could be combined, or used in conjunction with, the structures produced from processing word sense definition texts. This information is available as 'box codes' that give selectional restrictions, and 'subject codes' that indicate typical discourse domain usage of word senses (these codes occur in the machine-readable version of the dictionary, but not in the printed form). The suitability of LDOCE for work in computational linguistics has been analysed in detail by Michiels (1982). For the purpose of the work reported here, the most important property of LDOCE is the use of a restricted definition vocabulary of around 2000 words. Further, an important restriction imposed on LDOCE lexicographers is that only the 'central' senses of these words should occur in definition texts. Some ways in which the definitions diverge from a strict interpretation of this rule are discussed later. It should be remarked here that the LDOCE restricted definition vocabulary has more in common with a 'basic English' vocabulary than a set of semantic primitives. (A list of the words in the restricted definition vocabulary is given in an appendix to the published version (Procter, 1978) of the dictionary.)

If the output of processing LDOCE definitions was in the form of meaning postulates, then the logic expressions produced would have a new symbol for the word sense being defined along with symbols corresponding to the senses of words in the definition vocabulary. Similarly, producing semantic primitive formulae would involve building new formulae by putting together formulae corresponding to the word senses of the definition vocabulary.

For the third possible form of output listed earlier, we need a (hand-coded) classification of the central senses of the definition vocabulary together with a classification of concepts in the particular domain of discourse in terms of these word senses. The descriptions of implementations by Bobrow and Webber (1980), Mark (1981), and Alshawi (1987), show how such a classification can be organized and used during text processing. The LDOCE definition for a new word sense is processed using the mechanism described in this paper in order to extract sufficient information for including the new word sense in such a classification. A natural language processing application that depended on a classification of concepts in the discourse domain should then be able to carry out its application task despite the occurrence of a new word in an input sentence.

Extracting the information necessary for classification will of course include locating superordinates in the definitions (which define the so called "ISA" relation) as is done in the work reported by Amsler (1981) and Calzolari (1984). However, this previous work suggests that achieving further semantic precision in a classification process requires making use of other information present in the definition (such as modifiers and predications). Examples of extracting this sort of information are presented in the next section.

This way of dealing with unknown words in language processing applications still requires good solutions to the problem of choosing between alternative possible word senses (Walker and Amsler (1986) have used the LDOCE subject codes for this purpose) and to the problems involved in the classification process (see Schmolze and Lipkis, 1983). Nonetheless, providing a mechanism, as described in this paper, for extracting the information required by the classification process is a necessary first step for this approach to handling unknown words.

Dictionaries vary in the level of detail provided by their semantic definitions, and, in general, producing a meaning postulate for understanding a sentence, or incorporating a word sense into a discourse domain classification, can often be done without making use of all the detail provided by the dictionary definition. Even only being able to locate the 'semantic head' (i.e. the main superordinate term) of a definition can be useful to a language processing application. This is fortunate since providing complete analyses of arbitrary dictionary definitions is beyond the current state of the art in computational linguistics. It is therefore reasonable to derive and make use of partial analyses of dictionary definitions when complete analyses are not possible. This is the approach taken in the implemented analysis system. (For certain text processing systems, for example precision information retrieval systems, such an approach may not be acceptable.)

Analysis examples

I will refer to the information derived for classifying word senses simply as 'semantic structures'; this rather vague term being chosen because these structures are not viewed as having formal semantic status, but only as data structures containing information relevant to the classification process (or perhaps some other semantic process). Roughly speaking, these structures have some properties of a linguistic analysis of definition texts and some properties of a semantic definition of word sense concepts; their gross syntactic form is that of nested feature lists.

The semantic structures are derived from various types of modifiers and relative clauses present in word sense definitions as well as the semantic head of the definition, if this is substantive. The syntactic category under which senses are grouped in the dictionary is important, in particular, for locating the semantic head of a definition and, more generally, for determining which analysis rules are applicable to the definition. The details of the analysis process are explained in a later section. Illustrative examples of the structures produced from analysing the main categories of definitions currently handled by the implemented system — noun, verb, adjective and adverb definitions — are now given. Oddities in the semantic structures are often due to peculiarities of the current analysis grammar and output format, and I would not wish to argue for their correctness, especially in view of the problems discussed later.

