Morphological Cues for Lexical Semantics

Marc Noel Light

Technical Report 624 and Ph.D. Thesis
June 1996

UNIVERSITY OF
ROCHESTER
COMPUTER SCIENCE

Morphological Cues for Lexical Semantics

by

Marc Noel Light

Submitted in Partial Fulfillment

of the

Requirements for the Degree

Doctor of Philosophy

Supervised by

Professor Lenhart K. Schubert

Department of Computer Science
The College
Arts and Sciences

University of Rochester
Rochester, New York

1996

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
June 1996

3. REPORT TYPE AND DATES COVERED
technical report

4. TITLE AND SUBTITLE
Morphological Cues for Lexical Semantics

5. FUNDING NUMBERS
ONR N00014-92-J-1512

6. AUTHOR(S)
Marc Noel Light

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESSES
Computer Science Dept.
734 Computer Studies Bldg.
University of Rochester
Rochester NY 14627-0226

8. PERFORMING ORGANIZATION

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESSES(ES)
Office of Naval Research
Information Systems
Arlington VA 22217

ARPA
3701 N. Fairfax Drive
Arlington VA 22203

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
TR 624 and Ph.D. Thesis

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Distribution of this document is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)
(see title page)

14. SUBJECT TERMS
lexical acquisition; lexical semantics; morphology

15. NUMBER OF PAGES
145 pages

16. PRICE CODE
free to sponsors; else $6.50

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
unclassified	unclassified	unclassified	UL

Curriculum Vitae

Marc Light [redacted] received his B.S. in Cognitive Science from the Massachusetts Institute of Technology in 1988. He then worked at the University of Zurich under the supervision of Dr. Marc Domenig on lexical databases.

He entered the Ph.D. program at the University of Rochester in 1989 and received an M.S. in Computer Science in 1991. At Rochester, he has worked primarily with Prof. Lenhart Schubert on natural language semantics and processing.

Acknowledgments

It is a rainy Sunday morning in Tübingen, Germany; my daughter is sleeping, my wife is reading, and my graduate career is almost over. I started the University of Rochester Computer Science graduate program in the Fall of 1989 at the age of twenty-three. Since that time I have lived in three different cities, married, become a father, and turned thirty—graduate school has been a long varied haul and I could never have made it alone. Thus it is a good time to thank all the people that helped make my completion of the program possible.

I will start by thanking the city of Rochester, the community in which I spent the majority of my graduate career. Rochester cannot claim to be one of the world's great cities. In fact, it cannot even claim to be one of the great American cities. Rochester does not offer the sites, restaurants, music, or theater that cities like New York, Boston, Chicago, or San Francisco do. However, Rochester has enough of these things for me. In addition, they all come with neither an elitist nor a defeatist attitude: Rochester doesn't think it is better or worse than the rest, it just thinks it is pretty good. I believe that this attitude was, in part, responsible for my enjoyment of the city. More specifically, I would like to thank the following Rochester organizations: Wegmans, the Amerks, Country Sweet, the Genesee Brewing Company, the Rochester Players Ultimate League, and most importantly Dog's Life.

While residing in Rochester, I lived in a house at 152 Shelbourne Road. I would like to thank Shinta Cheng, David Kumasaka, Peter Kumasaka, Beth Murphy, Mike Burek, and Jeff Thomas for making that house a great place to come home to.

One factor in my coming to Rochester was that the University of Rochester Computer Science department seemed, during the application process, to be a small community-oriented place. It is in fact such a place and I enjoyed my time there. I would like to thank Jill Forster, Peggy Franz, Marty Guenther, Pat Marshall, and Peg Meeker for providing a well-oiled administrative machine and Luidvikas Bukys, Jim Roche, Tim Becker, and Brad Miller for providing a well-oiled computer system. I would also like to thank Tom Leblanc for the hockey. Finally, thanks for all the good times, Chris Brown, Corinna Cortes, Mark Crovella, Virginia Desa, Cezary Dubnicki, Derek Gross, Yenjo Han, Peter Heeman, Janet Hitzeman, Chung Hee Hwang, Jonas Karlsson, Graham Katz, Leonidas Kontothanassis, Andrea Li, Nat Martin, Massimo Poesio, Polly Pook, Rajeev Raman, Eric Ringger, Jeff Schneider, Dave Traum, and Lambert Wixson. A special thanks goes out to George Ferguson and Pia Cseri-Briones for their friendship and hospitality.

A number of people provided more direct support of my graduate work. First, I would like to thank my advisor, Len Schubert. He pushed me to understand my topic more completely than I would have on my own. Perhaps more importantly, I always felt that I could trust him and that he truly wanted me to succeed. I would also like to thank the other members of my committee—James Allen, Greg Carlson, and Peter Lasersohn—for their support and advice. I regret that I was not able to make greater use of their wisdom. This was due in part to the fact that for the last four years of graduate school my presence in Rochester was only sporadic.

This brings me to another set of people that I want to thank. Due to my wife and my instance of the two-body problem, I lived the majority of 1993 and 1994 in Ithaca. I would like to thank the Cornell Computer Science Department and more specifically Dan Huttenlocker for letting me use their computer system during this time. I would also like to thank Sally McConnell-Ginet and Alessandro Zucchi of the Cornell Linguistics Department for a number of helpful discussions. Finally, I would like to thank John Cordo, Jackson Galloway, and Nick Straley for showing me many a good time and making me feel at home in Ithaca.

After Ithaca, we moved to Tübingen where I have been working at the University of Tübingen in the Computational Linguistics department. I spent many weekends and holidays working on my thesis and I often was anything but well-rested when I showed up for work on Monday morning. I would like to thank Steve Abney and Erhard Hinrichs for their understanding.

Since I began working on my dissertation, my wife, Jill Gaulding, has supported my every step. This has been particularly true in the last 6 months, during which my job and my thesis have left little time for anything else. She has also provided useful outsider advice at a number of critical moments. Thanks. And thank you, Ellen, for keeping us up a minimal number of nights.

Finally, and most importantly, I would like to thank my parents for always being there for me. My successful completion would not have been possible without the security and support they provided and still provide.

(This dissertation is based on work supported by ONR/ARPA research grant number N00014-92-J-1512.)

Abstract

Most natural language processing tasks require lexical semantic information such as verbal argument structure and selectional restrictions, corresponding nominal semantic class, verbal aspectual class, synonym and antonym relationships between words, and various verbal semantic features such as causation and manner. This dissertation addresses two primary questions related to such information: how should one represent it and how can one acquire it.

It is argued that, in order to support inferencing, a representation with well-understood semantics should be used. Standard first order logic has well-understood semantics and a multitude of inferencing systems have been implemented for it. However, standard first order logic, although a good starting point, needs to be extended before it can efficiently and concisely support all the lexically-based inferences needed. Using data primarily from the TRAINS dialogues, the following extensions are argued for: modal operators, predicate modification, restricted quantification, and non-standard quantifiers. These representational tools are present in many systems for sentence-level semantics but have not been discussed in the context of lexical semantics.

A number of approaches to automatic acquisition are considered and it is argued that a "surface cueing" approach is currently the most promising. Morphological cueing, a type of surface cueing, is introduced. It makes use of fixed correspondences between derivational affixes and lexical semantic information. The semantics of a number of affixes are discussed and data resulting from the application of the method to the Brown corpus is presented.

Finally, even if lexical semantics could be acquired on a large scale, natural language processing systems would continue to encounter unknown words. Derivational morphology can also be used at run-time to help natural language understanding systems deal with unknown words. A system is presented that provides lexical semantic information for such derived unknown words.

Table of Contents

List of Tables

1 Introduction

Information about individual words is crucial for every natural language processing (NLP) task—it is hard to imagine an NLP system without a lexicon of some form. Lexicons can contain phonological, morphological, syntactic, semantic, and/or pragmatic information. The form of this information can range from simple atomic features to designer non-monotonic intensional logics. Increasingly, such discrete representations are being augmented, and in some cases replaced, by statistical information. Thus, the builder of an NLP lexicon has many types of information and representations from which to choose. This thesis will concentrate on lexical semantic information.

Automatic methods for acquiring lexical semantic information are less plentiful and thus currently the semantic component of most NLP lexicons is built by hand. However, the situation is rapidly improving for some types of lexical semantic information. The appearance of large online corpora has made the collection of large amounts of co-occurrence data possible. Statistical methods can then be applied to this data to cluster words that appear in similar contexts (*e.g.*, [Brown *et al.*, 1992; Schuetze, 1993]). These methods are particularly good at inducing coarse-grained semantic information such as semantic relatedness and subject area. Such information is very useful for some NLP tasks: examples include information retrieval [Hearst and Schuetze, 1994] and word sense disambiguation [Schuetze, 1993; Brown *et al.*, 1991]. Semantic relatedness can also be obtained by the automatic processing of machine readable dictionaries (MRDs) (*i.e.*, dictionaries for humans) [Pentheroudakis and Vanderwende, 1993].

However, many NLP tasks require at least a partial understanding of every sentence or utterance in the input and thus have a much greater need for lexical semantics. Natural language generation, providing a natural language front end to a database, information extraction, machine translation, and task-oriented dialogue understanding all require fine-grained lexical semantic information, *e.g.*, verbal argument structure and selectional restrictions, corresponding nominal semantic class, verbal aspectual class, synonym and antonym relationships between words, and various verbal semantic features such as causation and manner. Such fine-grained information is more difficult to acquire. Statistical methods based on distributional data can provide some of this information such as selectional restrictions (*e.g.*, [Grishman and Sterling, 1992]): nouns are clustered based on the verbs they appear with and verbs are clustered based on the nouns they appear with. However, the link between distribution and information such as aspectual class or antonym is more tenuous and thus it will be more difficult for statistical methods to extract this information. MRDs would seem more promising since they are specifically designed to convey the meaning of a word. Unfortunately in most cases MRDs only contain such information implicitly and making it explicit often presupposes a general solution to NLP which requires lexical semantic information, the very information we are trying to acquire. However, some success has been achieved by keying off set phrases such as *any of a* and *one that* which correspond to the semantic relationships hyponym and humanness [Markowitz *et al.*, 1986; Chodorow *et al.*, 1985] but this method is restricted to particular semantic relations that correspond to easily identifiable patterns. [Hearst, 1992] describes a similar technique which utilizes robust tagging and parsing systems to enable the search of open text corpora for more complex syntactic contexts. A pattern utilized in [Hearst, 1992] to extract hyponyms is $NP_0 such\ as \{NP_1, NP_2, ... \{and|or\} NP$. From the sentence *mammals such as cats are covered with fur*, such a system would extract from that mammal is a hypernym of cat. Again the problem for such an approach is finding easily identifiable contexts which correspond to the desired semantic property. Another technique that has achieved partial success in acquiring fine-grained lexical semantics utilizes a partial processing of the text containing an unknown word, *i.e.*, the semantics of surrounding words constrain the meaning of an unknown word [Granger, 1977; Berwick, 1983; Hastings, 1994]. However, as with MRDs, this approach is hindered by the need for a large amount of initial lexical semantic information and the need for an initial robust natural language understanding system.

As a consequence of this situation, current NLP systems which require fine-grained lexical semantics have only small hand built lexicons and thus can only operate in restricted domains. Improved methods for automatic acquisition of lexical semantics could increase both the robustness and the portability of such systems. In addition, such methods might provide insight into human language acquisition.

This thesis provides a new method for acquiring fine-grained lexical semantic information. The method is based on derivational morphology which is used as a "cue" for fine-grained lexical semantic information. The method is analogous to the work described in [Hearst, 1992] and mentioned above: it also uses a surface "pattern" as a cue for a particular semantic characteristic. The advantage of using morphological cues is that they are often easy to identify and correspond to very fine-grained semantic information. The disadvantages are that not all the needed information is encoded morphologically and that not every word that has a semantic property is morphologically marked for it. However, it will be shown that even for a morphologically impoverished language like English a great deal of semantic information is encoded morphologically. In addition, it appears that the majority of English words are either morphological complex themselves or serve as a base of a morphologically complex word.[1]

An example of a morphological cue is the verbal prefix *un-* which applies to change-of-state verbs and produces change-of-state derived forms. A change-of-state verb is one that presupposes the negation of its result state, *e.g.*, in order to *fasten* something it must be at least partially unattached to begin with. Thus, it is possible to use *un-* as a cue for the change-of-state feature. By searching a sufficiently large corpus we are able to identify a number of change-of-state verbs. Examples from the Brown corpus include *clasp, coil, fasten, lace, screw*, and *load.* In addition, if an NLP system encounters, at run-time, an unknown verb such as *unfluggle* which appears to have *un-* as a prefix, it can infer that both this new verb and its base (*i.e.fluggle* are likely to be change-of-state verbs).

This information is crucial for some NLP tasks. For example, consider the following dialogue segment which was taken from the TRAINS dialogues.[2]

(1) M: bring it all the way down to Bath
: to pick up a boxcar and bring it all the way back to Avon
: load it with bananas and then take it off to Corning
S: okay
: if we did that
: it would take ...

In order to verify the plausibility of the manager's plan (M) and to carry it out, the system (S) needs to find a boxcar that is **empty**, since it will be filled with bananas. Thus, to pick an appropriate boxcar, the system would have to know that to load a boxcar, it has to be empty first. One possible place for this information to reside is as the lexical semantics of the verb *load*: the entry for the verb *load* should contain the information or pointers to the information that *load* is a change-of-state verb, *i.e.*, that before one can load a boxcar it needs to be empty or at least not full. [3]

[1]The following experiment supports this claim. Just over 400 open class words were picked randomly from the Brown corpus and the derived forms were marked by hand. Based on this data, a random open class word in the Brown corpus has a 17% chance of being derived, a 56% chance of being a stem of a derived form from the corpus, and an 8% chance of being both.

[2]In a TRAINS dialogue, two people, one as the system and the other as the user, together try to perform a planning task. The tasks involving trains, cities, factories, products, and deadlines. Spoken discourse was recorded by microphone and tape recorder. For an overview of the TRAINS project see [Allen *et al.*, 1994] and for a discussion and listing of the dialogues see [Gross *et al.*, 1993].

[3]One might argue that this information is really world knowledge not linguistic knowledge and

In addition, to describing a method based on derivational morphology for acquiring fine-grained lexical semantic (both off-line and at run-time), this thesis will also discuss what sort of representational system is required for encoding such information. It will be argued that a representation with a denotation should be used and that it should include extensions to standard first order logic (FOL) such as non-standard quantifiers, restricted quantification, modal operators, predicate modification, and predicate nominalization. One representation system that meets these needs is Episodic Logic [Hwang and Schubert, 1993a].

An example of the representation which will be used in this thesis is the axiom below which partially defines the change-of-state (COS) feature.

For all predicates P with features COS and DYADIC[4]:

$$\forall x, y, e\, [\mathrm{P}(x,y) ** e \supset [\exists e1 : [\textit{at-end-of}(e1, e) \wedge \textit{cause}(e, e1)]\; [\textit{rstate}(\mathrm{P})(y) ** e1] \wedge \exists e2 : [\textit{at-beginning-of}(e2, e)]\; [[\neg \textit{rstate}(\mathrm{P})(y)] ** e2]]]$$

The operator ** is analogous to $\models$ in situation semantics; it indicates that a formula describes an event. The axiom states that if a change-of-state predicate describes an event, then the result state of this predicate holds at the end of this event and that it did not hold at the beginning, *e.g.*, if one wants to formalize something it must be non-formal to begin with and will be formal after.

In sum, the main contributions of this thesis are i) a method of acquiring lexical semantics information which utilizes the morphological structure of words, ii) a method for processing some of the new [5] morphologically complex words as they appear in the input stream of an NLP system, and iii) an argument that a language more expressive than FOL is needed to represent fine-grained lexical semantic information.

This thesis is structured as follows. A review of the lexical semantic information utilized by various NLP system is presented in the Chapter 2. The following chapter, Chapter 3, examines the issue of representation and argues for the representation briefly illustrated above. It also reviews previous work on representing lexical semantic information. Chapter 4 reviews some acquisition methods and presents a new method: morphological cueing. Chapter 5 deals with the related problem of unknown words in the input stream of an NLP system. Many claims

has no place in the lexicon but instead should be in world knowledge module. The dispute over distinction between world and linguistic knowledge has a long history in philosophy. Whether the distinction exists and if so how it should be drawn are hotly contested. Regardless of the categorization of the information, it is clear that it is crucial for NLP tasks and that there must be some mechanism for tying a word in an utterance to this information.

[5]Here and throughout this thesis *new* will have a relative meaning: new to whatever system or human we are discussing. Whether or not a word is in some dictionary somewhere will most often be irrelevant to the discussion.

of this dissertation are based on semantic analyses of different derivational affixes. However, in order to allow for an uninterrupted discussion of representation and acquisition, the details of the analyses are presented in Chapter 6. Chapter 7 summarizes the dissertation and concludes. In the appendix the results of applying morphological cueing to the Brown corpus [Kucera and Francis, 1967] are listed.

2 The Need for Fine-grained Lexical Semantic Information

The previous chapter only contained one concrete example of fine-grained lexical semantic information and only one NLP task to which this information was relevant: change-of-state aspectual information and task-oriented dialogue understanding. In this chapter, examples and tasks will be provided for many types of lexical semantic information, namely, verbal argument structure and selectional restrictions, corresponding nominal semantic class, verbal aspectual class, synonym and antonym relationships between words, and various verbal semantic features such as causation and manner. The purpose of these examples is to make the meaning of the term fine-grained lexical semantic information clearer and to show why fine-grained information is crucial for NLP. The list is in no way complete.

2.1 Verbal argument structure

In argument structure, we include information about subcategorization, thematic arguments, and selectional restrictions. Argument structure is also referred to as verb valence and case frames. Although exactly what information should be part of argument structure and how it should be defined is disputed, the core phenomenon is clear: the argument structure of a verb is the information needed to relate its subcategorized (syntactic) complements to its thematic (semantic) arguments and vice versa. In other words, the argument structure of a verb specifies the main players in the situation described by the verb and their possible syntactic realization if any. For example, in (2a), the verb of the sentence is *send* and its argument structure specifies that the subject noun phrase is filled by the thing that does the action and the direct object noun phrase is filled by the thing that is affected by the action. The rest of (2) lists predicates that have various argument structures.

(2) a. Mary sent the book.

b. John received a book from Mary.

c. Mary ate.

d. The glass broke.

e. John laughed.

A number of generalizations can be made about how the meaning of a verb partially determines its argument structure. A simple example is that a verb that describes events involving two participants tends to be transitive. Levin [1993] documents a large set of such generalizations for English verbs.

Many of these generalizations involve semantic characteristics of the participants, *e.g.*, the participant that is the agent is usually realized as the subject. However, what is an agent? Following [Dowty, 1991], specifying that an argument is the agent of an action amounts to saying that it has some number of "agentive" properties such as volition, sentience, caused a change in the patient, and/or exists independent of the event. What is crucial is that "agentive" participant have more of these properties than any of the other arguments. Other thematic arguments can be defined similarly.

For a particular verb, many characteristics required of a participant are called selectional restrictions. The difference between selectional restrictions and thematic arguments is that, whereas the property sets defining thematic arguments are general and apply to many different verbs, selectional restrictions are often particular to a single verb. A clear example of a selectional restriction manifests itself in the contrast between the German verbs *essen* and *fressen* (see (3)). *Essen* restricts its agent to be human and *fressen* restricts it to be animate but not human.

(3) a. Die Frau ißt. (*The woman ate.*)

b. Der Hund frißt. (*The dog ate.*)

Selectional restrictions often only hold in limited domains. For example, if a dialogue understanding system only has to process dialogues about reserving seats on a plane, the verb *reserve* can restrict its direct object to be a seat on a flight. However, in a more general setting, it is possible to reserve things such as a particular car from a car rental office. Thus, selectional restrictions may be properly seen as a shortcut to doing more general domain reasoning.

Why is argument structure crucial for NLP tasks? Consider natural language understanding (NLU) systems which translate natural language into some internal language such as a database query language for database frontends or information template for abstraction systems. One of the main subtasks is mapping the verb to a semantic predicate and mapping the verb's syntactic complements to the predicate's arguments. There are two complicating aspects to this mapping that make it a problem for NLU tasks:

- many-to-many verb/predicate mapping: there is a many-to-many mapping between verbs and internal language predicates,
- many-to-many complement/argument mapping: a verb's arguments configure themselves syntactically in many different ways.

Consider, as an example, the verb *load*. For many tasks each of the sentences below would map to a different predicate.

(4) a. the woman loaded the gun with bullets

b. the man loaded the truck with hay

c. the child loaded up on chips

d. the courses loaded down the student

e. the occupants load the circuit with heaters in the winter

In addition, the predicate `load-container` could correspond to each of the verbs in the sentences below.

(5) a. the woman loaded the truck with hay

b. the woman filled the truck with hay

c. the woman put the hay into the truck

Thus, we potentially have a many-to-many mapping. The corresponding problem for an NLU system is to map each token of a verb to the correct predicate: verb sense disambiguation. Verbal argument structure helps by specifying the context that particular pairings occur in. One possible argument structure for *load* might specify that it has a human as the performer in its subject complement, a weapon as the one of the patients of the action in its object complement, and an ammunition as the other patients of the action in a prepositional phrase complement. This information would enable a system to pick out the `load-firearm` predicate for (4a) and rule it out for the (4b) and (4c).

As an example of the second aspect listed above consider the sentences below.

(6) a. The coach loaded the players onto the bus and left for Hershey

b. The coach loaded the bus with players and left for Hershey

c. The players loaded the bus and left for Hershey

Even if we assume that each sentence involves the same verb/predicate pair there is a problem mapping the complements of the verb to the arguments of the predicate. Again, argument structure information can help. It gives specific information about the characteristics of a predicate's arguments, *e.g.*, for `load-container(x,y,z)`, `x` has to be the initiator of the action, `y` has to be the container, and `z` the filler. This information can be used along with more general principles to match the complements to the arguments.

Verb sense disambiguation and multiple argument realizations are central problems of the MUC-3 task [Chinchor *et al.*, 1993]: separate a large number of messages into those that are relevant and those that are not; then fill set templates with information from the relevant messages. Relevant messages are those that have to do with terrorist acts in Latin America. The templates consist of slots such as type of incident, perpetrator, human target, physical target, instrument, effect. Verb sense disambiguation helps decide if a message is about terrorism. And dealing with multiple argument realizations helps fill the templates consisting of slots. Similarly, the project in [Pugeault, 1994] uses argument structure extensively to extract information from short research project descriptions. The extracted information is used for a variety of tasks among them listing who does what and what kind of results are available and identification of relations between projects. Other extraction projects that utilize argument structure are [Palmer, 1990; Palmer *et al.*, 1986; Herzog and Rollinger, 1991; Gomez, 1994].

Natural language front ends for data base systems also have to deal with verb sense disambiguation and multiple argument realizations. Early systems such as that described in [Bobrow and Webber, 1980] used argument structure to help solve these problems. Current systems continue this practice [Bates *et al.*, 1994; Alshawi, 1992; Dowding *et al.*, 1993; Backofen and Nerbonne, 1992].

For natural language generation (NLG), verb/predicate mapping and complement/argument mapping are equally important but in the opposite direction: the task is to pick an appropriate verb and syntactic complements based on a predicate, arguments, and other semantic/pragmatic information. This task is referred to as lexical choice or as lexicalization in the literature. A simple approach to lexical choice is to assume that these mappings are one-to-one. Then each predicate maps to one verb and that verb's argument structure specifies the syntactic complement to which each semantic argument maps. This simplistic approach often produces acceptable results. But if predicates map to different verbs depending on the situation or if discourse considerations require alternative syntactic complements then the two problems mentioned above, many-to-many verb/predicate mapping and the many-to-many complement/argument mapping will cause problems. Again, for many systems, verbal argument structure provides basic information for the choice of verb and its complements [Cumming, 1986; Matthiessen, 1991; Iordanskaja *et al.*, 1991]. For example, consider generating a sentence to express an event where Mary transferred a car to John in exchange for money. If Mary is currently in focus, for discourse structure reasons it might be desirable to have Mary in the subject position of the sentence. Thus, the verb *sold* should be used instead of *bought* as in *Mary sold the car to John*. The argument structures of *bought* and *sold* are crucial to this decision. (This example was adapted from [Pustejovsky and Nirenburg, 1987].)

Machine translation (MT) also makes extensive use of argument structure. Interlingua systems must grapple with the problems outlined for NLU and NLG above and thus argument structure is crucial for all the same reasons (*e.g.*, [Dorr, 1993; Nirenburg and Raskin, 1987]). In transfer systems, corresponding verbs must specify their argument structures so that the linking preserves thematic relations (*e.g.*, [Nagao, 1987]). In addition, lexical gaps in one language may force a correspondence between two verbs where the syntactic realization of semantic arguments are rearranged as in (7) (taken from [Dorr, 1994]).

(7) E: I like Mary ⇔ S: Maríe me gusta a mí ('Mary pleases me')

As for MT tasks, dialogue understanding tasks involve both NLU and NLG and thus inherits their problems and needs. In addition, for the TRAINS project, the argument structure in the lexicon is crucial for mapping to internal representations used for plan recognition and reasoning. Event types have argument role slots. For example, the *unload* event has an argument role for the container being unloaded and a role for the stuff being taken out. As we have seen this information can be used in conjunction with aspectual information to infer that the object filling the container role will be empty at the end of the *unload* event.

2.2 Aspect

The aspect of a verb specifies the structure of the event described by the verb. The aspectual classes that will be utilized in this thesis are those of [Vendler, 1967]: telic, non-telic, accomplishments, achievements, statives, and activities (see the examples in (8)).

(8) accomplishment: build a house, climb a mountain

achievement: awake, reach the top

stative: like, understand

activity: mingle, walk

For natural language tasks, aspect helps specify **when** events described occur. For example, aspectual class plays a part in the temporal ordering of the events in a dialogue.

(9) M: bring it all the way down to Bath
: to pick up a boxcar and bring it all the way back to Avon
: load it with bananas and then take it off to Corning

M: the tanker will carry the OJ and the entire train will go to Avon

The often noted (see [Kamp, 1979; Hinrichs, 1981; Partee, 1984; Eberle, 1991]) default generalization is that telics move the temporal reference forward and non-telics do not. Thus in the example, since the first three underlined verbs are telic, the picking up is followed by the bringing which if followed by the loading. In contrast, *carry* is an activity verb and thus the carrying takes place at the same time as the

In addition, the meaning of temporal modifiers is dependent on the aspect of the predicate they modify. For example, in (10a) temporal modifier *in an hour* specifies the amount of time leading up to the bouncing (activity) event. In contrast, *in an hour* specifies the length of the loading event itself in (10b) (see [Dowty, 1979] p.56).

(10) a. Mary will bounce the ball in an hour

b. Mary will load the truck in an hour

The examples given above show that aspect is important for NLU tasks, both those that involve multiple sentence inputs and those that might be constrained to single sentence inputs (*e.g.*, database front ends) [Eberle, 1991; Passonneau, 1988; Dahl *et al.*, 1987; Webber, 1988; Hwang and Schubert, 1993a].

Aspect is also clearly important for generating a discourse describing a set of temporally-related events. An example of the role of aspect in producing single sentences is given in [McDonald, 1991]. He describes the information and processing needed to choose between the two alternative ways in (11) of expressing the idea that the addressee needs to be in transition from being at a certain location to not being there.

(11) a. you need to leave by ...

b. you can stay until ...

The telic *leave* stresses the result of the transition whereas the activity *stay* stresses the activity before the transition. In the situation described in [McDonald, 1991] it is desirable to stress the result of the transition and thus (11a) is preferred.

An example of the need for is the aspectual information for MT is the translation of the Japanese verb *kaburu* into English [Hutchins and Somers, 1992]. If it is used to describe the process of getting a hat onto the agent then it should be translated as *put on* but if it describes the state of the hat being in place then it should be translated as *wear*—Japanese conflates the telic and activity events concerning the wearing of a hat whereas English has two separate verbs. To translate from Japanese to English the aspect of the event needs to be derived from context in Japanese and then matched to the aspectual information in the English lexicon.

2.3 Other verbal semantic information

In this section we will discuss some of the other semantic characteristics of verbs that can be utilized for NLP tasks. Almost any piece of information about a verb's meaning could be used in service of an NLP system. But there are a number of characteristics that seem to have extensive syntactic and semantic ramifications. Often they seem to underlie other semantic information such as verbal argument structure and aspect as many linguists have noted (*e.g.*, [Jackendoff, 1990; Pinker, 1989; Levin, 1993]). Some of these semantic characteristics are listed in (12) along with exemplary verbs.

(12) a. incorporated theme: *butter (the toast)*

b. causation: *(John) melted (the butter), kill*

c. creation: *build, bake, sew, cook, make* [Pinker, 1989]

d. magnitude: *toss vs. hurl*

e. incorporated instrument: *brush, comb, hose* [Pinker, 1989]

f. effect: *clean, cleanse, empty, strip* [Pinker, 1989]

g. manner: *spill vs. ladle, pour vs. spray* [Pinker, 1989]

[Wu and Palmer, 1994] shows how classes based on such information can be used in a lexical choice procedure for MT and NLG systems. And [Palmer *et al.*, 1986; Dahl *et al.*, 1987; Palmer, 1991] show how to use such information to solve the many-to-many complement/argument mapping problem for an NLU problem. Dorr [1993; 1994] shows how such information can be used to handle translation mismatches.

Given that there is an infinite number of semantic distinctions that could be drawn, a important question is how to decide what verbal semantic information should be part of the linguistic knowledge base. From an engineering standpoint, the answer is whatever is needed to solve the problem at hand. For example, in the TRAINS system, semantic distinctions are made that are needed to discuss and reason about plans in the TRAINS domain. Similarly, the system described in [Palmer, 1990], which uses verbal semantics to link syntactic complements to semantic arguments, uses the following principle: "the decompositions only need to go far enough to pick up any arguments to the verb that can be mentioned syntactically" ([Palmer, 1990], p.87). Here and throughout this dissertation, I will assume the same pragmatic stance.

2.4 Nominal information

In comparison to verbs the study of nominal semantics has received little attention. In fact much of the nominal semantics information in NLP systems is present to make applying verbal selectional restrictions possible (*e.g.*, [Palmer *et al.*, 1986; Gomez, 1994; Rickheit, 1991; Backofen and Nerbonne, 1992; Wu and Palmer, 1994; Pustejovsky and Nirenburg, 1987]. For example, we could classify knives as tools and men as humans and use this information to distinguish between the linking of the subject noun phrase with the instrument versus the agentive semantic argument in (13a). A similar mapping ambiguity in (13b) could be solved if trucks were classifying as containers and hay as a commodity. The sentence in (13c) contains an ambiguity between two senses of *start* or at least if there is only one sense of start it has different entailments depending on what sort of thing its patient argument is. For example, if it is a book we can assume Mary can read and if it is a car that she can drive.[1]

(13) a. the knife/the man cut the bread

b. Mary loaded the truck/the hay

c. Mary started the book/the car

Of course, the nouns themselves may be ambiguous and thus the process of using selectional restrictions to disambiguate verbs, nouns, and complement/argument mappings might involve some sort of constraint propagation.

In task-oriented dialogues, information about what the referent of a noun is used for or how it is created is needed to interpret an utterance. For example, in the dialogue fragment below (this dialogue is handled by the TRAINS system) the information that OJ is something that is made by processing oranges and that an OJ factory can do this processing is needed to recognize the second and third utterances as suggestions about which elements in the domain should be utilized to achieve the goal of making OJ and what role they should play.

(14) M: We have to make OJ.
: There are oranges at I
: and an OJ factory at B.

[1]This example is based on a discussion of *start* in [Pustejovsky, 1991].

: Engine E3 is scheduled to arrive at I at 3PM.
: Shall we ship the oranges?

S: Yes,
: shall I start loading the oranges in the empty car at I?

Another use of nominal semantics is for the lexical choice subtask of NLG. As with selectional restrictions, pretty much any piece of semantic information could be used. For example, in [Pustejovsky and Nirenburg, 1987] the problem of expressing the concept of a black, small, average-height, made of steel, 'location-of eat' table is addressed. They utilize the information about where it is usually located ('location-of eat') to pick out *dining table* from a set of words that include *desk, table, coffee table,* and *utility table.* Another example of information used for lexical choice is the characteristic of basic-level class. Following [Reiter, 1991], if a concept of a particular pit bull terrier, Spot, is to be expressed, and we don't want to specify anything special about Spot then we should use the word *dog.* If we use the word *mammal* or *pit bull* then we would be saying something special about Spot. Thus, for NLG, information about which nouns correspond to basic-level classes is needed. Finally, almost any piece of information might be useful. For example, the lexical choice system described in [Sondheimer *et al.*, 1989] utilizes a classifier that takes a set of characteristics of an entity and attempts to find the most general concept in the hierarchy that has these characteristics. Any characteristic could be helpful for the classification task.

The primary tool used for representing nominal semantic information is a sortal hierarchy. Concepts lower in a hierarchy are more specific than those higher up. In addition, concepts inherit properties from their supertypes. Such hierarchies are also called is-a hierarchies or taxonomies.

One problem with much of the work on nominal semantics is that there does not seem to be a theory behind decisions about what information should be stored. It appears that researchers have not been able to find constraints on what nominal semantic information is important to language. (This is in contrast to verbal semantics where information such as aspect or argument structure has been found to be important to all verbs and which is tightly constrained by language (*e.g.* one does not find verbs that refer to a process that includes a change from ¬X to X and back to ¬X). In fact a number of studies make strong claims about what verb meanings are possible, *e.g.*, [Pinker, 1989; Tenny, 1992].) One exception to this is the work of Pustejovsky (*e.g.*, [Pustejovsky and Anick, 1988; Pustejovsky, 1991]). He has tried to remedy this situation by introducing the concept of qualia structure. Qualia structure consists of four slots that each noun must fill:

Constitutive role: what it is made of?

Formal role: what picks it out from the other objects in the domain? (*e.g.*, orientation, magnitude, shape, color, dimensionality)

Telic role: what is it used for? what is its purpose?

Agentive role: what brought this thing about? how was it made?

Each noun would have these four roles and each would point to a concept in a sortal hierarchy. Thus, one can think of qualia structure as a way of linking a lexical item to the sortal hierarchy. Instead of simply having an is-a link, we have is-constitutive, is-formal, is-telic, and is-agentive links [Pustejovsky and Boguraev, 1993].

One use of such information would be to reduce the number of senses of verbs needed to get the right inferences. Consider the sentence in (15) from [Pustejovsky, 1991].

(15) Kim began a book

This sentence is ambiguous between meaning that Kim began to write a book and Kim began reading a book. If we assume that *begin* can combine with either *book*'s telic role or agentive role both of these readings fall out.

As with selectional restrictions, it seems that qualia structure provides a way to shortcut a more general domain knowledge inference chain. As with selectional restrictions, such shortcuts are often specific to a single domain. For example, if the Kim in (15) was a book binder, then the sentence would have another reading. The fact that books are bound could be incorporated into the Agentive role; however it seems unlikely that this reading should be present in other domains.

Up until this point we have been discussing nouns that pick out one thing independent of anything else in the domain: the noun *dog* will pick out the dog in the domain regardless of what else is there. However, there are other nouns, known as relational nouns in the linguistic literature, that pick out different elements in the domain depending on other elements in the domain. In [Barker and Dowty, 1993] they are defined as follows.

> In general, a relational noun is one such that an entity qualifies for membership in the extension of the noun only by virtue of there being a specific second entity which stands in a particular relation to the first, and where that relation is determined solely by the noun's lexical meaning.

One such noun is *father*; in some domains (where there is more than one father) you need to know who his children are. Some other nouns that are relational are *scratch* and *cut* (which are meaningless without the scratched or cut thing) and *window* and *door* (see [Pustejovsky and Anick, 1988; de Bruin and Scha, 1988;

Barker and Dowty, 1993]). Such relational nouns can be important in some domains. For example, databases that have information about workers and managers or commanders and military units need to handle relational nouns (*e.g.*, [de Bruin and Scha, 1988]). Another sort of relational noun is the eventive noun. Eventive nouns refer to events or situations; some examples are *the attack, the hit, the speech*. Eventive nouns relate an event to its participants in much the same way as a verb does—eventive nouns often take syntactic complements which map to semantic arguments (*e.g.*, *the attack on the town by the enemy*). In fact, many eventive nouns are nominalizations: nouns derived from verbs.

Eventive nouns are important to NLU for the same reason verbal argument structure is: to find out who did what to whom (see [Dahl *et al.*, 1987; Chinchor *et al.*, 1993]). Although there has been little work on eventive nouns in NLG, it seems likely that a system that could use nominalizations, to refer back to previously discussed events for example, would be able to produce better text. The next section contains an example (16) where knowledge of a nominalization and its relation to its verbal base makes generation possible. Similarly, the MT task could benefit from such knowledge. For example, if an eventive noun in a source language does not have a corresponding noun in the target language, it would be important to be able to relate an eventive noun and its arguments to the appropriate verb.

2.5 Relations between words

The two most common semantic relationships between words are antonym and synonym. Another might be the relation between a verb and the name of someone who does the verb habitually, *e.g.*, *sell/vendor, write/author*. Or the relation between a verb and a corresponding eventive noun, *e.g.*, *kiss/kiss, perform/performance, move/movement*.

Semantic relations such as antonym, synonym, and nominalized nouns can be used to provide multiple paths to verbalization [Iordanskaja *et al.*, 1991] within the NLG task.

> The ability of a generation model to provide paraphrase alternatives is not just a measure of its grammatical and lexical coverage. It is an integral part of its ability to produce compact, unambiguous and stylistically satisfying text. Texts in typically complex domains must meet a large number of constraints or optimize over a large set of preference functions. A linguistically rich model can provide a basis for "paraphrasing around" problems which arise when the most direct lexical and syntactic realization path runs into conflicting constraints, or scores poorly on the aggregate preference rating. [Iordanskaja *et al.*, 1991] p. 294

They give the following example (p. 307): Suppose that the verb *attack* has been chosen and the system attempts to generate (16a). It will find that there is a lexical gap in English: there is no adverb that magnifies the verb attack.

(16) a. A would-be penetrator attacked the system full-scale-ly(?)

b. A would-be penetrator mounted a full-scale attack on the system

However, if the system has access to eventive noun corresponding to $attack_V$ namely $attack_N$ then it could produce the sentence in (16b). Presumable MT systems would also gain from such an ability.

For many insights on semantic relations between words see the literature on lexical functions within Meaning Text Theory [Mel'čuk and Polguère, 1987; Haenelt and Wanner, 1992; Mel'čuk and Polguère, 1995].

2.6 General characteristics of the use of lexical semantic information

Consider the following observations about lexical semantic information.

- Lexical semantic information is often used in combination with other information. For example, aspectual information is only one of the pieces of information needed to determine the temporal ordering of events. Similarly, selectional restrictions must be combined with nominal lexical semantics and often other information to do word sense disambiguation. In addition, lexical choice in NLG often involves many different types of information.
- Lexical semantic information is used for multiple things by the same system. For example, nominal information is used for word sense disambiguation and for plan recognition in the TRAINS system and if the system had a full blown NLG system the information might also be used for lexical selection.
- Lexical semantic information often needs to "change its shape" to serve the given purpose. For example, the noun *plumber* might be classified as a human but a verb such as *eat* might restrict its subjects to animate objects. There must be a way to get from the fact that an object is human to the fact that it is animate. A more complicated example is that from the aspectual class of *make*, the sentence *the factory made the OJ* a system might need to know when the OJ came into existence; since *make* is telic, the OJ would come completely into existence at the end of the making event.

These observations support an important claim of this thesis: it is appropriate and beneficial to view the uses of lexical semantic information discussed in this chapter as applications of some form of deductive inference. The observations support this claim because (i) combining semantic information is what deductive inference is all about, (ii) deductive inferencing is a flexible general mechanism and can be used by many systems, (iii) deductive inferencing "changes the shape" of information.

It follows from this claim that the form of the information should lend itself to inference. This corollary is also supported by the fact that often the systems, with which NLP systems interface, perform inferencing themselves. For example, the output of natural language extraction systems is often used by inferencing systems [Palmer *et al.*, 1986; Herzog and Rollinger, 1991; Gomez, 1994]. Another example is NLG systems that generate text from expert systems. In order to support deductive inference, it will be argued in the next chapter that an expressive denotational logic should be used to represent lexical semantic information.