The following are examples of noun sense definitions together with the semantic structures derived from them. The words under which these examples occur in the dictionary are shown underlined. (The analysis system retrieves definitions from a 'lispified' version of the LDOCE type-setting tape, for example items preceded by an asterisk are Lisp atoms corresponding to font control characters present on the type-setting tape (see Alshawi, Boguraev, and Briscoe, 1985).)

(launch)

(a large usu. motor-driven boat used for carrying people on rivers, lakes, harbours, etc.)

```
((CLASS BOAT) (PROPERTIES (LARGE))
  (PURPOSE
    (PREDICATION (CLASS CARRY) (OBJECT
      PEOPLE))))
```

(mug)

(*46 BrE infml *44 a foolish person who is easily deceived *44 *63 see also *CA MUG'S GAME)

```
((CLASS PERSON) (PROPERTIES (FOOLISH))
  (PREDICATION (OBJECT-OF ((CLASS
    DECEIVE)))))
```

(hornbeam)

(a type of small tree with hard wood, sometimes used in *CA HEDGE *CB *46 s)

```
((CLASS TREE) (COLLECTIVE TYPE) (PROPER-
  TIES (SMALL))
   (HAS-PART ((CLASS WOOD) (PROPERTIES
  (HARD)))))
```

The semantic heads of these definitions are *boat, person,* and *tree* respectively, this being different in the last case from the syntactic head ("type") of the definition. The other information in these structures is derived from adjectives, prepositional phrases, and relative clauses. Not all the information present in the definitions is captured, for example the information conveyed by the phrase *sometimes used in HEDGEs,* in which *HEDGE* is capitalised because it is not part of the restricted definition vocabulary (but is defined in terms of this vocabulary elsewhere).

Verb sense definitions are, in general, infinitive verb phrases with adverbials (often prepositional phrases) and additional restrictions on the semantic class of agents and objects. These are some examples of deriving structures from verb sense definitions.

(launch)

(to send (a modern weapon or instrument) into the sky or space by means of scientific explosive apparatus)

```
((CLASS SEND)
(OBJECT
  ((CLASS INSTRUMENT) (OTHER-CLASSES
  (WEAPON))
(PROPERTIES (MODERN))))
(ADVERBIAL ((CASE INTO) (FILLER (CLASS
  SKY))))
```

(mug)

(to rob with violence, as in a dark street)

```
((CLASS ROB)
  (ADVERBIAL ((CASE WITH) (FILLER (CLASS
  VIOLENCE)))))
```

(club)

(to beat or strike with a heavy stick (*CA CLUB *CB))

```
((CLASS STRIKE) (OTHER-CLASSES ((BEAT)))
(ADVERBIAL
  ((CASE WITH)
  (FILLER (CLASS STICK) (PROPERTIES
  (HEAVY))))))
```

Similarly, adjective sense definitions tend to have adjectival or verbal predicates as their heads, and they often include restrictions on the class of objects to which the property corresponding to the adjective can apply. The adverbial phrases used to define adverbs are often prepositional phrases. Examples of adjective and adverb definitions are the following.

(bushy)

((of hair) growing thickly: *46 a bushy beard / tail)
(CLASS PROPERTY)
(PREDICATION (CLASS GROW) (MANNER THICKLY))
(RESTRICTED-TO ((CLASS HAIR))))

(undomesticated)

((of an animal) not serving man; not *CA TAME)

((CLASS PROPERTY)
(PREDICATION (NOT (CLASS SERVE) (OBJECT MAN)))
(RESTRICTED-TO ((CLASS ANIMAL))))

(overland)

(across or by land and not by sea or air)
((MANNER ((CASE ACROSS) (FILLER ((CLASS LAND))))))

LDOCE definitions for lexicalized compound noun and phrasal verbs are handled in exactly the same way as noun and verb definitions. Two examples of structures generated for such definitions are given below.