3 Representing Lexical Semantic Information

It was argued in the previous chapter that natural language processing applications require extensive inferencing. Many of these inferences are based on the meaning of particular open class[1] words. Providing a representation that can support such lexically-based inferences is a primary concern of this chapter.

The representation language of first order logic (FOL) has well-understood semantics and a multitude of inferencing systems have been implemented for it. Thus it is a prime candidate to serve as a lexical semantics representation. However, it will be argued that FOL, although a good starting point, needs to be extended before it can efficiently and concisely support all the lexically-based inferences needed.

Most lexical semantics representation systems utilize either KL-ONE-inspired terminological logics [Bobrow and Webber, 1980; Alshawi, 1987; Herzog and Rollinger, 1991; Kuhlen, 1983] or typed feature structure (TFS) logics [Copestake *et al.*, 1993; Copestake and Briscoe, 1992]. Representationally, terminological logics are subsets of FOL [Nebel and Smolka, 1991; Schmolze and Israel, 1983; Brachman and Levesque, 1984; Schubert, 1990; Hayes, 1979] as are TFS logics [Nebel and Smolka, 1991; Kasper and Rounds, 1986; Johnson, 1987; Smolka, 1991].[2] Thus, it is suggested here that lexical semanticists interested in supporting lexically-based inferences need to look for ways to enrich their representational systems. This suggestion has been made elsewhere (see [Burkert and Forster, 1992; Pletat, 1991; Mercer, 1992]). However, the specific extensions suggested here are novel to the lexical semantics literature.

Most of the examples on which the discussion below is based come from dialogues collected for the TRAINS project. The dialogues were collected by having

[1] Open class words are those that are either adjectives, adverbs, nouns, or verbs. Closed class words are prepositions, determiners, conjunctions, etc.

[2] It should be noted that many of the systems that use TFS logics, view their TFS representations as descriptions/short-hand for a more expressive semantic representation not as the representation itself [Nerbonne, 1993; Copestake and Briscoe, 1992; Rupp *et al.*, 1992]. The argument presented here is compatible with this position.

the role of the system played by a human (see [Gross, 1993]). As mentioned previously, the goal of the TRAINS project is to build a system that can assist a human manager who is attempting to solve a planning problem. A typical planning problem would be to deliver 1000 gallons of orange juice to a specific city by a certain time. To solve this problem the manager would be assisted by the system in scheduling the delivery of the oranges to the orange juice factory and the subsequent shipping of juice to the designated city. By relying mostly on examples taken from these dialogues, the relevance of the issues addressed to mundane, naturally occurring discourse is illustrated. Moreover, the dialogues provide a task-oriented context in which it is generally clear what inferences are required for understanding a given utterance; thus they provide a more constrained framework for semantic theorizing and experimentation than unrestricted texts or dialogues.

In addition to examples from the TRAINS dialogues, the extensions will also be supported by examples of lexical semantic information required to represent morphologically derived words.

Before particular representational extensions for lexically-based inferences can be introduced, a clearer idea of what are and what are not lexically-based inferences is needed. A lexically-based inference is one that depends on a lexical axiom. A lexical axiom is one that involves a semantic atom that is the translation of an open class word (assuming a meaning postulate approach). The following axiom is a lexical axiom and links the verb *enter* with its result state (using the predicate modifier *rstate*).

(17) $\forall x \forall y[\mathit{rstate}(\mathit{enter}(x,y)) \supset \mathit{contained\text{-}in}(x,y)]$

Using it, we could make a lexically-based inference from *the boxcar entered the factory* that the boxcar is in the factory. Furthermore, this inference is based on the word *enter* and not on the word *boxcar*. Such inferences can be contrasted with "structural" inferences such as in (18):

(18) there are at least three cities with orange juice factories and large train stations $\rightarrow$ there are at least three cities with orange juice factories

This depends on properties of certain classes of logical operators, specifically the class of upward monotone quantifiers [Barwise and Cooper, 1981] (*e.g.*, *at least three*) and the conjunction operator, rather than on the lexical semantics of specific open-class words. Note that if one substitutes the downward monotone quantifier *fewer than three* for *at least three* the inference no longer follows.

3.1 Extensions to FOL needed to support lexically-based inferences

This section introduces a number of extensions to FOL and provide examples that motivate them. More specifically, one should add

- restricted quantification and non-standard quantifiers,
- modal operators,
- predicate modification,
- and predicate nominalization.

These representational tools are available in some systems for sentence-level semantics [Hwang and Schubert, 1993a; Moore, 1981; Alshawi, 1989].

It should be noted at the outset that each example used to motivate an extension can be handled in FOL. However, the use of FOL leads to complex and unnatural paraphrases of intuitively simple facts, makes the encoded knowledge harder for system developers to comprehend and modify, and complicates inference. By adding a small amount of expressive power, concise and comprehensible representations can be given which facilitate efficient inferencing.

In our examples, *direct* and *indirect* motivation of specific extensions to FOL are distinguished; i.e., a lexical item may directly correspond to a type of operator (such as predicate modifiers) unavailable in FOL, and it may indirectly involve nonstandard operators through its axiomatization.

3.1.1 Non-standard quantifiers and restricted quantification

For this extension there seems to be little evidence from the TRAINS dialogues or derivational morphology. However, non-standard quantifiers and restricted quantification can be motivated from a more general perspective and are commonly used for representing sentence-level semantics.

Previously, when circumscribing lexically-based inferences, upward monotone quantifiers were mentioned. These include *at least three* (see (18)), *all, a few, most*, etc. Such examples motivate the augmentation of FOL with corresponding non-standard quantifiers; and the nominals with which they combine (as in *at least three <u>cities with orange juice factories</u>*) motivate the inclusion of formulas restricting the domains of the quantified variables.[3]. By utilizing these extensions,

[3]A number of terminological logics are sorted logics (*e.g.* [Pletat, 1991]). In sorted logics, the domain of a variable is restricted by the variable's sort. Representationally, this is much like restricted quantification, though more limited.

the following axiom enables inferences like the one in (18) to be made efficiently (*i.e.*, in one step).

(19) For all upward monotone quantifiers Q and all predicates P_1, P_2, and P_3:
$$Q\ x : P_1(x)\ [P_2(x) \wedge P_3(x)] \supset Q\ x : P_1(x)\ [P_2(x)]$$

Note that no reasoning about cardinality is required. This is in contrast to FOL: assuming a finite domain, something like the logical form in (20) for *there are at least three cities with orange juice factories and large train stations* and the axioms in (21) for cardinality would be required to support the information in FOL. It addition, multiple reasoning steps would be required.

(20) $$\begin{aligned} &\exists\ x[card(x) \geq 3 \wedge \\ &\quad \forall\ y[in(y,x) \supset [city(y) \wedge \\ &\qquad \exists z[have(y,z) \wedge ojfactory(z)] \wedge \\ &\qquad \exists z1[have(y,z1) \wedge trainstation(z1) \wedge large(z1)]]]]\end{aligned}$$

(21) a. $\forall x(card(x) = 0) \equiv \neg(\exists y\ in(y,x))$

b. $\forall x(card(x) = 1) \equiv [\exists y\ in(y,x) \wedge \forall z[in(z,x) \supset z = y]]$

c. $\forall x, y(card(x) = y \wedge y > 0) \equiv [\exists z[card(z) = 1 \wedge card(x - z) + 1 = card(x)]]$

As a direct motivation for a nonstandard quantifier syntax, the above argument pertains only to the closed category of determiners (in combination with certain adverbs and numeral adjectives). However, this syntax will also simplify the axiomatization of many open-class words. Consider the constructed example (22) and assume that the system has in its knowledge base that of ten cars, seven are tankers.

(22) M: Are the *majority* of the cars tankers?

It should be able to answer affirmatively. By using the axiom below and a suitable treatment of conjunction, the system could make use of the axioms for upward monotone quantifiers.

(23) $\forall a, b\ majority(a,b) \supset Most\ z : [z \in a]\ [z \in b]$

In the above axiom we assume that a and b denote collections and that $\in$ has been appropriately axiomatized.

Some other words that would benefit from non-standard quantifiers and restricted quantification are *scarce, rare, minority, scant,* and *predominate*. It also seems likely that degree adjectives such as *expensive, difficult*, or *intelligent* will require axiomatizations involving nonstandard, restricted quantifiers. For example, a *difficult problem* in the TRAINS domain is one that exacts more time and effort from the problem solver(s) than **most problems in this domain.** Similarly dispositional adjectives such as *perishable* or *fragile* and frequency adverbs such as *usually* also call for restricted non-standard quantification in their axiomatization, since they require a quantification other than over events of a certain type that is neither existential or universal.

3.1.2 Modal operators (or modal predicates)

Standard FOL has difficulty representing necessity, possibility and propositional attitudes. For example, it is not possible, directly, to differentiate a closed formula that could be true from one that could never be true. There is no built in conception of possible world nor is it possible to apply a predicate to a closed formula since a closed formula does not denote an individual in the ontology (it denotes a truth value). One can make worlds and propositions individuals of the ontology which correspond to terms in the logic and then axiomatize the relations between worlds, between propositions, and between closed formulas and propositions. However, this approach increases the complexity of the ontology and increases the complexity of the axiom set and thus makes it harder to maintain the system and to deduce things with it.

Modal logics were introduced to remedy this situation. They include operators whose interpretation involves possible worlds. And thus move the complexity out of the logic and syntactic proof system and into the model theory and interpretation function. A model structure with either possible worlds (*e.g.* [Montague, 1974]) or situations (*e.g.* [Barwise and Perry, 1983]) is required. Since the model theory for modal logics is well understood, this move is a win. For example, possibility can be encoded as an operator that applies to closed formulas and is interpreted as true if there exists a world accessible to the current one in which the formula is true.

Examples like (24) and (25) involve adverbs that are most naturally viewed as modal operators:

(24) M: That will probably work

(25) M: Maybe we'll get lucky again

In its context of occurrence, the second sentence refers to the possibility that all the orange juice needed for certain deliveries already exists, obviating the need for orange juice production. It would clearly be hazardous for the system to ignore the adverbs, turning wishful thinking into fact.

Such examples provide direct motivation for allowing modal operators in lexical semantics. The argument is weakened by the fact that modal adverbs are somewhat marginal as an open class of lexical items; but we can also argue from adjectives such as *reasonable, reliable, correct* and *right*, and verbs such as *found out that ..., said that ..., would like to ..., make sure ..., trying to ..., wonder if ..., believe*, and *assume*. Further comments are restricted here to some observed uses of *correct* and *reliable*. For instance, in the following request for confirmation, the system should interpret *correct* as applying to the proposition that the time is 2 pm:

(26) M: The time is two pm – is that correct?

Now if the system believes that the time is indeed 2 pm, it should infer an affirmative answer to the question – i.e., that it is *correct* that the time is 2 pm. Thus, for the relevant sense of *correct*, the lexical semantics should tell the system that for any closed formula ϕ,

(27) $correct(\phi) \equiv \phi$.

If we adopt such a schema for the meaning of *correct*, we are treating it as a modal operator since it is being applied to a closed formula not a term. An alternative is to assume that *correct* is a predicate, but one that applies to reified *propositions*. In turn, such an approach calls for the introduction of a reifying operator (such as *that*) for converting sentence contents (propositions) into individuals, allowing their use as predicate arguments. In either case, we are introducing a modal extension to FOL. The case of *reliable* is similar but more subtle. In actually occurring examples this property is often ascribed to items of information:

(28) M: That's reliable information

Intuitively, reliable information is not necessarily correct, though it is necessarily well-founded (i.e., there are good reasons for the presumption of truth). So the axiomatization is less trivial than (27), but it still calls for use of a modal operator or modal predicate in the same way.

Concerning indirect motivation for modals, an interesting example is *compatible (with)*, as used below.

(29) M: So that sounds like a good temporary plan – let's see if it's compatible with our next objective here which is to deliver a boxcar of bananas to Corning

In order for the plan in (29) to be compatible with the additional banana delivery, it must be *possible* to realize both action types (within the given temporal and other constraints). In general,

(30) $\forall x, y[[action\text{-}type(x) \wedge action\text{-}type(y) \wedge compatible\text{-}with(x,y)] \supset$
$\Diamond \exists x', y'[realize(x', x) \wedge realize(y', y)]]$.

$\Diamond \Phi$ entails that there exists a situation or possible world sufficiently connected to the current one where Φ is true.

The suffix *-able* also provides indirect evidence for modal operators. A adjective derived using *-able* from a verb entails that it is possible to perform the action denoted by the verb, *e.g.*, something is enforceable if it is possible that something can enforce it [Dowty, 1979]. Dowty uses the $\Diamond$ operator in his treatment of the semantics of the suffix *-able* as shown below.

(31) For adjectival predicates p_{adj} derived from verbs p_v using *-able*
$\forall x[p_{adj}(x) \supset \Diamond \exists y[p_v(y, x)]]$

3.1.3 Predicate modification

By predicate modification the transformation of one predicate into another is meant. Within a general setting for language understanding, the case for allowing predicate modifying operators is most easily made by pointing to non-intersective attributive adjectives such as *former, cancelled, fake, supposed, simulated*, or *fictitious*. For instance, applying *cancelled* to an event nominal such as *trip* yields a new predicate which is *not* true of actual trips, and so should not be analyzed as $cancelled(x) \wedge trip(x)$, the straightforward FOL encoding. An FOL treatment that fares better is to reify predicates corresponding to nouns and then use a predicate such as *hasprop* to relate such properties to objects that have them: $hasprop(Trip, x)$. Adjectives such as *cancelled* would then be functions from terms to terms: $hasprop(Cancelled(Trip), x)$. Axioms such as (32) would partially define such functions.

(32) $\forall x, e\ hasprop(Cancelled(x), e) \supset \neg hasprop(Actual(x), e)$

However, this treatment also has problems. Consider the pair of sentences below.

(33) a. Kim wrestled skillfully with Pat

b. Pat wrestled awkwardly with Kim

Using this treatment, the wrestling event would be both skillful and awkward at the same time. A workaround also exists for this problem but introduces further complexity. In contrast, treating modifiers like *skillfully* as predicate modifiers produces a more straightforward solution: $Skillfully(wrestled\text{-}with)(Pat)(Kim)$ vs. $Awkwardly(wrestled\text{-}with)(Kim)(Pat)$.

Such adjectives and adverbs do not occur in the TRAINS dialogues collected so far. However, verbs such as (*make, get, look, sound, seem, begin, construct*) do occur and also provide direct motivation for predicate modifiers.

For instance, the dialogues contain instances where the manager asks

(34) M: Does that sound reasonable?

(referring to a plan), or comments

(35) M: Problem number two looks difficult.

A plan can *sound* reasonable even if more careful analysis reveals it to be *un*reasonable. So the system should realize that an affirmative response to the query merely requires the absence of *obvious* flaws in the plan (detectable with limited inferential effort), rather than an actual proof of correctness.

One could attempt to handle such locutions by decomposing them into more complex modal patterns; e.g., x *sounds* P, for P a predicate, might be decomposed into something like "when one considers x, one (initially) feels that x is P". This is precisely the strategy that has often been suggested for intensional verbs such as *seeks*. But while plausible definitions (decompositions) exist for some intensional verbs, they are very difficult to contrive for ones like *resemble* (as in *the one-horned goat resembled a unicorn*) or *imagine*. A more general, straightforward approach is to add predicate modifiers to FOL. Thus the translation of *sounds* (when it takes an adjectival complement) would be a predicate modifier, whose meaning is constrained by axioms like the following:

(36) For all monadic predicates P:
$\forall x sounds(P)(x) \supset$
$\forall s,t[person(s) \wedge consider(s,x,t) \supset feel\text{-}that(s,P(x),end\text{-}of(t))]$

where various subtleties are being neglecting for the sake of exposition.[4] Thus to answer (34), the system would make use of (36) to infer that it need only "consider" the plan in question, until it "feels-that" (i.e., *tentatively* concludes that) the plan is reasonable or otherwise.

A third class of examples also directly motivating predicate modifiers, namely certain VP adverbs such as *almost, nearly* and *apparently.* Again, these do not appear (as yet) in the TRAINS corpus, but are common in other corpora. For example, the "pear stories" of Chafe [Chafe, 1980] contain examples such as "[the

[4]This is, of course, the Montagovian approach, though Montague's intension operator is being dispensed with (writing $sounds(P)(x)$ rather than $sounds(^\wedge P)(x)$) by relying on a slight departure from standard intensional semantics that treats the world (or situation) argument as the last, rather than first, argument of the semantic value of a predicate [Hwang and Schubert, 1993b].

boy on the bicycle] *almost* ran into a girl", where the desired inference is that he did *not* run into her, but came very close to her.

The prefixes *co-, pre-, post-*, and *mid-* can also be handled straightforwardly by representing them as predicate modifiers. Some examples are *co-captain, co-lead, pre-primary, pre-payment, mid-April*, and *mid-continent.* In addition, many verbal affixes have meanings that are much easier to axiomatize when predicate modifiers are available. For example, the suffix *-ize* produces verbs where their result state is the adjectival base, *e.g.*, the result of formalizing something is that it is formal. The axiom for this information is listed in (37)

(37) For predicates p produced by *ize*
$\forall y[rstate(p)(y) \supset base(p)]$

The predicate modifier *rstate* produces the result state that holds at the end of events described by the verb and the predicate modifier *base* produces the adjectival base predicate from the derived verb predicate.

3.1.4 Predicate nominalization

In natural language, many types of expressions can be turned into noun-like entities. For example, in (38b), *that* turns the statement *the car was blue* into something that can play the same role as *car* in (38a).

(38) a. the car made Mary happy

b. that <u>the car was blue</u> made Mary happy

This section will argue that FOL should be extended with the operators that nominalize predicates. Such operators form terms (denoting individuals in the domain of discourse) from predicates (denoting classes of objects, actions or events). In other words, predicate nominalization involves the *reification* of properties (including *kinds/species*, *action types*, and *event types*) by application of nominalizing (reifying) operators.

The line of argument for utilizing such operators for representing lexical semantics is less direct than for the previous extensions. It is claimed that (i) many words denote predicates with one or more arguments ranging over kinds of things, properties, and actions/events (this already came up incidentally in (29)); and (ii) the lexical axioms for these predicates will either explicitly involve nominalized predicates or require the substitution of nominalized predicates when used for inference.

As examples of a variety of lexical predicates over (reified) action types, consider the italicized words in the following TRAINS excerpts. Corresponding action-type arguments where these are explicitly present have been underlined.

(39)M: What is the best *way* for me to accomplish my *task* ...

(40) S: <u>That</u>'s a little beyond my *abilities*

(41) S: The *way* it's going to work is, <u>engine E2 is going to go to city E</u> ...

(42) S: Our current *plan* is <u>to fill ... tankers T3 and T4 with beer ...</u>

S: One other *suggestion* would be that <u>you take the other tanker which isn't being used</u> ...

(43)M: <u>That</u>'s not gonna *work*

(44) S: Well that will *delay* <u>departure</u>

(45) S: Right, we can *begin* <u>production</u> ...

(46)M: ... send it off on a particular route and *do* <u>it</u> several times

Way, task, plan, and *suggestion* as used in (39 - 42) are predicates over types of actions or events, as the underlined arguments confirm. For instance, the action descriptions underlined in (41) and (42) do not refer to particular future actions at particular times, but to types of actions whose eventual *realization* is hoped to solve the problem at hand. (And the ability deictically referred to in (40) is the ability <u>to specify the best way for the manager</u> <u>to accomplish the current task</u> in (39) – again an action type.) Similarly (43 - 46) illustrate verbs whose subject or object ranges over action types. Note for instance in (44) that a *particular* departure event cannot be delayed – particular events have fixed times of occurrence, but event types in general do not. Likewise in (46), only an action type, not a particular action, can be done "several times".

Similarly the following excerpts contain predicates over kinds/species, again with corresponding arguments underlined:

(47)M: <u>One boxcar of oranges</u> is *enough* to produce the required amount of orange juice

(48)M: And *fill* two tankers *with* <u>beer</u>

(49) M: There's [an] unlimited *source of* malt and hops ...

Note that in (47) the phrase predicated by *enough* refers to a *kind* of load or quantity, not to any *particular* load. Similarly the underlined objects of *fill with* and *source of* in (48) and (49) are *kinds* of stuff, not particular realizations of them. (In fact, no particular batch of malt and hops could be "unlimited".)