(roller coaster)

(a kind of small railway with sharp slopes and curves, popular in amusement parks)

((CLASS RAILWAY) (COLLECTIVE KIND)
(PROPERTIES (SMALL)))

(bring out)

(*46 becoming rare *44 to introduce (usu. a young lady) into the social life of a great city *63 see also *CA COME OUT *CB (7))

((CLASS INTRODUCE)
(OBJECT ((CLASS LADY) (PROPERTIES (YOUNG))))
(ADVERBIAL
((CASE INTO)
(FILLER (CLASS LIFE) (PROPERTIES (SOCIAL))))))

Some problems

The current implementation is able to locate the correct semantic heads of dictionary definitions in most cases, although the examples above are untypical in the amount of additional information they recover from the definitions. Some quantitative remarks about the performance of the system are given later. This section briefly discusses a number of problems that were encountered while testing the implemented system.

In some respects the information conveyed by the output structures, being too closely tied to the surface definitions, only provides constraints for further semantic analysis. Perhaps the most important case of this is that the relationships implicit in compound nouns and certain prepositional phrase adverbials cannot, in general, be made more explicit without further interpretation apparatus (see e.g. Alshawi, 1987) beyond that available to the definition analyser. The phrasal context can, however, sometimes allow further specification of relationships implicit in prepositions, for instance derivation of *PURPOSE* from *for* in cases exemplified by the noun sense of *launch* (although, of course, errors can result from attempting to make relationships more explicit in this way). The actual words appearing in the semantic structures are, on the other hand, further disambiguated than might be assumed given the high degree of polysemy of many of the words in the restricted vocabulary. This is because the analysis process identifies the syntactic category of these words and because of the LDOCE rule that only the most central senses of words from the restricted vocabulary should appear in definitions (but see the remarks below on phrasal verbs).

The fact that definition texts are often not analysed completely means that information that is central to a definition is sometimes not taken into account, as illustrated by the following example. In this case the usual 'purpose' of nails is not recovered.

(nail)

(a thin piece of metal with a point at one end and a flat head at the other for hammering into a piece of wood, usu. to fasten the wood to something else)

((CLASS PIECE) (MATERIAL METAL) (PROPERTIES (THIN))
(HAS-PART ((CLASS POINT))))

Although the base forms of all the words in the restricted vocabulary and simple morphological variants of these are handled by the analysis process, there are many cases of derivational morphology which are not currently handled. Difficulties are also caused by the liberal use in LDOCE definitions of phrasal verbs made up from verbs and particles taken from the restricted vocabulary. The idiomatic nature of phrasal verbs means that the rule of using only the central senses of words in the definition vocabulary is violated in many cases in which phrasal verbs are used to implicitly increase the size of the defining vocabulary. An example is the occurrence of *look after* and *bring up* causing an error in the analysis of the following sense definition for *foster*:

(foster)

(to look after or bring up (a child or young animal) as one's own . . .)

((CLASS LOOK)).

Another problem encountered in LDOCE entries is that word senses are sometimes defined in terms of previous senses of the same homograph. For example immediately after the definition of the sense of *hornbeam* given earlier, the following dependent word sense definition is present, for which the system produces a structure containing the special symbol '*previous-sense*'.

(the wood of this tree)

((CLASS WOOD) (RELATED-TO
PREVIOUS-SENSE))

A problem related to the one just mentioned is that only the simplest forms of cross references to words not included in the definition vocabulary are handled at present. However, given the compositional nature of nested feature lists, and the fact that definitional cross references are intended to be non-circular in LDOCE, it should be feasible to use semantic structures for the referenced words (and previous senses as in the *hornbeam* example) in building other semantic structures.

The use of a restricted vocabulary in LDOCE definitions means that the lexicographers have already engaged in a substantial amount of semantic analysis of word senses that is potentially useful for automatic natural language processing. However, as observed by Michiels (1982), there is a tradeoff between the size of the definition vocabulary and the syntactic complexity of definitions. This implies that in order to take full advantage of the potential of LDOCE entries for language processing we need to pay special attention to the design of the definition analyser; this is the issue addressed in the rest of this paper.

Phrasal analysis hierarchies

The analysis mechanism has the flavour of a pattern-based phrasal analyser. It was designed to overcome some of the more obvious difficulties of applying a simple pattern matching approach to robust phrasal analysis. In particular it was required that the mechanism should have the means to specify which components of a phrase are more important, and to index analysis rules so that the mechanism would be reasonably efficient.

Pattern matching has played an important role in several previous parsing systems, for example those of Wilks (1975), Parkinson et al (1977), Wilensky and Arens (1980), and Hayes and Mouradian (1981). A characteristic of the mechanism used here is that it depends on a hierarchy of phrasal analysis patterns in which more specific patterns are dominated by less specific ones.