Turning to the second step of the argument, concerning the explicit occurrence of nominalization operators in argument positions of predicates like those above, consider the sense of *do* with an action type as object, as in (46). To understand (46), the system will have to substitute a term for the action type, "send [the train] off on a particular route", for the pronoun. To infer any further consequences, it will need a meaning postulate something like the following:

(50) For all monadic action predicates P:
$\forall x\ do(Ka(P))(x) \supset P(x)$

where Ka reifies an action predicate (in this case, "send [the train] off on a particular route"). It can then apply semantic and world knowledge about P to draw conclusions about the effects of $P(x)$ (in this case that the train will follow the route in question and reach its destination).

3.2 Conclusion

Many NLP tasks require lexicons that can support extensive lexically-based inferencing. In order to support these inferences, representations for lexical semantics will have to be richer than they are now. This chapter has motivated for particular extensions to FOL using examples drawn from actual dialogues in the TRAINS domain. Although these extensions are not new, their motivation based on the semantics of open class words is.

These extensions are a subset of the extensions that are available in Episodic Logic (EL) [Hwang and Schubert, 1993a; Hwang and Schubert, 1993b], a logic designed to be expressively and inferentially adequate as both a logical form for natural language and as a general representation for commonsense knowledge. Similar formalisms are those used in the Core Language Engine [Alshawi, 1987; Alshawi, 1989] and Nerbonne and Laubsch's NLL [Laubsch and Nerbonne, 1991; Nerbonne, 1992]. EL is an intensional, situational extension of FOL that provides a systematic syntax and formal semantics for sentence and predicate nominalization (reification), sentence and predicate modification, nonstandard restricted quantifiers, λ-abstraction, and pairing of arbitrary sentences with situation- (episode-) denoting terms (where those sentences are interpreted as describing or characterizing those situations).

Inference in EL has been shown to be practical through the EPILOG implementation [Schaeffer *et al.*, 1993], with examples ranging from fairy tale fragments and aircraft maintenance reports [Hwang and Schubert, 1993a; Hwang and Schubert, 1993b] to the Steamroller theorem-proving problem. In addition, EL as been used as the front-end logical form in the TRAINS system [Allen *et al.*, 1994]. A conclusion from the text understanding experiments is that increased expressiveness often simplifies inference, allowing conclusions to be drawn in one or two steps that would require numerous steps in an FOL "reduction" of the same information.

4 Morphological Cues for Lexical Semantic Information

The first two chapters of this thesis discussed what fine-grained lexical semantic information is and why it is needed. The representation of lexical semantic information was discussed in the previous chapter. This chapter discusses the acquisition of such information and describes a method based on morphological cues. The chapter starts by discussing three approaches to acquisition that have been debated in the human language acquisition literature: perceptual cueing, surface cueing, and language semantics cueing. It then discusses these approaches from a computational linguistics perspective. Finally, morphological cueing, a type of surface cueing, is introduced. Results derived from an implementation are presented and the method is evaluated on the basis of these results.

4.1 Three approaches to the acquisition of lexical semantic information

In recent years there has been a fruitful dialectic between the supporters of perceptual cueing of verb semantics and those supporting surface cueing of verb semantics. [Grimshaw, 1981; Pinker, 1989; Pinker, 1994] advocate perceptual cueing and [Gleitman, 1990; Fisher *et al.*, 1991; Naigles *et al.*, 1989] surface cueing.[1] This dialectic will be used as a vehicle to introduce the different approaches.

Perceptual cueing is based on the intuitively appealing idea that humans acquire the meanings of words by relating them to semantic representations resulting from perceptual or cognitive processing. For example, in a situation where the father says *Kim is throwing the ball* and points at Kim who is throwing the ball, a child might be able learn what *throw* and *ball* mean.

However, [Gleitman, 1990] argues that a language learner whose sole learning mechanism is perceptual cueing will have difficulty learning the meaning of verbs because of the following:

[1] Many of the articles relevant to this debate are collected in [Gleitman and Landau, 1994].

1. The event being referred to by the verb is often not present when the verb is uttered. The results of experiments suggest that it is often the case that a verb is uttered when the participants of the event described by the verb are not physically present.

2. There is an overabundance of salient features of the world that could be part of the semantics of an uttered verb.

   "For instance, over the discourse as a whole, probably the mother has different aims in mind when she tells the child to 'look at' some object than when she tells her to 'hold' or 'give' it. The child could code the observed world for these perceived aims and enter these properties as aspects of the words' meanings. But also the mother may be angry or distant or lying down or eating lunch and the object in motion may be furry or alive or large or slimy or hot, an the child may code for these properties of the situation as well, entering them too, as facets of the words' meanings." ([Gleitman, 1990], p. 10)

3. There are an overabundance of events in any given situation. Using an example from [Gleitman, 1990], whenever someone pushes a toy truck that truck is also moving. Thus there is the pushing event and the moving event. Thus using cross-situational observation will be difficult. This problem has two especially bad cases.

   (a) There are events that always occur together and depend on the perspective of the observer such as buying and selling, fleeing and chasing, and winning and beating (these examples were taken from [Gleitman, 1990]). Cross-situational observation will not help since in every situation where there is buying, there is selling.

   (b) Some events are more specific than others: whenever someone taps something she also touches it. If the learner chooses "touch" as the meaning of *tap* then cross-situational observation will not help correct this mistake since in every situation where there is tapping, there is touching.

4. Some verbs, such as *think* and *doubt* refer to events that are not directly observable and thus if the learner is basing its meanings on the observed alone it cannot acquire these verbs. In Gleitman's words, "If the child is to learn the meanings from perceptual discrimination in the real world, the primitive vocabulary of infant perception has to be pretty narrow to bring the number and variety of data storing and manipulative procedures under control. But no such narrow vocabulary of perception could possibly select the thinkingness properties from events. I conclude that an *unaided* observation-based verb-learning theory is untenable because it could not acquire *think*" (Gleitman's emphasis) ([Gleitman, 1990] p. 18).

To help the perceptual semantic cueing mechanism produce better results, Gleitman suggests that a form of surface cueing be used to restrict the events and features to which the learner attends. In her own words, "children's sophisticated perceptual conceptual capacities yield a good many possibilities for interpreting a scene, but the syntax acts as a kind of mental zoom lens for fixing on the interpretation, among these possible ones that the speaker is expressing" ([Gleitman, 1990], p.21). More specifically, she suggests that the subcategorization frames of verbs correspond to "... certain semantic elements such as 'cause,' 'transfer,' 'symmetry,' and 'cognition' ..." ([Gleitman, 1990] p.29) and that this correspondence can be used to acquire lexical semantic information about verbs. This is just the information needed to reduce the search space for the perceptual cueing mechanism.

She refers to this approach as "syntactic bootstrapping" but in this thesis it will be referred to as syntactic cueing, a form of surface cueing. Surface cueing does not require any semantic information related to perceptual input or the input utterance. Instead it makes use of fixed correspondences between surface characteristics of language input and lexical semantic information: surface characteristics serve as cues for lexical semantic information. For example, if a verb is seen with a noun phrase subject and a sentential complement, it often has semantics involving spatial perception and cognition, *e.g.*, *believe, think, worry*, and *see* [Fisher *et al.*, 1991; Gleitman, 1990].

[Pinker, 1994] replies that although syntactic cueing could be used to zoom in on relevant aspects of a situation to which an utterance corresponds, the perceptual semantic cueing mechanism does not require this information. In addition, he claims that the semantic information provided by syntactic cueing is peripheral and only applies to the use of the verb in the particular frame from which the information was derived.

> "I conclude that attention to a verb's syntactic frame can help narrow down the child's interpretation of the perspective meaning of the verb in that frame, but disagree with the claim that there is some in-principle limitation in learning a verb's content from its situations of use that could only be resolved by using the verb's set of subcategorization frames." ([Pinker, 1994], p. 379)

[Pinker, 1994] specifically addresses each of [Gleitman, 1990]'s arguments listed above. Only some of his major points are presented here. One point he makes is that perceptual cueing involves using semantic representations produced by both perceptual processing *and* cognitive processing. Thus, one would not expect the learner to only attend to events that are perceivable at the time of the utterance. Instead, all of the past, present, and possible events conceivable by the learner for both the physical and the mental "world" could be used as a basis for deriving lexical semantic information for an uttered verb. This point addresses Gleitman's concerns about the fact that a verb is often uttered when the the participants of the event described by the verb are not physically present. In addition, it addresses the problems [Gleitman, 1990] claims perceptual cueing has with words like *believe.*

Another point Pinker makes is that comparing and contrasting the situations in which a verb has been heard does provide, in principle, enough information to tease apart the meaning of verbs like *push/move*, *buy/sell* and *touch/tap*. This point is is based on two claims for which there is a fair amount of supporting data: language learners have a very rich representational system and different verbs never have exactly the same semantics (no true synonyms). Thus, since the learner is always trying to give different verbs different meanings, the learner will attend to different usages. To use one of Pinker's examples, one can *buy* a soda from a coke machine even though the machine did not *sell* it. Using these sorts of situations and the assumption that *buy* and *sell* mean different things, the learner could tease apart their meanings.

While arguing against a particular set of experiments designed to support the argument for surface cueing, [Pinker, 1994] discusses the third approach to the acquisition of lexical semantics that will be discussed in this section. This approach is based on the idea that the semantics of an unknown word can be derived from the elements of the utterance that are known by the learner. Consider the following example.

> "... if someone were to hear *I glipped the paper to shreds* or *I filped the delicious sandwich and now I'm full,* presumably he or she could figure out that *glip* means something like 'tear' and *filp* means something like 'eat'. But although these inferences are highly specific and accurate, no thanks are due to the verbs' syntactic frames (in this case, transitive). Rather, we know what those verbs mean because of the semantics of *paper, shreds, sandwich, delicious, full,* and the partial syntactic analysis that links them together (partial, because it can proceed in the absence of knowledge of the specific subcategorization requirements of the verb, which is the data source appealed to by Gleitman)." ([Pinker, 1994] p. 382)

This approach will be referred to here as language semantics cueing. Pinker notes that in some cases what looks like syntactic cueing may in fact be language semantics cueing.

To summarize, three approaches have been introduced: perceptual cueing, surface cueing and language semantics cueing. These approaches differ in that their input is different: surface cueing takes non-semantic representations derived by language processing, perceptual cueing takes semantic representations derived by perceptual processing and/or cognitive processing along with the corresponding uttered verb, and language semantics cueing takes semantic representations derived directly from the utterance.

4.2 A computational linguistics perspective

It seems likely that perceptual cueing is the central method human children use to acquire lexical semantics. However, key modules are currently lacking that would be needed for a large scale computer implementation, namely perceptual and conceptual processing modules complex enough to produce the semantic information perceptual cueing requires as input. There does not yet exist a robot that can interact with and reason about the world with the complexity required. Accordingly, the research projects that fall into this class [Pustejovsky, 1988; Siskind, 1990; Webster and Marcus, 1989] do not claim to provide a method for acquiring a large NLP lexicon. Instead they are meant as proof of concept studies with an aim at explaining human language acquisition.

Language semantics cueing is more promising from a computational perspective and consequently has a long (for computational linguistics) history. The acquisition systems described in [Granger, 1977; Selfridge, 1980] use scripts [Schank and Abelson, 1977] to represent the semantics of a passage and then use various heuristics to fit an unknown word into a slot of the script and then use the semantic restrictions corresponding to the slot to infer information about the word. For example, if the word *garçon* was used in a passage that could be matched to a restaurant script and *garçon* could be fit into the slot for the taker of the order then one could infer that a *garçon* is a person. [Berwick, 1983] presents a system that is similar but uses a terminological logic representation system. More recently [Hastings, 1994] presents a system that uses verb selectional restrictions to assign unknown nouns to noun classes and noun semantic classes to assign unknown verbs selectional restrictions. For example, *to bomb* restricts its direct object to be a `Human-or-Place` and thus if an unknown word is encountered as the direct object of *to bomb* then the system can deduce that it is a member of the `Human-or-Place` class and possibly a subclass. The constraints would work in the opposite direction if the noun was known and the verb unknown. Such systems are able to acquire very fine-grained lexical semantic information. However, a central problem for these approaches is that they rely on having lexical semantic information to begin with. If they do not have enough seed information, then they do not have enough constraints to effectively infer information for unknown words.

The most attractive feature of surface cueing is that it does not, at least directly, rely on any lexical semantic information. Often the surface cues can be identified relatively easily. For example, [Hearst, 1992] describes a system that searches for syntactic patterns to find semantic relations. One pattern the system uses is $NP, NP *$ *,and other* NP and this pattern cues a hyponym relationship between the initial noun phrases and the final one: *temples, treasuries, and other important civic buildings.* Note that the lexical semantic information is inferred from a single token. [Pustejovsky *et al.*; Jacobs and Zernik, 1988; Dorr and Lee, 1992; Velardi *et al.*, 1991; Andry *et al.*, 1995] describe similar systems.

Another type of surface cueing does not base its inferences on any single token. Instead it makes inferences based on statistical measures of distributional similarity. The methods differ on what information they base their measures of distributional similarity and on the statistical measure they employ. One method described in [Hindle, 1990] bases its measure of distributional similarity on syntactic predicate argument relations and employs mutual information as a statistical measure. Thus two nouns are judged similar if they are likely to occur (or not occur) as direct objects of the same verbs. More accurately, the comparison of nouns is based on the mutual information between verbs and nouns where mutual information corresponds roughly to the amount that the presence of one word affects the probability of the other. Thus, *drink* and *beer* have a high mutual information in most corpora. One set of nouns that had a high similarity in Hindle's tests was *boat, ship, plane, bus, jet, vessel, truck, helicopter, ferry*, and *man*. Some of the verbs that had high mutual information measures with these nouns were *hijack, charter, intercept*, and *park*. Some methods that use predicate argument structure but different statistical measures are [Rooth, 1995] which uses a latent class language model and Baum-Welsh reestimation to set the parameters, [Pereira *et al.*, 1993] which uses deterministic annealing to cluster words, and [Grishman and Sterling, 1993] which uses co-occurrence smoothing. [Schuetze, 1992a] describes a method that bases its measure of distributional similarity on co-occurrence in a sentence and applies single value decomposition to the co-occurrence matrix and measures the similarity of resulting vectors to measure similarity.

Whether statistical or non-statistical, the main advantage of the surface cueing approach is that the input required is currently available: there is an ever increasing supply of online text, which can be automatically part-of-speech tagged, assigned shallow syntactic structure by robust partial parsing systems, and morphologically analyzed, all without any prior lexical semantics. And it is the method that will be pursued further here.

A possible disadvantage of surface cueing is that surface cues for a particular piece of lexical semantics might be difficult to uncover or they might not exist at all. In addition, the cues might not be present for the words of interest. Thus, it is an empirical question what subset of the needed lexical semantic information corresponds to easily identifiable abundant surface cues. The rest of this chapter explores the possibility of using derivational affixes as surface cues for lexical semantics.

4.3 Morphological cueing

Many derivational affixes only apply to bases with certain semantic characteristics and only produce derived forms with certain semantic characteristics. For example, the verbal prefix *un-* applies to telic verbs and produces telic derived forms. Thus, it is possible to use *un-* as a cue for telicity. By searching a sufficiently large corpus we should be able to identify a number of telic verbs. Examples from the Brown corpus include *clasp, coil, fasten, lace*, and *screw*.

A more implementation-oriented description of the process is the following: (i) collect a large corpus of text, (ii) tag it with part-of-speech tags, (iii) morphologically analyze its words, (iv) assign word senses to the base and the derived forms of these analyses, and (v) use this morphological structure to assign semantics to both the base senses and the derived form senses. Tagging the corpus is necessary to make word sense disambiguation and morphological analysis easier. Word sense disambiguation is necessary because one needs to know which sense of the base is involved in a particular derived form, more specifically, to which sense should one assign the feature cued by the affix. For example, *stress* can be either a noun *the stress on the third syllable* or a verb *the advisor stressed the importance of finishing quickly.* Since the suffix *-ful* applies to nominal bases, only a noun reading is possible as the stem of *stressful* and thus one would attach the lexical semantics cued by *-ful* to the noun sense. However, *stress* has multiple readings even as a noun: it also has the reading exemplified by *the new parent was under a lot of stress.* Only this reading is possible for *stressful.*

In order to produce the results presented in the next section, the above steps were performed as follows. The Penn Treebank version of the Brown corpus [Marcus *et al.*, 1993] served as the corpus. Only its words and part-of-speech tags were utilized. Although these tags were corrected by hand, part-of-speech tagging can be automatically performed with an error rate of 3 to 4 percent [Merialdo, 1994; Brill, 1994]. The Alvey morphological analyzer [Ritchie *et al.*, 1992b] was used to assign morphological structure. It uses a lexicon with just over 62,000 entries. This lexicon was derived from a machine readable dictionary but contains no semantic information. No word sense disambiguation was performed for the bases and derived forms. However, there exist systems for word sense disambiguation which do not require explicit lexical semantic information [Yarowsky, 1993; Schuetze, 1992b]. Because of the lack of word senses, the semantics corresponding to the morphological structure of the derivation was assigned to each normalized word (*i.e.*, uninflected) involved in the derivation and the semantics assigned to a particular word was considered correct if it held for all senses present in the corpus. Such scoring requires that the semantics are expressed in a precise way. The cued lexical semantic information are axiomatized using Episodic Logic [Hwang and Schubert, 1993a] which was mentioned in the previous chapter.

Let us consider an example. The suffix *-ize* applies to adjectival bases (*e.g.*, *centralize*). There is a semantic distinction that can be made between this affix and the *-ize* that applies to nouns (*e.g.*, *glamorize*). The first affix will be referred to as *-Aize* and the second as *-Nize*. First, the regular expressions "`.*IZ(E|ING|ES|ED)$`" and "`^V.*`" are used to collect tokens from the corpus that were likely to be derived using *-ize*. The Alvey morphological analyzer is then applied to each type and each of the analyses it produces is considered. It strips off *-Aize* from a word form if it can find an entry with a reference form of the appropriate orthographic shape and has the features "uninflected," "latinate," and "adjective." It may also build an appropriate base using other affixes, *e.g.*, [[*tradition -al*] *-Aize*]. If this also fails, it will attempt to construct a base on its own. Finally, the derived forms are assigned the lexical semantic feature COS (change-of-state) defined by the axiom below.

```
For all predicates P with features COS and DYADIC:
(∀x,y,e [[[x P y]**e] -> [(∃e1:  [[e1 at-end-of e] ∧
                                   [e cause e1]]
                                  [[y (rstate P)]**e1]) ∧
                           (∃e2:  [e2 at-beginning-of e]
                                  [(¬[y (rstate P)])**e2])]])
```

In this axiom and throughout the remaining of this dissertation, lisp-like bracketing conventions are used with the addition of square brackets to indicate that the operator is infixed as is the convention in Episodic Logic. As mentioned in the first, the operator `**` is analogous to $\models$ in situation semantics; it indicates that a formula describes an event. The axiom states that if a change-of-state predicate describes an event, then the result state of this predicate holds at the end of this event and that it did not hold at the beginning, *e.g.*, if one wants to formalize something it must be non-formal to begin with and will be formal after. The result state of an *-Aize* predicate is the predicate corresponding to its base; this is stated in another axiom (see Section 4.3.1). `P` is a place holder for the semantic predicate corresponding to the word sense which has the feature. It is assumed that each word sense corresponds to a single semantic predicate. In addition, note that the axiom only uses a one-way entailment and thus it only constrains the meaning of the predicate; it does not define it.[2] As mentioned above, in this study a word is considered to have a feature if its axiom is true of all the word senses present in

[2] This method of assigning every word a unique predicate and then constraining the meaning of the predicates with one-way meaning postulates was suggested in [Chierchia and McConnell-Ginet, 1990]. This scheme avoids some of the problems that a decompositional approach has. For example, a decompositional approach might have *formalize* denote **λx,y [x (cause (become formal)) y]**. There are two problems with this approach. First, it can introduce spurious scoping ambiguities since there are now a number of operators in the logical form and modifiers might be scoped to apply to only part of the word. It appears that such internal word scopings do not exist. Second, unwanted inferences could occur since the meanings of the words are logically equivalent to the decompositions. For example, *formalize* is being defined as **[x (cause**

the corpus. As shown in Table 1, there were 63 *-Aize* derived forms of which 78% conformed to the COS axiom.