The basic analysis algorithm that applies the hierarchy of patterns is as follows. Starting at the top of the hierarchy, a pattern is matched against the input definition. If the match with this pattern succeeds then a match is attempted with each of its daughter patterns (i.e. the more specific forms of this pattern placed immediately below it in the hierarchy). This procedure is repeated recursively so that we end up with the most specific matches against the input definition. This parsing technique is different in kind from the more common approach to robust parsing in which exact grammar rules are tried first before being relaxed by the parser (see e.g. Weischedel and Black, 1980, Kwasny and Sondheimer, 1981, and Pulman, 1984).

The hierarchy provides a natural solution to the indexing problem mentioned above since it restricts the application of patterns to those that are more likely to succeed, enabling efficient phrasal analysis. It also provides a solution to the problem of specifying the more important components of phrases since less specific patterns tend to be concerned with more important components only. This ensures that reasonable incomplete analyses can be produced when more detailed analyses are not possible.

Each analysis rule consists of a rule identifier, a phrasal pattern, and a list of rule identifiers for daughter patterns. It is written in the following form:

(rule identifier phrasal pattern daughter identifiers).

The rule identifier also appears in a semantic structure building rule. These two types of rule are kept separate in order to allow different kinds of output structures to be generated for the same analysis grammar. Building semantic structures is basically a simple process of fleshing out templates provided by the semantic structure building rules using variable bindings generated by the matching algorithm.

The following section gives some examples of analysis and structure building rules, explaining the notation in which the phrasal patterns are written. The notation currently provides a limited number of facilities, but it should be clear that these facilities could be extended in various ways while remaining within the the overall framework of applying a hierarchy of phrasal patterns as discussed above.

Analysis rules

A typical analysis rule, n-100, for noun definitions, and two of its descendants, n-110 and n-135, are shown below. n-110 is a daughter of n-100, and n-135 is a daughter of n-130 (not shown). More mnemonic identifiers for these rules might be "Noun-phrase", "Simple-NP", and "NP-with-relative" instead of n-100, n-110, and n-135 respectively.

(n-100 (n && +0det && &0adj &noun &&) n-110 n-120 n-130 n-140)

(n-110 (n +0det +0intens &0adj &noun *0pp-mod &&))

(n-135 (n +0det &0adj &0noun +noun1 +that-which *verb-pred &&))

In the phrasal pattern part of these rules, the initial "n" restricts the pattern to matching definitions for senses with lexical category "n", i.e. nouns. The other pattern elements match zero or more items in the input depending on the type of the element (indicated by its first one or two characters) and restrictions in terms of lexical features. Digits at the end of pattern elements simply distinguish different occurrences of elements with the same properties. Examples of pattern elements and what they can match are the following.

for	the word *for*
+noun	exactly one noun
+0det	zero or one determiner
&noun	one or more nouns
&0adj	zero or more adjectives
&&	an arbitrary segment of input words
*0pp-mod	zero or one prepositional phrase modifier
*verb-pred	a segment that matches a verb phrase pattern

The last element is an example of an element with subsidiary patterns (in this case for verb phrases) which use the same kinds of element as above. There are thus associated with this element a list of rules including '(passive-pred (be +vtrans))'. Here 'passive-pred' is just the name of the subsidiary pattern '(be +vtrans)'. Similarly, one subsidiary pattern of '*0pp-mod' is '(for-pp (+0used for +noun-verb)'. The use of elements with such subsidiary patterns allows for recursion and a more compact set of patterns, in much the same way as for conventional context free phrase structure grammars.

Given this interpretation of pattern elements, it should be clear that the phrasal patterns of rules n-110 and n-135 will match subsets of the set of definitions matched by the phrasal pattern of rule n-100. The noun sense definition examples given earlier for *launch* and *mug* matched n-110 and n-135 respectively. (There is an initial morphological analysis phase which discards items like *usu.* which it does not recognize.) The analysis algorithm outlined in the previous section ensures that n-110 and n-135 are tried only if n-100 succeeds.

The structure building rule associated with an analysis rule is applied when none of the immediate descendants of the analysis rule succeed. Thus the structure building rule for n-100 is applied when none of n-110, n-120, n-130, or n-140 succeed, ensuring that a semantic structure is built according to the analysis provided by n-100 if no more specific version of this analysis is possible. The structure building rules for n-100 and n-135 are given below.

```
(n-100 ((compound-class &noun) (properties &0adj)))
```

```
(n-135 ((class +noun1) (noun-mods &0noun)
        (properties &0adj) (predication *verb-pred)))
```

The semantic structure given earlier for the definition *a foolish person who is easily deceived* . . . (a British English sense of *mug)* was generated using the rule just given above. Applying the rule involves replacing the variables with bindings generated by the matching process; splicing-out substructures associated with uninstantiated optional pattern elements; and recursively applying this process to the structure building rule associated with the appropriate (i.e. successfully matching) subsidiary pattern for the element '*verb-pred'. This last step results in building the substructure "(predication (object-of ((class deceive)))" using a rule associated with the subsidiary pattern 'passive-pred' that was mentioned earlier. There is also an optional further stage of the structure building process which applies transformations specified as attached procedures associated with items in the structure building rules, for example the item 'predication'. This phase gives greater freedom than would be possible by the use of structure building templates alone, for example it allows moving items (such as those indicating negation) 'upwards' out of substructures.

The analysis algorithm follows all paths from successful matches. This does not lead to inefficiency because it is rare for several deep, but disjoint, paths to be followed successfully down the pattern hierarchy, and because the implementation maintains a well-formed substring table to avoid a certain amount of redundant computation. Alternative semantic structures can result from processing a definition when there is more than one most specific successful analysis rule. At present one such analysis is chosen by an over simplistic heuristic that basically prefers analyses accounting for more words of the input definition.

Performance remarks

Although some of the difficulties mentioned in the section on problems are not easy to accommodate in the present system, most of the errors in identifying semantic heads of definitions were not due to these. Instead these errors appear to be caused mainly by failure on the part of the rudimentary morphological analysis performed, and the inadequate coverage of the present set of phrasal patterns.

In order to evaluate the performance of the system, it was tested against a sample of 500 definitions chosen at random using a facility for automatic random selection of entries provided by the dictionary access system. Only a few of these definitions will have coincided with those processed during the development of the system and its analysis rules. The selection process ignored definitions for idioms and those with complex cross references, so these are not taken into account in the figures given below.

The results of the test were as follows. The semantic head was identified correctly for 387 definitions (77%). Additional information was recovered for 236 (61%) of these definitions, and this additional information was judged to be correct for 207 (88%) of these cases.

Thus only identifying the head is much more typical than might be suggested by the examples given earlier for illustrative purposes, and in fact the present set of rules rarely takes into account all the words appearing in a definition. There are altogether some 90 phrasal patterns in the hierarchy, including subsidiary patterns. It should be emphasized that this grammar of definitions was written as a feasibility test, and I believe it is reasonable to expect that the number of patterns in the hierarchy could be enlarged to 400, say, before the problem of diminishing returns becomes a serious one.

The current implementation of the definition analyser takes around half a second to access the dictionary and process a definition. This is the elapsed time on a lightly loaded GEC-63 (a 32 bit mini-computer); the definition analyser was implemented in Cambridge Lisp and low level dictionary access in the language C. Finally, perhaps it is worth mentioning that the development effort was only a few man months for each of the program and grammar, which, compared with other natural language processing systems we have developed recently at Cambridge, represents a relatively small effort.

Further research

The work carried out so far seems to suggest that dictionary definitions can be analysed with a reasonable degree of success using hierarchies of phrasal patterns, but it still remains to be demonstrated that this technique can enable an actual natural language application system to cope effectively with unknown words.

Although dictionary definitions exhibit a rich variety of forms, these are mostly variations on a manageably small number of basic forms, and it is this property of definitions that makes phrasal pattern hierarchies particularly appropriate for analysing them. It seems likely however that the analysis technique developed here would be useful for the same reason in other language processing applications, for example specialized interactive applications.