Precision figures for the method were collected as follows. The method returns a set of normalized (*i.e.*, uninflected) word/feature pairs. A human then determines which pairs are "correct" where correct means that the axiom defining the feature holds for the instances (tokens) of the word (type). Because of the lack of word senses, the semantics assigned to a particular word is only considered correct, if it holds for all senses occurring in the relevant derived word tokens.[3] For example, the axiom above must hold for all senses of *centralize* occurring in the corpus in order for the *centralize*/ COS pair to be correct. The axiom for IZE-DEP must hold only for those senses of *central* that occur in the tokens of *centralize* for the *central*/IZE-DEP pair to be correct. This definition of correct was constructed, in part, to make relatively quick human judgements possible. It should also be noted that the semantic judgements require that the semantics be expressed in a precise way. This discipline is enforced in part by requiring that the features be axiomatized in a denotational logic. Another argument for such an axiomatization is that many NLP systems utilize a denotational logic for representing semantic information and thus the axioms provide a straightforward interface to the lexicon.

To return to our example, as shown in Table 1, there were 63 *-Aize* derived words (types) of which 78 percent conform to the COS axiom. Of the bases, 80 percent conform to the IZE-DEP axiom which will be discussed in the next section. Among the conforming words were *equalize, stabilize,* and *federalize.* Two words that seem to be derived using the *-ize* suffix but do not conform to the COS axiom are *penalize* and *socialize (with the guests).* A different sort of non-conformity is produced when the morphological analyzer finds a spurious parse. For example, it analyzed *subsidize* as *[sub- [side -ize]]* and thus produced the *sidize*/COS pair which for the relevant tokens was incorrect. In the first sort, the non-conformity arises because the cue does not always correspond to the relevant lexical semantic information. In the second sort, the non-conformity arises because a cue has been found where one does not exist. A system that utilizes a lexicon so constructed is interested primarily in the overall precision of the information contained within and thus the results presented in the next section conflate these two types of false positives.

`(become formal)) y]` which might also be the logical form for *X caused Y to become formal.* Although the meanings are similar, it is unlikely that they are logically equivalent (see [Fodor, 1970]).

[3]Although this definition is required for many cases, in the vast majority of the cases, the derived form and its base have only one possible sense (*e.g.*, *stressful*).

4.3.1 Results

This section starts by discussing the semantics of 18 derivational affixes: *re-, un-, de-, -Aize, -Nize, -en, -Aify, -Nify, -le, -ate, -ee, -er, -ant, -age, -ment, mis-, -able, -ful, -less*, and *-ness*. (These affixes were chosen primarily because they provide a range of lexical semantic information. In addition, the author has a less-than-complete background in semantics and thus many affixes were excluded from the list simply because he was not familiar with with area of semantics with which the affixes dealt. The important point is that these affixes where not chosen on the basis of their precision scores and thus the precision numbers presented here should not be seen as an upper bound.) Following the discussion of these affixes, a table of precision statistics for the performance of these surface cues is presented. The word types on which these statistics are based are listed in the appendix along with other information about the actual implementation. The lexical semantics cued by these affixes are loosely specified here but are axiomatized in Chapter 6 in a fashion exemplified by the COS axiom above. In addition, the semantic analyses represented by these axioms are supported in Chapter 6.

The verbal prefixes *un-, de-*, and *re-* cue aspectual information for their base and derived forms. Some examples from the Brown corpus are *unfasten, unwind, decompose, defocus, reactivate*, and *readapt.* Above, it was noted that *un-* is a cue for telicity. In fact, both *un-* and *de-* cue the change-of-state (COS) feature for their base and derived forms—the COS feature entails the TELIC feature. In addition, for *un-* and *de-*, the result state of the derived form is the negation of the result state of the base (BASE-NEG-RSTATE), *e.g.*, the result of unfastening something is the opposite of the result of fastening it. As shown by examples like *reswim the last lap*, *re-* only cues the TELIC feature for its base and derived forms: the lap might have been swum previously and thus the negation of the result state does not have to have held previously [Dowty, 1979]. For *re-*, the result state of the derived form is the same as that of the base (BASE-RSTATE), *e.g.*, the result of reactivating something is the same as activating it. In fact, if one reactivates something then it is also being activated: the derived form entails the base (ENTAIL-BASE). Finally, for *re-*, the derived form entails that its result state held previously, *e.g.*, if one recentralizes something then it must have been central at some point previous to the event of recentralization (REPRESUP) [Light, 1992].

The suffixes *-Aize, -Nize, -en, -Aify, -Nify* all cue the COS feature for their derived form as was discussed for *-Aize* above. Some exemplars are *centralize, formalize, categorize, colonize, brighten, stiffen, falsify, intensify, mummify*, and *glorify.* For *-Aize, -en* and *-Aify* we can say a bit more about the result state: it is the base predicate (RSTATE-EQ-BASE), *e.g.*, the result of formalizing something is that it is formal. Finally *-Aize, -en*, and *-Aify* cue the following feature for their bases: if a state holds of some individual then either an event described by the derived form predicate occurred previously or the predicate was always true of the individual (IZE-DEP), *e.g.*, if something is central then either it was centralized or it was always central.

The "suffixes" *-le* and *-ate* should really be called verbal endings since they are not suffixes in English, *i.e.*, if one strips them off one is seldom left with a word. (Consequently, only regular expressions were used to collect types; the morphological analyzer was not used.) Nonetheless, they cue lexical semantics and are easily identified. Some examples are *chuckle, dangle, alleviate*, and *assimilate.* The ending *-ate* cues a COS verb and *-le* an ACTIVITY verb.

The derived forms produced by *-ee, -er*, and *-ant* all refer to participants of an event described by their base (PART-IN-E). Some examples are *appointee, deportee, blower, campaigner, assailant*, and *claimant.* In addition, the derived form of *-ee* is also sentient of this event and non-volitional with respect to it [Barker, 1995].

The nominalizing suffixes *-age* and *-ment* both produce derived forms that refer to something resulting from an event of the verbal base predicate. Some examples are *blockage, seepage, marriage, payment, restatement, shipment*, and *treatment.* The derived forms of *-age* entail that an event occurred and refer to something resulting from it (E-AND-R), *e.g.*, seepage entails that seeping took place and that the seepage resulted from this seeping. Similarly, the derived forms of *-ment* entail that an event took place and refer either to this event, the proposition that the event occurred, or something resulting from the event (E-OR-P-OR-R), *e.g.*, a restatement entails that a restating occurred and refers either to this event, the proposition that the event occurred, or to the actual utterance or written document resulting from the restating event. (This analysis is based on the work of Zucchi [1989].)

The verbal prefix *mis-*, *e.g.*, *miscalculate* and *misquote*, cues the feature that an action is performed in an incorrect manner (INCOR). The suffix *-able* cues a feature that it is possible to perform some action (ABLE), *e.g.*, something is enforceable if it is possible that something can enforce it [Dowty, 1979]. The words derived using *-ness* refer to a property of a kind of state where instances of such a kind have the property of the base (NESS), *e.g.*, in *Kim's fierceness at the meeting yesterday was unusual* the word *fierceness* refers to a kind of state where for instances of this kind, Kim is fierce. The suffix *-ful* marks its base as abstract (ABST): *careful, peaceful, powerful*, etc. In addition, it marks its derived form as the antonym of a form derived by *-less* if it exists (LESSANT). The suffix *-less* marks its derived forms with the analogous feature (FULANT). Some examples are *colorful/less, fearful/less, harmful/less*, and *tasteful/less.*

The precision statistics for the individual lexical semantics features discussed above are presented in Table 4.1 and Table 4.2. Lexical semantic information was collected for 2535 words (bases and derived forms). One way to summarize these tables is to calculate a single precision number for all the features in a table *i.e.*, average the number of correct types for each affix, sum these averages, and then divide this sum by the total number of types. The result: if a random word is derived, its features have a 76 percent chance of being true. If it is a stem of a derived form, its features have an 82 percent chance of being true.

Data for recall was only collected for the verbal prefix *re-* and it was found to be 85 percent. The majority of the missed *re-* verbs were due to the fact that the system only looked at **verbs** starting with *re* and not other categories and many nominalizations such as *reaccommodation* contain a morphological cue token. However, increasing recall by looking at all open class categories would probably decrease precision. Another cause of reduced recall is that some stems were not in the Alvey lexicon or could not be properly extracted by the morphological analyzer. For example, *-Nize* could not be stripped from *hypothesize* because Alvey failed to reconstruct *hypothesis* from *hypothes*. However, for the affixes discussed here, 89 percent of the bases were present in the Alvey lexicon.[4]

4.3.2 Evaluation

Good surface cues are easy to identify, abundant, and correspond to the needed lexical semantic information (Hearst (1992) identifies a similar set of desiderata). With respect to these desiderata, derivational morphology is both a good cue and a bad cue.

Let us start with why it is a bad cue: there may be no derivational cues for the lexical semantics of a particular word. This is not the case for other surface cues, *e.g.*, distributional cues exist for every word in a corpus. In addition, even if a derivational cue does exist, the reliability (on average approximately 76 percent) of the lexical semantic information is too low for many NLP tasks. This unreliability is due in part to the inherent exceptionality of lexical generalization and thus can be improved only partially.

However, derivational morphology is a good cue in the following ways. It provides exactly the type of lexical semantics needed for many NLP tasks: the affixes discussed in the previous section cued nominal semantic class, verbal aspectual class, antonym relationships between words, sentience, etc. In addition, working with the Brown corpus (1.1 million words) and 18 affixes provided such information for over 2500 words. Since corpora with over 40 million words are common and English has over 40 common derivational affixes, one would expect to be able to increase this number by an order of magnitude. In addition, most English words

[4] Note that collecting a relatively complete list of English stems is a task that can be performed in a couple of weeks by one person. Primitive stemmers can be used to get a large initial list of types from a corpus and these types can be corrected by hand quickly. Alternatively, the head words of a MRD can be used as is the case in this study.

Feature	Affix	Types	Precision	Feature	Affix	Types	Precision
TELIC	*re-*	164	91%	RSTATE-EQ-BASE	*-Aify*	17	58%
BASE-RSTATE	*re-*	164	65%	COS	*-Nify*	21	67%
ENTAIL-BASE	*re-*	164	65%	COS	*-ate*	365	48%
REPRESUP	*re-*	164	65%	PART-IN-E	*-ee*	22	91%
COS	*un-*	23	100%	SENTIENT	*-ee*	22	82%
BASE-NEG-RSTATE	*un-*	23	91%	NON-VOL	*-ee*	22	68%
COS	*de-*	35	34%	PART-IN-E	*-er*	471	85%
BASE-NEG-RSTATE	*de-*	35	20%	PART-IN-E	*-ant*	21	81%
COS	*-Aize*	63	78%	E-AND-R	*-age*	43	58%
RSTATE-EQ-BASE	*-Aize*	63	75%	E-OR-P-OR-R	*-ment*	166	88%
COS	*-Nize*	86	56%	INCOR	*mis-*	21	86%
ACTIVITY	*-le*	71	55%	ABLE	*-able*	148	84%
COS	*-en*	36	100%	NESS	*-ness*	307	97%
RSTATE-EQ-BASE	*-en*	36	97%	FULANT	*-less*	22	77%
COS	*-Aify*	17	94%	LESSANT	*-ful*	22	77%

Table 4.1: Derived words

Feature	Affix	Types	Precision	Feature	Affix	Types	Precision
TELIC	*re-*	164	91%	IZE-DEP	*-AenV*	36	72%
COS	*Vun-*	23	91%	IZE-DEP	*-Aify*	15	40%
COS	*Vde-*	33	36%	ABST	*-ful*	76	93%
IZE-DEP	*-Aize*	64	80%				

Table 4.2: Base words

are either derived themselves or serve as bases of at least one derivational affix.[5] Finally, for some NLP tasks, 76 percent reliability may be adequate. If higher reliability is required then only the affixes with high precision might be used. In addition, the precision could be increased by ordering the analyses provided by the morphological analyzer and only considering the top ones. Thus analyses such as *[[re- ache] -ed]* would be ignored in favor of a more plausible parse such as *[reach -ed]*. A list of monomorphemic words would be helpful here. In addition, stochastic methods might also be used to order the parses (see [Heemskerk, 1993]).

The above discussion makes it clear that morphological cueing provides only a partial solution to the problem of acquiring lexical semantic information. However, as mentioned in previously there are many types of surface cues which correspond to a variety of lexical semantic information. A combination of cues should produce better precision where the same information is indicated by multiple cues. For example, the morphological cue *re-* indicates telicity and as mentioned above, the syntactic cue the progressive tense indicates non-stativity [Dorr and Lee, 1992]. Since telicity is a type of non-stativity, the information is mutually supportive. Even some morphological cues are mutually supportive, *e.g.*, the verbs *activate, decorate,* and *calculate* end in *ate* which cues telicity and they also occur prefixed with *re-* which also cues telicity. In addition, using many different types of cues should provide a greater variety of information in general. Thus morphological cueing is best seen as one type of surface cueing that can be used in combination with others to provide lexical semantic information.

It should be mentioned that there is semantic information about any given word that surface cueing is unlikely to be able to provide. For example, from the token *careful*, one can deduce that *care* refers to an abstract concept. However, when one does something with *care*, what does this action look like? How does it differ from a *careless* version? Such information differentiates *care* from other words. (Pinker [1994] refers to this sort of information as the core meaning of the word and argues that perceptual cueing can provide it and surface cueing cannot.) To uncover lexical semantic information, surface cueing utilizes a mapping from a surface cue to a specific semantic feature. The idea is that this cue can be used for many different words. Thus, information specific to one word will not be cued.

4.3.3 Morphological cueing in human language acquisition

As a final note, this section returns to human language acquisition and considers evidence that children make extensive use of derivational morphology cues for

[5]The following experiment supports this claim. Just over 400 open class words were picked randomly from the Brown corpus and the derived forms were marked by hand. Based on this data, a random open class word in the Brown corpus has a 17 percent chance of being derived, a 56 percent chance of being a stem of a derived form, and an 8 percent chance of being both.

learning new words. [Anglin, 1993] studied the English vocabulary knowledge of 6, 8, and 10 year olds. More specifically, he tested the semantic knowledge of these children on 196 words (chosen randomly from a large dictionary). By multiplying the number of words in the dictionary by the proportion of the random sample the children knew, he estimated the size of their vocabularies. Thus, using a hypothetical example, if a test group of children knew half of the words in the sample and the dictionary had say 200,000 words, then he would estimate that the children knew 100,000 words. Using the same technique, he also estimated the make up of their vocabularies with respect to the morphological structure of the words, *e.g.*, again using a hypothetical example, if derived words make up a third of the sample and the children know half of these derived words and they know a third of the sample overall, then he estimates that the children know 33,333 words and of these half are derived. This study supports the view that children between 6 and 10 have an extensive knowledge of derivational morphology and this knowledge accounts for a significant portion of their learning. The tables below (adapted from [Anglin, 1993] summarize the results).[6] First consider Table 4.3 which illustrates the makeup of the main entries of Webster's Third New International Dictionary of The English Language. Note that derived words make up the largest segment of words. Next consider Table 4.4 where the words the children knew are partitioned by their morphological category. Again notice that, for children in Grade 3 and Grade 5, derived words make up the single largest portion of the words they could understand and use.

Word Type	Number	Percentage
Root words	124	29
Inflected words	20	5
Derived words	140	32
Literal compounds	68	16
Idioms	82	19

Table 4.3: The number and percentage of morphologically defined word type in the sample of 434 Main Entries (adapted from [Anglin, 1993] p. 47)

[6][Gordon, 1989; Tyler, 1989; Bowerman, 1982; Clark and Cohen, 1984; Clark, 1982], as referenced in [Anglin, 1993], also report on the knowledge of derivational morphology children have and use.

Word Type	Grade 1	Grade 3	Grade 5
Root words	.31	.24	.20
Inflected words	.27	.22	.15
Derived words	.16	.28	.39
Literal compounds	.25	.23	.21
Idioms	.01	.03	.05

Table 4.4: The mean proportion of children's main entry vocabulary knowledge accounted for by each morphologically defined type at each grade (adapted from [Anglin, 1993] p. 70)

5 Handling New Words in the Input Stream

5.1 Introduction

The previous chapter discussed the acquisition of lexical semantic information for NLP systems. However, even if lexical semantics could be acquired on a large scale and thus NLP systems would all have large lexicon filled with lexical semantic information, it is likely that such systems would still encounter words that are not included in this lexicon. For example, Sproat [1992] describes an experiment where 300,000 different words were collected from the Associated Press newswire in 10 and a half months (44 million words of text). On the day after the last day of collection the following new words were encountered: *armhole, groveled, fuzzier, oxidized, over-emphasized, outclassing, antiprejudice,* and *refinancings.* A quick search through a current news magazine produced the following words: *uncontaminated, titleless, mini-series, underretailed, megamall,* and *unfussy.* None of the words are listed in the 200,000 definition American Heritage dictionary [Berube *et al.*, 1982].

This chapter considers the problem of dealing with such unknown words. More specifically, it presents a system, Kenilex, that uses morphological cues to help process unknown derived words. Upon encountering an unknown derived word, Kenilex uses a modified version of the Alvey morphological system [Ritchie *et al.*, 1992a] to analyze the word and then based on the analysis constructs a new entry for the word. In addition, it creates a new Episodic Logic predicate and assigns it lexical semantic features corresponding to the affix. (Such features were introduced in the previous chapter and are discussed in detail in Chapter 6.). One can view the Kenilex system as a run-time version of the acquisition process described in the previous chapter.

Before Kenilex is discussed further, consider the following study of the makeup of unknown words. The Alvey morphological analyzer (with 60,000 entry lexicon) was run on the first 100,000 tokens of the Brown corpus. Words that did not correspond directly to an entry citation or an inflected form of a citation were considered unknown. Thirteen percent of the tokens were unknown. These tokens were classified by hand into the groups listed in the table. Derivations are tokens that involve a derivational affix—a token "involves" a derivational affix if it contains the affix orthographically and its semantics are built compositionally from the base and the affix. Names are tokens that involve a proper name. Number tokens are those that involve a number (*e.g.* , 1987, 10-year-old, 256). Finally, the "others" category is filled with all the tokens that don't fit into the groups defined above.

derivations	11%
names	58%
numbers	15%
others	16%

Table 5.1: Breakdown of the Unknown Tokens

Although derived forms make up a smaller percentage of unknown tokens than names or numbers, they are often an important predicate of the sentence (*e.g.* a matrix verb, predicate adjective, or central nominalization). And as such, they are central to understanding what the sentence predicates of its participants. In the sentences below, the emphasized tokens were unknown.

(51) a. Note, too, that the *Kennedy* textile plan looks toward *modernization* or shrinkage of the *U.S.* textile industry.

b. The resolution urged the *reconvention* of the *Congolese* parliament and the *reorganization* of the army.

c. Future clouded *Barnett*, as the titular head of the Democratic Party, apparently must make the move to *reestablish* relations with the National Democratic Party or see a movement come from the loyalist ranks to completely bypass him as a party functionary.

d. The *Congolese* were clamoring for their independence, even though most were *unsure* what it meant.

As a final note: since sentences in the Brown corpus are, on average, 18 tokens long, every fourth sentence will, on average, contain an unknown token that is a derivation and it is likely that it will be central to that sentence's meaning.

5.2 Previous work

Most of the work in computational linguistics has focused on the syntactic and phonological aspects of derivational affixation and ignored its semantics. Systems have been developed that can take a complex word as an input, break it into its constituent parts, and use the information stored for these parts to derive the syntactic and phonological features of the word. This task is usually broken into two parts: segmentation and parsing. These subtasks can be interleaved or done one after the other. The segmentation process undoes the phonological rules of a language and produces a normalized form of the constituent parts that can then be looked up in the lexicon. The parsing task amounts to using a rule set to assign an internal structure to a word. Example (52) contains a complex word, its segmentation, and its parse.

(52) a. undecidability

b. un-decide-able-ity

c. [[un- [decide -able]] -ity]

The most influential segmentation system is the KIMMO system originally developed by Kimmo Koskenniemi [Koskenniemi, 1984] and expanded upon by Karttunen [1983]. Most of the systems in use today use a formalism related to that used by KIMMO for segmentation [Black *et al.*, 1986; Domenig, 1988; Bear, 1988]. These systems are quite successful at the segmentation task. Large subsets of morphologically complicated languages such a Finnish can be successfully segmented by such systems.

The systems that attempt word parsing use different subsets of the techniques developed for sentence-level parsing (*e.g.* unification, the concept of head, complex features, chart parsing, etc.). The Alvey morphological parser uses a chart parser and a GPSG-inspired set of percolation constraints to perform the word parsing task [Ritchie *et al.*, 1992b]. Byrd's work [Byrd, 1983] combines parsing and segmentation into a single process based on Aronoff's [1976] conception of word formation rules. [Trost, 1993] describes a system that uses an HPSG-inspired word grammar and parser. For an insightful review of the literature on segmentation and parsing see [Sproat, 1992].

Two works that do address the semantics of derivational affixation are [Schubert, 1982] and [Alshawi, 1992]. They describe systems that build a semantics for a derived word from its parse compositionally, *i.e.*, each affix denotes a logical operator and this operator is applied to the denotation of the base to produce the denotation of the derived form. The Kenilex system also builds the semantics of a derived form from the affix and the base but, in contrast to the systems mentioned above, the denotation of the derived form is a unique semantic atom whose meaning is explicated by axioms. As mentioned in the previous chapter, some advantages of this approach are that no spurious scoping ambiguities are introduced and each derived form maintains its own individual meaning which is only constrained by the axioms, not defined.

5.3 Kenilex

The Kenilex system is an extension of the the Alvey morphological analyzer. Thus the Alvey system will be briefly described first.

The Alvey morphological analyzer has three modules: a dictionary, a segmenter, and a word parser. The segmenter is a KIMMO-like system. It breaks the word into pieces that are consistent with its two-level rules which encode orthographic information. The pieces (or segments) of the word may or may not turn out to correspond to affixes or stems. The parser, as mentioned above, is a chart parser augmented with feature percolation principles. The dictionary consists of entries for each word which contain syntactic features and a semantic atom. In addition, at the time when the dictionary is compiled into a run-time system, a set of feature cooccurence rules are applied to these entries to augment the features of the entries and to produce new entries.

The modules interact as follows. At run-time, the input string is given to the word parser. It calls the segmenter to get the first segment of the word. The segmenter uses the dictionary and its own spelling rules to propose a number of segments where each segment must correspond to an (expanded) entry in the dictionary. The word parser uses the features of these entries to drive a bottom-up chart-parsing process. This process uses the word grammar rules to license possible structures. When it has constructed the possible structures for the segments that have been discovered so far, the word parser asks the segmenter for the next segment. The modules constrain each other as follows: the word parser only uses entries that correspond to segments proposed by the segmenter, the segmenter only uses segments that can correspond to entries in the dictionary, the word parser uses features from the lexicon, and the word parser rejects segmentations for which it cannot find an acceptable parse.

The input to the Kenilex system is a parse tree from the word parser. (Ambiguity is discussed shortly.) If the parse tree does not contain any derivational affixes then it is returned by Kenilex unchanged. If it does contain derivational affixes, Kenilex creates dictionary entries for the derived words. The syntactic features of the entries are taken from the parse tree. A unique semantic atom is produced for the entry and it is assigned the features that correspond to the affix. In addition the feature cooccurence rules are applied to expand the entry.

Consider the sentences in (53) and assume that *reload* is new to the system

(53) Get the empty boxcar along with a boxcar of oranges from Elmira. Use the oranges to make OJ in Bath. **Reload the boxcar** with the bananas and take it back to Elmira. Leave the other boxcar in Bath.

Notice that the use of *reload* instead of *load* is crucial to the disambiguation of *the boxcar* in the emphasized sentence in (53). Thus a relatively complete understanding of *reload* is required. Intuitively the system should be able to make the inferences that the boxcar will be in the state of "is loaded" if the command is carried out and that the boxcar was previously in the state of "is loaded."

When the sentence-level parsing system encounters the string `"reload"` it queries the lexicon system for information concerning the string. On the basis of the Alvey system's parse of the string and the semantics of the prefix *re-*, Kenilex creates an entry for the word and augments Alvey's dictionary. It returns this lexical entry to the sentence-level parser which proceeds with its parse. The symbol `reload` will show up in the logical form produced by the the NLU system for the sentence *reload the boxcar* and it will have the features TELIC, BASE-RSTATE, BASE, and REPRESUP whose corresponding axioms could be used the other modules of the NLU system to make the inferences mentioned above.

There are two complications to this process that should be mentioned. First, in most cases there are multiple parses for any given input string. Kenilex orders these analyses using the following rules and takes the first analysis as correct.[1]

- Analyses that do not involve affixes go before those that do involve affixes.
- Analyses whose outermost affix is inflectional go before those that have a derivational affix as their outer-most affix.
- Analyses that involve fewer derivational affixes are preferred.

The second complication is that for some derived words, the base will not be in the Alvey dictionary. Consider an input string like *rewugable*. Even if people do not have any prior knowledge about what *wug* means, they can infer some things about what *rewug*, *rewugable*, and even *wug* itself mean, when they encounter *rewugable*. Kenilex can also make some inferences based on the internal structure of a possible word like *rewugable*. Of course, such inferences are inherently less reliable than the analogous one that would be made for *reloadable*. They are more useful than simply giving up when presented with such a word. A number of minor modifications to the parser had to be made to the Alvey system to deal with new bases. In addition, the dictionary had to be modified so that it contained new stem entry that are so constructed that they match any surface string of equal length.

When Kenilex, running in normal mode, fails to produce a parse for a word, it switches to new stem mode and starts again. Kenilex processes the parses produced by the word parser as described above, producing new entries when derivational affixes are used. In addition, when a new stem entry is encountered in the parse tree structure, the surface input string covered by the node, its features unified with those provided by the feature percolation conventions, and

[1] Stochastic methods might also be used to order the parses (see [Heemskerk, 1993]).

a new semantic atom are used to construct a raw entry for the new stem and the features corresponding to the affix are assigned to it. It is expanded and added to the dictionary with the other new entries.

The Kenilex system has been integrated with the TRAINS-93 parsing and logical form producing systems. Thus, when a word is not in the TRAINS lexicon, Kenilex system attempts to analyze it. If an analysis is found, Kenilex makes a new entry for the word and returns the TRAINS parser syntactic information about word. It also constructs axioms for a new semantic atom. These axioms are important for both the language understanding tasks and the planning tasks in the TRAINS scenario. For example, (53) illustrates their usefulness for anaphora resolution. In (53), the axioms would also enable the plan recognition system to at least partially incorporate the "reload" action into the plan. Although these uses are possible in principle, the TRAINS-93 did not make use of the axioms produced by Kenilex. This is primarily the case because the system was built to work for pre-selected dialogues and thus handling new words was not a focus.

6 Semantic Analyses of Various Derivational Affixes

This chapter presents the semantic analyses on which the dissertation depends. They have been arranged into the following categories: event structure, event participants, referring to events, other verbal semantics, abstractions, and antonyms. These groupings follow, in part, from the types of information needed for NLP tasks and also from the types of information that English derivational affixes provide. The examples and statistics that are presented for each affix were taken from the study of the Brown corpus using the Alvey lexicon presented in Chapter 4.

6.1 Event structure

Events are ubiquitous in natural language: verbs and sentences describe them, nouns and pronouns refer to them, and noun phrases often refer to their participants. Thus, most NLP systems have found it useful to have some notion of an event in their representation system. Many treat events as first-class members of their ontology. In addition, many NLP tasks require classification of events and even information about their internal structure (*i.e.*, their internal temporal structure). The aspectual classes described in [Vendler, 1967] specify some of this information. For example, statives like *love* in *Mary loves John*, provide a description of an event that is true at all times within the event; the event is homogeneous. Activities do the same down to a particular granularity of description. Telics combine an activity and a state in such a way that the state results from the activity. These pieces of an event may either have duration or be punctual. These distinctions only scratch the surface of the topic of event structure but they will suffice for the semantic analyses presented here.

Before I can formalize some of the observations mentioned above in Episodic Logic, I need to discuss what events are in Episodic Logic. In Episodic Logic, events are elements of the ontology and play a central role. A crucial feature of Episodic Logic is that the truth of a proposition is evaluated with respect to an event (as in Situation Semantics). Thus propositions pick out sets of events and sets of events propositions. There are two ways for a sentence to be true of an event: it can either characterize it or it can merely describe it. A sentence characterizes an event when no proper part of it can be described by the same sentence. `P**e` specifies that `P` is true in `e` and furthermore that it characterizes `e`.[1]

`P*e` specifies that `P` is true in `e` and nothing more. The `*` operator is roughly equivalent to $\models$ in situation semantics. In Section 6.3 below, I will discuss the distinction between `**` and `*` in more detail.

Event structure, in Episodic Logic, is represented by relations between events, most often between subevents and the main event, making use of predicates such as `at-end-of` and `sub-ep`. In Episodic Logic, event structure is represented by relations between events such as `at-end-of` and `during`. In addition, a subepisode relation, `subep-of`, over events forms a join-semilattice of events from the same world. These representational tools can be used to partially axiomatize the aspectual classes as follows.

(54) a. `For all predicates P with features TELIC and DYADIC:`
```
(∀x,y,e [[[x P y]**e] -> (∃e1:  [[e1 at-end-of e] ∧
                                 [e cause e1]]
                                [[y (rstate P)]**e1])])
```
b. `For all predicates P with features ACTIVITY and DYADIC:`
```
(∀x,y,e [[[x P y]**e] -> (∀t:  [[t time] ∧ [t during e] ∧
                                ((appr-grain P) t)]
                               [[x P y]*t])])
```

The axioms above and throughout this chapter work as follows. `P` is a place holder for the semantic predicate corresponding to the word sense which has the named features. There is a one-to-one correspondence between word senses and semantic atoms. Thus an axiom can be seen as a template that produces formulae for all the predicates with the appropriate feature. Many features are assigned to the word sense by the lexical rule for the affix. In addition to axioms particular to the affix, most affixes also assign the feature COMPLEX to their derived forms. This feature is defined in (55) (for dyadic predicates) and links a derived form to its base using the predicate modifier `base`.

(55) `For all predicates D with features COMPLEX and DYADIC and for`

[1]The characterizing relation φ`**e` does not rule out homogeneity. Although `e` cannot have proper parts described by φ — it is in that sense indivisible — it can perfectly well coexist with "smaller" events that occur during it and are described by φ. In fact, if φ is a homogeneous predication, the existence of such smaller events is an entailment: `(∀t,t' [[t' during t] ∧` φ`*t ->` φ`*t'])` where `t,t'` range over time intervals. (In Episodic Logic, time intervals are special cases of events, with "exhaustive" information content.)

```
their corresponding base predicate B with feature DYADIC
(∀x,y [[x (base D) y] -> [x B y]])
```

It is assumed that predicates are typed by their arity and that features such as DYADIC refer to these types. Most of the axioms will only be written for dyadic predicates; however, the axioms for other arities follow straightforwardly.

Returning to our discussion of (54a), the predicate modifier, `rstate`, is applied to the telic predicate, `P`, and the resulting predicate to the second argument of `P`. This sentence-intension is true of an event whose beginning is temporally located at the end of the central event. When `rstate` is applied to a telic predicate, it produces its result state. Axioms specific to `P` provide semantic content. One example would be the axiom for *capture* given below.

(56) `(∀y [y (rstate capture)] -> (∃x [x control y]))`

The result state of telic verbs is crucial for the semantics of a number of affixes and thus what a result state is and how it should be represented needs to be explored here in some detail. Quoting Wechsler: "A result state is a specific state built into the predicate which serves as a criterion for the action to be perfected" [Wechsler, 1989] (p.421).

There are numerous states that hold at the end of an event described by a telic verb. The question that will now be considered is: given a verb, what is its result state? Notice that for many telics there exists a pair of states, one presupposed and one that holds true at the end of the event described by the verb; see (57). The first is a negation of the latter.

(57) open, enter, capture, create, awake

Consider *enter*: one cannot enter a room if he or she is already inside that room and after one has entered the room he or she is inside. The presupposition is stated formally in (58).[2]

```
(58) (∀x,y,e [[[x enter y]**e] ->
             (∃e1:  [e1 at-beginning-of e]
                    [(¬[x contained-in y])**e1])])
```

The result state for such verbs is the state in the pair that holds at the end of the event. For *enter* this state is `contained-in`. Axioms can be used to encode information about the result state of specific verbs. For example, (59) encodes the result state for *enter*.

(59) `(∀y [[y (rstate enter)] -> (∃a [a contained-in y])])`

[2]Throughout this dissertation, presupposition is treated as entailment for expository reasons.

There is another subset of telic verbs that do not have such a pair of states (see (60)). For example, a book does not have to be "not read" to be read.

(60) play (a game), perform (an opera), run (a race), read (a book), write (a memo), type (a letter), design (a deck)

The class exemplified by the first four verbs in (60) is called the 'creation of a performance object' class by Dowty [1979] and the 'path accomplishments' class by Wechsler [1989]. Both Dowty and Wechsler do not group words like *type* and *write* in this class. However, these verbs also lack any sort of negated presupposed state and, for the analysis presented here, this is the crucial distinction between the verbs in (60) and (57).

What are the result states for the verbs (60)? One possibility is that they do not have any result state. This is the position taken by Wechsler. However, intuitively something has changed about the patient: the game is played, the race is run, the letter is typed. I claim that a stative predicate, intuitively similar to their adjectival passive, is the result state of the verbs in (60). Thus, axioms like that in (61) will specify the result state of such verbs.

(61) `(∀y [[y (rstate play)] -> [y (adj-pas play)]])`

The predicate modifier `adj-pas` represents the portion of the meaning of the adjectival passive form that is of concern here. Namely, (62), which keys off `adj-pas` and states that an episode of the action described by the base verb occurred and it involved an agent.[3]

(62)
```
For all predicates P with features VERBAL and DYADIC:
(∀y,e [[[y (adj-pas P)]*e] -> (∃e1:  [[e1 before e] ∧
                                      [e1 at-beginning-of e]]
                                     [(∃x [x P y])**e1])])
```

Thus, the rule for assigning result states for telic verbs is as stated in (63). The axioms in (59) and (61) are products of this rule.

(63) a. If a verb presupposes the negation of a state and this state holds true at the end of events described by the verb, this state is its result state. (change-of-state verbs)

b. Otherwise, a verb's result state is its adjectival passive.

In summary, it is assumed, unlike in [Dowty, 1979] and [Wechsler, 1989], that all telic verbs have a result state and provide a principled way of discovering

[3]Note that this information is not necessarily implicit in the type of the direct object: an opera my never have been performed or a book never read.

what the result state of a verb is. This analysis localizes the difference in types of result states, in the way the result state is chosen. The difference does **not** force a systematic ambiguity in the system. Once the result state is set, the basic analysis of telic verbs proposed here can handle both kinds of verbs with the same mechanism. The axiom for the change-of-state (COS) feature is listed below.

(64)
```
For all predicates P with features COS and DYADIC:
(∀x,y,e [[[x P y]**e] -> [(∃e1:  [[e1 at-end-of e] ∧
                                  [e cause e1]]
                                 [[y (rstate P)]**e1]) ∧
                          (∃e2:  [e2 at-beginning-of e]
                                 [(¬[y (rstate P)])**e2])]])
```

As a final note on result states, the two different kinds of verbs have result states with different properties: the result states corresponding to performance verbs and verbs like *type* are intrinsically dependent on the action denoted by the verb; the state cannot come about if the action has not been performed. Dowty makes this observation for verbs like perform but does not discuss verbs like *type*. Wechsler also makes this observation but considers it evidence that these verbs do not have a result state. In contrast the result states corresponding to change-of-state verbs can often come about for a variety of other reasons. An opera can only become 'performed' if someone performs it. But a tract of land can become controlled by someone if it is bought, inherited, received as a gift, captured, etc. [Smith, 1970] (as referenced in [Levin and Rappaport Hovav, 1994]) calls this the independence of the state.

6.1.1 *re-*

The discussion above has set the stage for an analysis of the semantics of the verbal prefix *re-*. First note that *re-* applies to telic verbs and derives a new verb that is itself telic. (Of the 164 types encountered in the Brown corpus 91 percent had telic bases and were themselves telic.) Thus, this derived verb has a result state. The first semantic generalization about *re-* is that *re-* verbs have the same result states as their base verbs (*e.g.* the result state of *reenter* is the same as that for *enter*). This is stated in (65).

(65)
```
For all predicates P with features BASE-RSTATE
(∀y [[y (rstate P)] -> [y (rstate (base P))]])
```

Of 164 types encountered, 65 percent conform to this axiom.

The second generalization is that if a derived verb holds true of a set of arguments then so does the base verb (*e.g.* if you reenter a room then you also enter a room). This is stated in (66).[4]

```
(66) For all predicates P with features ENTAIL-BASE and DYADIC:
     (∀x,y [[x P y] -> [x (base P) y]])
```

Again, 65 percent of the 164 types conform to this axiom.

Finally, not surprisingly, the *re-* form of a verb presupposes that its result state held previously. Dowty [1979], Marchand [1969], and Wechsler [1989] make this observation. This presupposition is represented in (67).

```
(67) For all predicates P with features REPRESUP and DYADIC:
     (∀x,y,e [[[x P y]**e] ->
             (∃e1:  [e1 before e] [[y (rstate P)]*e1])])
```

(As before, 65 percent of the 164 types conform to this axiom.)

What is surprising is that this is all that needs to be presupposed. For example, *re-* forms do not presuppose the entire event described by the telic verb occurred previously. Consider the examples in (68). (68a) would be true in a situation where the door was hung open when it was created and was closed only once before Mary came along and opened it. (68b) would be true even if the satellite has never entered the atmosphere before. (68c) would be true even if the Druids never captured their homeland before.

(68) a. Mary reopened the door.

b. The satellite reentered the atmosphere.

c. The Druids recaptured their homeland.[5]

As mentioned above for verbs such as *capture* and *open* there are multiple ways to achieve their result states (*e.g.* the state of being possessed by someone can come about as a result of being stolen, received, bought, etc.). In contrast, for verbs such as *play (the game)* there is only one way to achieve the result state: do the action described by the verb. Thus the *re-* form of such verbs indirectly entails that the action denoted by the telic verb occurred previously. This entailment

[4]The reader may have noticed that (66) in conjunction with the axiom for the TELIC feature (54) entail that the result state of the base verb holds at the end of episodes described by the *re-* verb. Thus, it seems that the axiom in (65) is unnecessary. However, *re-* verbs are telic and should, therefore, denote telic predicates. To be consistent, these telic predicates should have a result state. In addition, remember that the general axiom for telic verbs in (54) applies to all telic predicates and involves the **rstate** predicate modifier.

[5]These three examples are from [Dowty, 1979].

falls out of the fact that I am representing the result state of such verbs as their adjectival passive. As noted above, the adjectival passive stative predicate entails that the relevant episode is also described by the base predicate. For a verb such as *retype*, (65) and (67) along with (62) and (69) chain to produce the inference that a typing event occurred previously.

(69) `(∀y [[y (rstate type)] <-> [y (adj-pas type)]])`

In addition, *re-* verbs do not presuppose, by virtue of being *re-* verbs, the negation of their result state. Dowty assumes that *re-* makes this stronger presupposition.[6] He claims that not only does the *re-* form of a verb presuppose that the result state held previously but, in addition, it presupposes that it did **not** hold at some point between when it previously held and the current time (see (70) where *re-* denotes *again'*).

(70) a. $\forall p \Box [again'(p) \leftrightarrow [\hat{}p \wedge H[\neg \hat{}p \wedge H \hat{}p]]]$

b. "That is, *again'*(*p*) is true just in case *p* is now true, there was an earlier time at which *p* was false, and a still earlier time at which *p* was true."([Dowty, 1979], p. 261)

For verbs like *reenter* and *recapture*, this additional presupposition is unnecessary; for verbs like *retype* and *replay the game*, it is wrong. To see this, consider *enter*. As mentioned above, *enter* presupposes that the space being entered does not already contain the object entering (see (58)). Since *reenter* entails *enter* (see (66)), all the presuppositions for *enter* apply for *reenter*. Therefore, they do not need to be stated explicitly as presuppositions for *reenter*.