One direction in which it is hoped to extend the work reported in this paper is in enhancing the capabilities of natural language processing systems for coping with idioms. Intuitively, some sort of pattern matching seems to be appropriate for analysing idioms (see e.g. Wilensky and Arens, 1980). In the context of a parsing system using an analysis hierarchy the patterns for idioms would be placed as most specialized patterns (i.e. leaves). LDOCE entries contain a wealth of information on idiomatic uses of words, and the meanings of idioms are expressed using the restricted definition vocabulary. It is hoped to extend the definition analysis system so that it would attempt to generate appropriate phrasal patterns when it encountered a definition for an idiomatic use of a word. It may then be possible to use the generated pattern and the definition of the idiom to produce a paraphrase of an input sentence before further processing takes place. In any case, a comprehensive treatment of dictionary entry analysis for language understanding systems clearly needs to take account of idiomatic word usage.

Acknowledgments

I am very grateful to the Longman Group for making this work possible by granting me access to the LDOCE tape for research purposes. I would also like to thank Bran Boguraev, Ted Briscoe, and two anonymous Computational Linguistics referees for many helpful comments that improved earlier drafts of this paper.

References

Alshawi, H. 1987 *Memory and Context Mechanisms for Language Interpretation.* Cambridge University Press, Cambridge, England.

Alshawi, H.; Boguraev, B.; and Briscoe, E. 1985 Towards a Dictionary Support Environment for Real Time Parsing. In *Proceedings of the Second Conference of the European Chapter of the Association for Computational Linguistics,* Geneva: 171-178.

Amsler, R. 1981 A Taxonomy for English Nouns and Verbs. In *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics,* Stanford, California: 133-138.

Bobrow, R.J.; and Webber, B.L. 1980 Knowledge Representation for Syntactic/Semantic Processing. In *Proceedings of the First AAAI Conference,* Stanford, California: 316-323.

Calzolari, N. 1984 Detecting Patterns in A Lexical Database. In *Proceedings of the 10th International Conference on Computational Linguistics,* Stanford, California: 170-173.

Carnap, R. 1952 Meaning Postulates. *Philosophical Studies* 3: 65-73.

DeJong, G. 1979 Prediction and Substantiation: A New Approach to Natural Language Processing. *Cognitive Science* 3: 251-273.

Hayes, P.J.; and Mouradian, G.V. 1981 Flexible Parsing. *American Journal of Computational Linguistics* 7(4): 232-242.

Katz, J.J.; and Fodor J.A. 1963 The Structure of a Semantic Theory. *Language* 39: 170-210.

Kwasny, S.C.; and Sondheimer, N.K. 1981 Relaxation Techniques for Parsing Grammatically Ill-Formed Input in Natural Language Understanding Systems. *American Journal of Computational Linguistics* 7(2): 99-108.

Mark, W. 1981 Representation and Inference in the Consul System. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence,* Vancouver, British Columbia: 375-381.

Michiels, A. 1982 Exploiting a Large Dictionary Database. Ph.D. thesis, Université de Liège, Liège.

Parkinson, R.C.; Colby, K. M.; and Faught, W. S. 1977 Conversational Language Comprehension Using Integrated Pattern-Matching and Parsing. *Artificial Intelligence* 9: 111-134.

Procter, P., Ed., 1978 *Longman Dictionary of Contemporary English.* Longman Group Limited, Harlow and London.

Pulman, S.G. 1984 Limited Domain Systems for Language Teaching. In *Proceedings of the 10th International Conference on Computational Linguistics,* Stanford, California: 84-87.

Schmolze, J.G.; and Lipkis, T.A. 1983 Classification in the KL-ONE Knowledge Representation System. In *Proceedings of Eighth International Joint Conference on Artificial Intelligence,* Karlsruhe: 330-332.

Walker, D.; and Amsler, R. 1986 The Use of Machine Readable Dictionaries in Sublanguage Analysis. In *Analyzing Language in*

Restricted Domains, edited by Ralph Grishman and Richard Kittredge. Lawrence Erlbaum Associates, Hillsdale, New Jersey: 69-83.

Weischedel, R.M.; and Black, J.E. 1980 Responding Intelligently to Unparsable Inputs. *American Journal of Computational Linguistics* 6(2): 97-109.

Wilensky, R.; and Arens, Y. 1980 PHRAN — A Phrasal Natural Language Understander. In *Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics,* Philadelphia, Pennsylvania: 117-121.

Wilks, Y. 1975 An Intelligent Analyser and Understander of English. *Communications of the ACM* 18: 264-274.