For verbs such as *retype* and *reread*, there is no presupposition that the negation of the result state held previously (*e.g.* once a book is read it can never be "unread"). And as mentioned above, the base forms of these words do not presuppose the negation of their result state. Thus, the weaker entailment for *re-* verbs proposed here (*i.e.* (67)) performs better than the stronger form (70) proposed by Dowty.

Re- provides us with an opportunity to explore the nature of result states further. More specifically which participants in an event are involved in the result state. With respect to *re-*, Wechsler poses the question as follows: "In what ways must the presupposed earlier situation resemble the denoted one? Specifically, which participants must be common to both?" [Wechsler, 1989] (p.425). Wechsler calls these common participants the nuclear arguments. He gives the following heuristic for deciding which participants of a telic event are nuclear arguments: "a simple test for nuclear argumenthood is that criteria for the completion of

[6]Wechsler lists the axiom in (70) but does not say anything for or against it in his final analysis.

an accomplishment are given in terms of them" [Wechsler, 1989] (p.424). For example, in (71a), John and the English Channel are nuclear arguments since one has to watch both in order to determine whether the swimming event is complete. However, the flippers are not nuclear since they are not crucial to the completion. Consequently, they are not required to be a part of the presupposed event.

(71) a. John reswam the English Channel with flippers.

b. Mary reread Ulysses in one day.

c. Kim reran the last lap of the race for the TV cameras.

d. The train recrossed the border.[7]

When the heuristic is applied to the other sentences in (71), it gives the same result: both the agent and the theme are nuclear arguments and must be common to both the current situation and the presupposed one. The heuristic also works the sentence in (72).

(72) John reawoke at the sound of his cat scratching the door.

Thus far the heuristic works well. However, as Wechsler notes there are a number of problematic sentences. Some of his examples are in (73).

(73) a. John swam the Channel yesterday and found nothing; but I don't believe he looked carefully enough. So I'm going to reswim it today.

b. After Smith's famous crossing in 1930, the Channel remained unconquered for ever 50 years. Finally, in 1986, a young swimmer named Jones reswam the Channel.

c. On September 17th, Dr. Jones reentered the crypt of the pharoah.

d. Mary read the poem aloud and then John reread it.

All of these sentences have a reading where the agent of the current event is not the same as the agent of the presupposed event. Wechsler points out that the themes in these examples are 'affected' in a way that is not present in a reading where the agent must be the same in both events. Thus, one need only pay attention to the theme to tell if the telic event is over. If affected themes can be effectively delineated then his heuristic would remain untarnished. In his own words:

> "Notice that these shifts in scope are not the result of any extra principles of a pragmatic nature or any other kind. The verbs we have looked at (even in unprefixed form) exhibit a certain plasticity with

[7] The examples in (71) and (72) are from Wechsler.

respect to interpretation. But once that interpretation is fixed, the interpretation of *re-* simply follows from the meaning we needed to assign to the morpheme anyway on the basis of the simple cases." [Wechsler, 1989] (p.427)

However, the concept of affectedness is notoriously slippery. In addition, the following examples cannot be handled by the heuristic.

(74) a. Mary swam the last lap of the relay race but the judge did not see it. Since he was fresh, John immediately jumped in and reswam the last lap. Amazingly their team still placed.

b. The US recaptured Kuwait for the Kuwaitis.

Since the theme of *reswim* in (74a) not affected in any way and the observation of both the agent and the theme are required to tell if the telic event is complete, Wechsler's heuristic would predict that both arguments are nuclear. However, only the theme is common to both events. In (74b), the result state of *recapture* involves both the capturer and the capturee and thus both should be common. However, (74b) seems true in a situation that involves the following sequence of events: the Kuwaitis control Kuwait, the Iraqis capture it, the US captures Kuwait, the US gives Kuwait back to the Kuwaitis. Again, the heuristic makes the agent nuclear when it should not.

Although Wechsler's analysis provides many insights and his heuristic seems to be on the right track, this aspect of result states is an area where further research is needed. For the time being, however, the following generalization will be incorporated into the analysis presented here: the theme is **always** a nuclear argument. This is encoded in my analysis by given wide scope to the quantifier for the theme argument variable y in (67).

The agent of the presupposed event is left indeterminate and it is assumed that context will decide the most salient element to fill this argument. This approach is encoded as follows. If the result state involves a dyadic stative predicate, the agent position is treated as being existentially bound (see the variable a in (59)). Thus, for examples like *the platoon captured the hill*, some process will have to equate the platoon with the thing which is in possession of the hill.[8] Notice that such a process is needed for other existentials. Consider the example in (75).[9]

[8]Some verbs seem to focus more on the change of state of the agent. Consider the contrast between *the US (re-)invaded Kuwait* and *the US (re-)captured Kuwait*. One could not call the US invasion a re-invasion because of the previous Iraqi invasion whereas one could call the US capture a recapturing because of the previous control of Kuwait by the Kuwaitis. Thus it seems that for such verbs the result state should predicate the agent.

[9]This example and general approach was pointed out to me by Lenhart Schubert, personal communication.

(75) As the parents of the murder victim looked on, the murderer was executed.

A meaning postulate can represent the entailment that if someone is a murderer, they killed someone. There must be some way to equate that someone who was killed with the murdered child of the parents. Episodic Logic is particularly well-suited to handle such cases since the truth conditions for expressions involving existentially bound variables state that if the variable is bound elsewhere that binding should be used (see [Schubert and Hwang, 1990], page 19, footnote 13).

There is one last feature of the arguments of *re-* verbs that should be noted before leaving this topic. The arguments of *re-* have an extensional quality.[10] Contrast the sentences in (76a,c) with those in (76b,d). Those in (76a,c) allow a reading where the agents and themes are intensional (*e.g.* it could be that it was a different sixth grade class that evaluated a different teacher previously). The sentences in (76b,d) do not allow such a reading.

(76) a. The sixth grade class evaluated their teacher again.

b. The sixth grade class reevaluated their teacher.

c. The Queen inaugurated the Prime minister again.

d. The Queen reinaugurated the Prime minister.

Note that the wide scope of the quantifiers in (67) captures this observation. However, no explanation for this phenomenon is offered here.[11]

6.1.2 *un-* and *de-*

In [Marchand, 1969], three senses are assigned to these negative prefixes:

privative: 'clear - of' defrost, dehusk, debone, unnerve,

ablative: 'remove from -' unsaddle, dethrone

reversative: 'reverse the effect of -ing' unbutton, desegregate, desynchronize

The reversative sense is dominant in the Brown corpus. In addition, it only applies to verbal bases whereas the other two senses tend to apply to nouns. Thus here the concentration will be on the reversative sense of *de-* and *un-*.

[10] Wechsler also makes this observation.

[11] It should be noted that there might be a further problem with this analysis of *re-* related to intensionality. For example, when one *redesigns* a house, there does not seem to be a single house that has been in the same state twice. One way to fix this would be make *re-* apply to the entire verb phrase, *i.e.*, `re` would apply to the monadic predicate `(design house)`.

The first thing to notice is that the reversative only applies to change-of-state verbs. In addition, change-of-state verbs are derived. Remember from our discussion of result states above that change-of-state verbs presuppose the negation of their result state. Of the 23 *un-* types encountered all were change-of-state themselves; 91% of them had change-of-state bases. The counterexamples were *unnerve* and *undo. Unnerve* seems to be an example of the privative sense and thus is derived from the noun *nerve* not the verb. *Do* is a light verb and, on its own, seems to only entail that something is done to the participant correspondent of the direct object. This action may be a change-of-state or it may be simply telic. It would seem plausible that only the actions that are change-of-state can be "undone" and therefore *undo* can be seen as regular in a more complicated way. The sentences that contain *undo* tokens in the Brown corpus are *he undid the bow* and *there was a divine justice in one wrong thus undoing another.* The first clearly conforms to the hypothesis. The second has uses a metaphorical meaning for *undo* and it is unclear what should be made of it.

Moving on to *de-*, of the 35 *de-* types encountered 43 percent of them had change-of-state bases and 34 percent were change-of-state themselves. The numbers for *de-* are less than encouraging. However, most of the counterexamples (*e.g., debut, decry, defend, delay, delight, depart*) seem to have little to do with the prefix, *i.e.*, it seems unlikely that they were derived from the putative stem.

For both *un-* and *de-*, there is more to the relation between the base and the derived form than the fact that they are both change-of-state verbs—the presupposed state and the result state of the base are swapped in the derived form. For example, in order to lace a pair of shoes they must not be already laced up and the result is that they are laced; unlacing has the reversed pre- and post-conditions.

This observation is formalized in the axiom (77).

```
(77) For all predicates P with feature BASE-NEG-RSTATE:
     (∀y [((rstate P) y) -> ¬((rstate (base P)) y)])
```

Note that the idea that the result state is monadic has been retained. This claim was argued for in Section 6.1.1 based on data concerning the prefix *re-* that showed the only the affected participant was necessarily involved in the presupposed state (see examples (71-75)). The presupposed states for *un-* and *de-* derived forms reinforce this claim. For example, to decentralize something, it is only crucial that the affected participant is central prior to the event of decentralizing. This is true of all the *un-/de-* change-of-state examples in the Brown corpus.

6.1.3 *-ize, -ify, -en,* and *-ate*

The central meaning of verbs derived using *-ize, -en, -ify,* or *-ate*[12] is a description of a process of obtaining a result state that is related in some way to the base. The process is usually left largely underspecified. What is important is that the result state is obtained for one of the participants—the function of these verbs in English seems to be to provide very general verbs for referring to acquisition of a result state. Some examples are *equalize, stabilize, awaken, darken, glorify, intensify, alienate*, and *contaminate.* In contrast, some telics that do incorporate specific restrictions on the process are *fax (a letter), quaff (a beer)*, and *paint (a picture).* The verb *fax* specifies how the letter is sent, *quaff* specifies how the beer was drunk, and *paint* specifies how the picture was created. Because the derived verbs are so general, it is hard to imagine the result state coming about in a way that could not be described by the derived verb. For example, it is hard to conceive of a process by which something becomes legal which could not be described with the verb *legalize.* The only way something could be legal and not have been legalized if it started its existence legal, *e.g.*, drinking water was never legalized in New York State—it has always been legal. Not all states are like this. For example, the result state of *capture* is 'control' and one can control something by buying it, inheriting it, or claiming it—just because something is controlled doesn't mean it was captured. In sum, when a base adjective holds of an object, this presupposes that either an event described by the deadjectival verb or the adjective has always been true of the object. This is encoded with the following axiom.

```
(78) For all predicates P with features IZE-DEP and DYADIC:
    (∀y [((base P) y)*e -> (∃e1 [[(∃x [x P y])**e1 ∧ [e1 before e]]
                                   ∨ (∀e2:[e2 before e]
                                         ((base P) y)*e2)]]))
```

80 percent of the 64 adjectival bases of *-ize*, 72 percent of the 36 bases of *-en*, and 93 percent of the 15 adjectival bases of *-ify* conform to this axiom.

The majority of the derived forms are change-of-state verbs: 78 percent of the 63 *-Aize* types, 56 percent of the 86 *-Nize* types, 100 percent of the 36 *-en* types, 94 percent of the 17 *-Aify* types, 67 percent of the 21 *-Nify* types, and 48 percent of the 365 *-ate* types.

In addition, for *-Aize, -Aify*, and *-en*, the result state of a derived verb is often the base itself.

```
(79) For all predicates P with features RSTATE-EQ-BASE and DYADIC:
```

[12]The "suffix" *-ate* should really be called a verbal ending since it is not a suffix in English, *i.e.*, if one strips it off one is seldom left with a word. Nonetheless, it corresponds to lexical semantic information.

```
(∀x ((rstate P) x) -> ((base P) x))
```

For 75 percent of the 63 *-Aize* types, for 58 percent of the 17 *-Aify* types, and for 100 percent of the 36 *-Aize* types this axiom holds. Some of the counterexamples are *materialize, penalize* and *specialize.*

A subclass of the denominal *-ize* forms also conform to this template, namely those that can be glossed as 'to make into a -'. Some examples are *novelize, colonize, deputize, fetishize, idolize*, and *hypothesize.* In Episodic Logic nouns, adjectives, and verbs are all of the same type: predicates. Thus (79) could be used for these nouns also. However, denominal forms are much less homogeneous than deadjectival forms. [Marchand, 1969] gives four principal semantic types in addition to the one just mentioned. The types are, quoting from him 'convert into, put into the form of, give the character or shape of -', *e.g.*, *itemize* and *dramatize*, 'subject to the action, treatment, or process of -', *e.g.*, *propagandize* and *hospitalize*, 'subject to a special (technical) process connected with -', *e.g.*, *winterize* and *weatherize*, 'impregnate, treat, combine with -', *e.g.*, *alcoholize* and *alkalize.* These types cannot be distinguished from each other on purely morpho-syntactic grounds and thus the axiom cannot be automatically assigned to any of the derived forms.

Finally, notice that many of these derived verbs seem to have readings that are contrary to the change-of-state axiom: their patient changes its state but not in a binary fashion. For example, one can blacken a pot but not make it completely black and one can blacken it again. Some other examples are listed in (80).

(80) he brightened the room, she further centralized the power

Such verbs move their patient in a particular direction along a scale and thus the binary terms of negation and statement which we have been using to refer to change of state need to be defined for verbs that involve such scalar adjectives. This problem will not be address here, however.

6.1.4 *-le*

All of the attention in this section has been on telic verbs and this is because most derivational affixes in English seem to be concerned with telic verbs. However, there is a verb ending, in the sense that *-ate* is a verb ending, in English that tends to mark activities: *-le.* Some examples are *chuckle, crumble, dazzle, grumble, guzzle, mingle, ramble, rustle,* and *struggle.* 55 percent of the 71 types ending in *-le* are activities (as defined by the axiom for the ACTIVITY feature given above). If one removes by hand the denominals, *i.e.*, any type with a corresponding homonym noun that is not eventive, 71 percent of the remaining 49 types are activities.

6.2 Event participants

Some verbs specify extremely particular properties for their participants. For example, the German verb *fressen* specifies that its agent cannot be a human. However, there are properties that many verbs specify for their participants. For example, many verbs specify that the doer of the verb must be volitional: must make a conscious choice to perform the verbal action. Some examples are *tell, pay, write,* and *sell.* Such properties are the basis for Dowty's [1991] theory of argument selection. Argument selection is the mapping from participants in an event to syntactic complements of a verb that described the event. Dowty posits that there are two sets of properties: proto-agent properties and proto-patient properties. The syntactic realization of participants is determined by the number of such properties that they have. For example if three participants are to be syntactically realized, then the one with the most proto-agent properties will be realized as the subject; of the remaining two, the one with the most proto-patient properties will be realized as the direct object; and the remaining one will be realized as the oblique object.[13] Dowty's features are listed in (81) and (82) (taken verbatim from [Dowty, 1991] p. 572).

(81) Contributing properties for the Agent Proto-Role:

a. volitional involvement in the event or state

b. sentience (and/or perception)

c. causing an event or change of state in another participant

d. movement (relative to the position of another participant)

(e. exists independently of the event named by the verb)

(82) Contributing properties for the Patient Proto-Role:

a. undergoes change of state

b. incremental theme

c. causally affected by another participant

d. stationary relative to movement of another participant

(e. does not exist independently of the event, or not at all)

If one knows which participants have which properties for a verb then one is a long way towards knowing the verb's argument structure. The following quote taken from [Barker and Dowty, 1993] explores this point further.

[13] Note that Dowty's system does not predict how many or which participants will be realized.

> "This view of roles suggests there many (sic) not be a need for 'argument structure' as a distinct level of linguistic representation from d-structure (or other syntactic structure) or for 'linking rules' to connect one to the other, as these exist in the currently more common views of Grimshaw [1990] and other works cited there. For if semantic arguments of verbs do not fall into disjoint categories like traditional Agent, Theme, Goal, etc. but are fine-grained to an indefinitely specific degree, and if the distribution of a verb's semantic arguments among its subcategorized NPs is really only partially predictable on such semantic grounds at all (*i.e.* is to an extent idiosyncratic to the individual verb), then the most a priori straightforward theoretical architecture is one in which the association of semantic arguments with syntactic ones is trivial and a separate argument structure level is not invoked: semantic arguments can just as well be indexed solely by the grammatical role the verb's lexical entry associates with them, not in terms of Agent, Goal, etc. Generalizations of linking or argument selection principles can be simply static partial generalizations in the lexicon about meaning and subcategorized NPs; this is the view assumed by Dowty [1991] ... To be sure, the proto-role hypothesis is not really incompatible with the more complex view of argument structure; it only suggests it may need more motivation and/or rethinking."

Can these ideas about argument selection be formalized? One can start by creating dyadic predicates that take a participant and an event as arguments, *e.g.*, `[x volitional-involved-in e]` and `[x sentient-of e]`. Then a number of entailments relationships could be based on these predicates (see (83)).

(83) a. `(∀x (∃e [x volitional-involved-in e]) -> [x sentient-of e])`

b. `(∀x (∃e [x sentient-of e]) -> (animate x))`

These ideas have been introduced here because they are useful for specifying the semantics of the nominalizing suffix *-ee* [Barker and Dowty, 1993]. In addition, they set the stage for two other nominalizing affixes: *-er* and *-ant*. Some examples are listed in (84).

(84) a. appointee, assignee, draftee, honoree, interviewee

b. adapter, admirer, advertiser, adviser, analyzer, autoloader

c. accountant, acceptant, applicant, assailant, assistant

Before looking at each affix individually, consider a lexical inference that is supported by all of the derived forms: the existence of an event describable by their base, *e.g.*, if one encounters the noun *assailant* in a dialogue then an event of assailing must have occurred or be occurring. This can be axiomatized as follows.

(85) For all predicates P with features PART-IN-E and DYADIC-BASE:
$(\forall x\ (P\ x) \rightarrow (\exists e\ (\exists y,z\ [y\ (base\ P)\ z]*e) \wedge [x\ part\ e]))$

6.2.1 *-ee*

At first glance it might seem that *-ee* nouns refer to the direct object of the verb as in (86).

(86) employee, nominee, draftee

However a central point of [Barker and Dowty, 1993] is that a syntactic characterization of the referent of *-ee* nouns is "either descriptively inadequate or severely disjunctive" [Barker, 1995]. He goes on to say that "the main difficulty for these syntactic theories is that from a syntactic point of view, the set of possible referents for *-ee* nouns does not seem to be a natural class" [Barker and Dowty, 1993] and gives the examples below.

(87) indirect object: *donee, payee, addressee*

object of governed prep: *experimentee, laughee, gazee*

subject: *attendee, standee, escapee, retiree*

As a final conclusive argument against a syntactic approach, he points out that many *-ee* nouns have no corresponding verbal syntactic complement, *e.g.*, *amputee, jokee, complainee,* and *twistee.*

In lieu of a syntactic analysis, he provides a semantic one. He posits that the referent has the properties listed in (88).

(88) a. episodically linked

b. sentient of the event

c. non-volitional with respect to some aspect of the event

In the previous section the property (88a) marking it with the feature PART-IN-E was discussed and it was axiomatized in (85). Properties (88b) and (88c) are, in part, familiar from our above discussion of verbal argument selection. The additional phrase "with respect to some aspect of the event" of (88c) was added by Barker to handle examples like *retiree, resignee,* and *escapee* which refer to volitional participants. He argues that often the referent of the *-ee* noun will have no volitional control of the duration of the event or of the occurrence of the event: *draftee, honoree, licensee.* However, if the event is punctual then it is often the result state that the participant does not have control of: *retiree, resignee.* Finally, Barker argues that *escapee* "emphasizes the dire consequences resulting from the decision to escape, since escapees are fugitives" [Barker, 1995]. These seem less than convincing since they are apply only to a small group of counterexamples and are heterogeneous with respect to them also. It seems preferable to rephrase (88c) as (89c′) and treat words like *escapee* as exceptions.

(89) c′. non-volitional with respect to the event

(88b) and (89) are axiomatized in (90) and (91) respectively.

```
(90) For all predicates P with features SENTIENT and DYADIC-BASE:
     (∀x (P x) -> (∃e (∃y,z [y (base P) z]*e) ∧ [x sentient-of e]))
```

```
(91) For all predicates P with features NON-VOLITIONAL and
     DYADIC-BASE:
     (∀x (P x) -> (∃e (∃y,z [y (base P) z]*e) ∧
                        ¬[x volitional-involved-in e]))
```

91 percent of the 22 *-ee* derived forms conform to the PART-IN-E axiom; 82 percent conform to the SENTIENT axiom; and 68 percent conform to the NON-VOLITIONAL axiom.

6.2.2 *-er* and *-ant*

Since *-ee* data can be explained using a semantic generalization, it would seem likely that the corresponding deverbal affixes *-er* and *-ant* might also. However, unlike *-ee* nouns, *-er/-ant* nouns refer to a semantically heterogeneous set of participants; see (92).

(92) non-volitional and non-sentient ('instrument'):
- *-er*: adapter, baffler, binder, boiler, feeder, hanger, heater, poker, sander, tranquilizer, transformer, viewer
- *-ant*: deterrent, coolant, depressant, lubricant, stimulant

causer but non-volitional and non-sentient:
- *-er*: reminder, clencher, eye-opener, shocker, thriller, contender(?), disturber(?)
- *-ant*: descendant, communicant 'one who communicates information', participant

habitual:
- *-er*: smoker, announcer, achiever, baker, commuter
- *-ant* : accountant, assistant, attendant, informant, servant

episodic:
- *-er*: borrower, backer, bather
- *-ant*: celebrant, combatant, assailant, discussant

undergoes change-of-state with respect to the base V, (patient-like):
-*er*: bouncer 'this ball is a bouncer',
melter 'that ice cream flavor is a melter',
keeper 'this fish is a keeper', sampler 'I ordered a sampler'

location:
-*er*: kneeler, scribbler

This situation is similar to that of semantic generalizations about participants that correspond to the subject position of a verb. Thus, in contrast to -*ee*, the primary generalization about the participant deverbal affixes -*er* and -*ant* seems to be a syntactic one: -*er*/-*ant* nouns can refer to a participant that can appear as the subject of the base verb. There appear to be very few exceptions to this generalization; however, see (93) for a few.

(93) *kneeler* 'stool to kneel on' [Marchand, 1969], *scribbler* 'writing pad' [Marchand, 1969], *confidant* 'someone you confide in'

Thus, upon encountering an -*er*/-*ant* derived form token, one can only conclude that an event of the base has occurred or is occurring and the referent of the noun is a participant in this event (*i.e.*PART-IN-E). Of the 471 -*er* types, 85 percent conform to this generalization. Of the 21 -*ant* types, 81 percent conform.

6.3 Referring to events

Until now events have been something that an expression describes: *Mary kissed John* has the following logical form (ignoring tense):

(94) `(∃e [Mary kiss John]**e).`

However nominalizations can refer to an event directly: *the performance was fabulous* has the following logical form (ignoring tense):

(95) `(the e:  (performance e) (fabulous e)).`

There are a number suffixes in English that produce such nouns from verbs, *e.g.*, -*ment*, -*ance*, -*ion*, and -*al*.

All of the derived forms presuppose an event of the base verb. For example, the use of the word *performance* presupposes that something was performed. In fact, the individual referred by the word *performance* is often this presupposed event. However, *performance* can also refer to the proposition that something was performed, not to the performance itself. Consider the (96).

(96) Mary's performance of the cello suites upset John

This sentence could either mean (i) that the particular performance upset John or (ii) that the fact that Mary performed the suites upset John. In reading (i), *performance* refers to the event and in (ii) to the proposition. This distinction is discussed at length in [Zucchi, 1989].

For many nominalizations, there is a third possible referent type and that is to an object resulting from the event: *encampment, restatement, approximation, characterization, rental.* Actually, a rental car does not result from the renting event—it would still exist if no event of renting occurred. More accurately, if no renting occurred it would not have the property of being a rental. The object that results from the event is one of the participant of the event.

The axiom encoding the presupposed event and the three possible referents is listed below.

```
(97) For all predicates P with features E-OR-P-OR-R and DYADIC-BASE:
     (∀x (P x) -> (∃e [(∃y,z [y (base P) z]**e) ∧
                       [[x = e] ∨
                        [x = (that [y P z]*e)] ∨
                        [[x part e] ∧
                         [(∀y1,z1,e1 (¬([y1 (base P) z1]**e1))) ->
                          ¬(P x)]]]]))
```

Notice that the three reference possibilities are disjunctively expressed in the last three lines of (97). This axiom also contains the presupposition of an event described by the base verb.

Notice also that in the sentence *the performance was fabulous*, information is being "added" to the event, *i.e.*, not only was it a performance, it was fabulous. There is a problem with this addition of information. If the ** operator is used as is done in the first line of (97) then the event is a minimal one: the characterizing expression, in this case `[y (base P) z]` constitutes the full information content of the event. Thus the "the fabulous performance" will be an event that is coextensive but informationally larger than the "performance". However, intuitively there is only one event. If the * is used instead of ** then the event which is referred to could have too much or too little information in it. For example, exactly what upset John in (98a) is ambiguous: it could be the whole of the performance itself or it could be some particular aspect such as when it happened or the manner in which it happened. However, it probably does not have anything to do with the snow storm that was taking place in New Zealand.

(98) a. Mary performed the 1st cello suite and it upset John

b. The performance took place at 7pm

c. The performance was too slow.

Episodic Logic includes ** as a tool for controlling the information content of an event and thus excluding unrelated pieces of information that could be spuriously added to the information content of an event. It says that its first argument, and the things following from it via axioms, constitute the whole of the information in the event. What nominalizations make clear is that it is too crude of a tool for the intricate work of circumscribing the information content of an event.

One might think that one could introduce an event with slightly more information than the first when another piece of information is given. However, the fact that one can refer to events with pronouns makes this approach less appealing. Consider (99).

(99) Mary performed the 1st cello suite and it upset John. It took place at 7pm and it was too slow.

If a new event is introduced for each piece of information, each pronoun would refer to a different event which is undesirable. Thus, ** will be continued to be used in the axioms presented here and hope that a better definition can be devised.

Nominalizations present of a number of other interesting semantic problems including the relation between the arguments of the nominalization and those of the base verb (*e.g.*, (100a)), the count/mass characteristics of the nominalization and the relation between the aspect of the base verb (*e.g.*, (100a)), and the relation of adjectival modification of the nominalization and adverbial modification of the base verb (*e.g.*, (100a)). However, I will not discuss these problems here. See [Zucchi, 1989] and [Hoeksema, 1985] for formal approaches to some of these problems.

(100) a. *a depreciation in the value of dollar* $\leftrightarrow$ *the value of the dollar depreciated*

b. **Preventions of diseases are good* but *Developments in the Middle East are encouraging*

c. *The skillful performance of the song by Mary* $\leftrightarrow$ *Mary performed the song skillfully*

6.3.1 *-ment*

Nouns derived by *-ment* conformed to the E-OR-P-OR-R axiom 88 percent of the time. There were 166 types. Some non-conforming examples are *commencement, compliment, basement,* and *department.*

A way to distinguish the semantics of *-ment* from that of the other nominalizing suffixes would be of interest and a number of minimal pairs exist (*e.g.*, *excitation/excitement* and *installation/installment*). However, there does not seem to be any coherent semantic distinction.

6.3.2 *-age*

Like *-ment, -ance, -ion,* and *-al, -age* produces deverbal nouns which presuppose an event described by their base. Some examples are *blockage, seepage, marriage, drainage,* and *wreckage.* However, these derived forms can only refer to some object resulting from their presupposed event. They do not refer to the event itself or to a corresponding proposition, *e.g.*, seepage entails that seeping took place and that the seepage resulted from this seeping. The feature is called E-AND-R and the axiom is listed below. It is a reduced version of (97).

(101)
```
For all predicates P with features E-AND-R and DYADIC-BASE:
(∀x (P x) -> (∃e [(∃y,z [y (base P) z]**e) ∧
                 [x part e] ∧
                 [(∀y1 z1 e1 (¬([y1 (base P) z1]**e1))) ->
                  ¬(P x)]]))
```

Of the 43 *-age* derived types, 58 percent conformed to this axiom. The counterexamples include *massage, salvage, sewage, frontage, carriage,* and *average.*

6.4 Other verbal affixes

In addition to aspect, there is a variety of verbal lexical semantic information to which derivational affixes refer. Some of these affixes include *-able, mis-, pre-, post-*, and *out-*. The affixes *mis-* and *-able* are discussed below.

6.4.1 *mis-*

The verbal prefix *mis-*, *e.g.*, *miscalculate* and *misquote*, produces verbs where an action is performed in an incorrect manner. The axiom is presented below. (Of the 21 types, 86 percent conformed to this axiom.)

(102)
```
For all predicates P with features INCOR and DYADIC and
DYADIC-BASE:
(∀x,y [x P y] -> [x ((adv-m incorrect) ((base P) y))])
```

This axiom makes use of the operator `adv-m` which takes a predicate and produces a predicate modifier that modifies a verb's manner. Exactly what `incorrect` entails would depend on the meaning of the base verb. However, one generalization that can be made is that most of the ramifications of a verb would be put into doubt by an incorrect performance.

6.4.2 *-able*

The words derived using *-able* state that it is possible to perform the action denoted by the base verb on the predicated object, *e.g.*, something is enforceable if it is possible that something can enforce it. This analysis is taken from [Dowty, 1979] and his axiomatization translated into Episodic Logic is presented below.

```
(103)For all predicates P with features ABLE and DYADIC-BASE:
    (∀y (P y) -> ◇(∃x,e [x (base P) y]**e))
```

84 percent of the 148 types conform to this axiom.

6.5 Abstractions

English provides a number of affixes that produce nouns for referring to abstract entities. For example, nouns that refer to the state of having some property are produced by *-ancy, -ness,* and *-ity* as in *dormancy, happiness,* and *fluidity.* Nouns that refer to some abstract place or theory are produced by *-ship* and *-ism* as in *chomskyism* and *friendship.* Finally, *-ful* only applies to abstract nouns: *careful, fearful,* and *graceful.* Two of these affixes, *-ness* and *-ful*, are discussed below.

6.5.1 *-ness*

The base predicates for *-ness* are adjectives and thus are stative predicates. The derived forms refer to a kind of event where instances of this kind are events of something having this state (as formalized below).

```
(104)For all predicates P with the feature NESS:
    (∀x (P x) -> (∀e:  [e instance-of x] (∃y ((base P) y)**e)))
```

Thus, "happiness" predicates a kind of event, instances of which are described by someone being happy. 97 percent of the 307 types conform to this axiom.

6.5.2 *-ful*

As mentioned above *-ful* and *-less* only apply to abstract nouns. The word *abstract* has an intuitively clear meaning: having no physical substance. However, it is very difficult to axiomatize this intuition satisfactory. (105) represents an attempt.

```
(105)For all predicates P with the feature ABST:
    (∀x (P x) -> (no-physical-extent x))
```

What exactly would follow from `no-physical-extent` is unclear. However, of the 76 *-ful* bases 93 percent conformed to this intuition.

6.6 Antonyms

The prefixes *dis-, un-, de-*, and *non-* produce antonyms of their bases. The prefixes *un-* and *de-* have already been looked at. Below we will look at a pair of suffixes *-ful/-less* that produce derived forms that are antonyms. Another pair of affixes with this behavior is *anti-/pro-*.

6.6.1 *-ful/-less*

Of the 22 pairs of *-ful/-less* pairs, 77 percent stand in an antonym relationship. Some examples are *art, care, color, fruit, harm, meaning*, and *purpose*.

In order to state the antonym relationship it is necessary to be able to refer to the *-less* and *-ful* forms in the representational logic. Towards this end all the derived forms of *-ful* and *-less* have the features FUL and LESS respectively. The axioms corresponding to these features introduce predicate modifiers that relate the base predicate to the derived predicate (see (106)).

(106)
```
For all predicates P with the feature LESS:
(∀x (P x) <-> (less (base P)) x)

For all predicates P with the feature FUL:
(∀x (P x) <-> (ful (base P)) x)
```

These links back from the base predicate to the derived form predicate can be used to specify a relationship between two forms derived from different affixes but using the same base as in (107).

(107)
```
For all predicates P with the feature LESSANT:
(∀x (P x) -> ¬((less (base P)) x))

For all predicates P with the feature FULANT:
(∀x (P x) -> ¬((ful (base P)) x))
```

Note that both of these axioms could be derived from axiom below.

(108)
```
For all predicates P with the feature LESSFULANT:
(∀x [x (less P)] -> ¬[x (ful P)])
```

7 Conclusion

Some natural language processing (NLP) tasks can be performed with only coarse-grained semantic information about individual words. For example, a system could utilize word frequency and a word co-occurrence matrix in order to perform information retrieval. However, many NLP tasks require at least a partial understanding of every sentence or utterance in the input and thus have a much greater need for lexical semantics. Natural language generation, providing a natural language front end to a database, information extraction, machine translation, and task-oriented dialogue understanding all require lexical semantics. The lexical semantic information commonly utilized includes verbal argument structure and selectional restrictions, corresponding nominal semantic class, verbal aspectual class, synonym and antonym relationships between words, and various verbal semantic features such as causation and manner. The previous chapters have addressed two primary concerns related to such lexical semantic information: how to represent it and how to acquire it.

With respect to representation, the focus has been on supporting lexically-based inferences. The representation language of First Order Logic (FOL) has well-understood semantics and a multitude of inferencing systems have been implemented for it. Thus it is a prime candidate to serve as a lexical semantics representation. However, it has been argued here that FOL, although a good starting point, needs to be extended before it can efficiently and concisely support all the lexically-based inferences needed. Using data primarily from the TRAINS dialogues, the following extensions were argued for: modal operators, predicate modification, restricted quantification, and non-standard quantifiers. These representational tools are present in many systems for sentence-level semantics but have not previously been discussed in the context of lexical semantics.

A number of approaches to acquisition were considered and the approach of surface cueing was found particularly attractive given the current state of technology. Morphological cueing, a type of surface cueing, was introduced. It makes use of fixed correspondences between derivational affixes and lexical semantic information. The semantics of 18 affixes were discussed briefly and data resulting from the application of the method to the Brown corpus presented. To summarize, lexical semantic information for over 2500 words was collected and the information had a precision of 76 percent on average. These words are listed along with their semantic features in the appendix. The axioms defining these features and the affixes that cue them were discussed in detail in Chapter 6.

In addition to this off-line process of sifting through corpora looking for morphological cues, these cues can also be used at run-time to help a natural language understanding system deal with unknown words. On average, every fourth sentence in the Brown corpus contains a derived word not in the 60,000 word Alvey dictionary. The Kenilex system provides lexical semantic information for such derived unknown words.

In sum, this dissertation has considered the representation and acquisition of lexical semantic information and presented a method for acquiring such information based on derivational morphology.

Of course there are a great many extensions of the work presented here that should be considered. Only a small number will be mentioned here.

First, more data is needed. Only 18 derivational affixes have been considered and English has over 40. In addition, there is certainly more semantic information that could be gleaned from the 18 affixes than has been presented here. Furthermore, corpora larger than the Brown corpus should to be used.

Next, as mentioned in Chapter 4, word sense disambiguation needs to be performed for the base of the derived forms. As was also mentioned in Chapter 4, instead of considering all of the analyses provided by the morphological analyzer, only the most plausible ones should be considered. In addition, the mechanism for dealing with unknown bases needs to be improved.

Finally, only one aspect of representing lexical semantic information has been considered here: supporting lexically-based inferences. Other pressing issues include representing the default nature of lexical semantics generalizations and representing the hierarchical structure of this information (see [Copestake, 1992]).

Bibliography

[Allen *et al.*, 1994] J. Allen, L. Schubert, G. Ferguson, P. Heeman, C. H. Hwang, K. Tsuneaki, M. Light, N. Martin, B. Miller, M. Poesio, and D. Traum, "The TRAINS Project: a Case Study in Building a Conversational Planning Agent," *Journal of Experimental and Theoretical Artificial Intelligence*, 7, 1994.

[Alshawi, 1987] H. Alshawi, *Memory and Context for Language Interpretation*, Cambridge University Press, 1987.

[Alshawi, 1989] H. Alshawi, "Analysing the Dictionary Definitions," In Bran Boguraev and T. Briscoe, editors, *Computational Lexicography for Natural Language Processing*, pages 153–169. Longman, 1989.

[Alshawi, 1992] H. Alshawi, *The Core Language Engine*, MIT Press, 1992.

[Andry *et al.*, 1995] F. Andry, J.M. Gawron, J. Dowding, and R. Moore, "A Tool for Collecting Domain Dependent Sortal Constraints from Corpora," In *Proceedings of the 33th Meeting of the Association for Computational Linguistics*, 1995.

[Anglin, 1993] J. M. Anglin, "Vocabulary Development: a Morphological Analysis," *Monographs of the Society for Research in Child Development*, 58(10), 1993.

[Aronoff, 1976] M. Aronoff, *Word Formation in Generative Grammar*, MIT press, 1976.

[Backofen and Nerbonne, 1992] R. Backofen and J. Nerbonne, "Realization and Directed Parsing," In *Proceedings of the German Workshops on Artificial Intelligence*, 1992.

[Barker, 1995] C. Barker, "Episodic -ee in English: Thematic Relations and New Word Formation," In *Proceedings of Semantics and Linguistic Theory SALT 5 Conference*. DMLL Cornell University, 1995.

[Barker and Dowty, 1993] C. Barker and D. Dowty, "Non-verbal Thematic Protoroles," In A. J. Schafer, editor, *Proceedings of the North East Linguistic Society 23*, pages 49–62. GlSA Amherst, 1993.

[Barwise and Cooper, 1981] J. Barwise and R. Cooper, "Generalized Quantifiers and Natural Language," *Linguistics and Philosophy*, 4:159–219, 1981.

[Barwise and Perry, 1983] J. Barwise and J. Perry, *Situations and Attitudes*, MIT Press, 1983.

[Bates *et al.*, 1994] M. Bates, R. Bobrow, R. Ingria, and D. Stallard, "The DELPHI Natural Language Understanding System," In *Proceedings of the 4th Conference on Applied Natural Language Processing (ANLP)*, pages 132–137, 1994.

[Bear, 1988] J Bear, "Morphology with Two-Level Rules and Negative Rule Features," In *COLING*, pages 272–276, 1988.

[Berube *et al.*, 1982] M. Berube, D. Neely, and P. DeVinne, *American Heritage Dictionary Second College Edition*, Houghton Miffin, 1982.

[Berwick, 1983] R. Berwick, "Learning Word Meanings from Examples," In *Proceedings of the 8th International Joint Conference on Artificial Intelligence (IJCAI-83)*, 1983.

[Black *et al.*, 1986] A.W. Black, S.G. Pulman, G.D. Ritchie, and G.J. Russell, "A Dictionary and Morphological Analyser for English," In *COLING*, pages 277–279, 1986.

[Bobrow and Webber, 1980] R. Bobrow and L. Webber, "Knowledge Representation for Syntactic/Semantic Processing," In *Proceedings of the First National Conference of the American Association for Artificial Intelligence (AAAI-80)*, pages 316–323, 1980.

[Bowerman, 1982] M. Bowerman, "Reorganizational Processes in Lexical and Syntactic Development," In E. Wanner and L. R. Gleitman, editors, *Language Acquisition: the State of the Art.* 1982.

[Brachman and Levesque, 1984] R. J. Brachman and H. J. Levesque, "The Tractability of Subsumption in Frame-based Description Languages," In *Proceedings of the fourth National Conference of the American Association for Artificial Intelligence (AAAI-84)*, pages 34–37, 1984.

[Brill, 1994] E. Brill, "Some Advances in Transformation-Based Part of Speech Tagging," In *Proceedings of AAAI-94*, 1994.

[Brown *et al.*, 1991] P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer, "Word-Sense Disambiguation Using Statistical Methods," In *Proceedings of ACL 29*, 1991.

[Brown *et al.*, 1992] P.F. Brown, P.V. deSouza, R.L Mercer, V.J Della Pietra, and J.C. Lai, "Class-Based *n*-gram Models of Natural Language," *Computational Linguistics*, 18(4), 1992.

[Burkert and Forster, 1992] G. Burkert and P. Forster, "Representation of Semantic Knowledge with Term Subsumption Languages," In J. Pustejovsky and S. Bergler, editors, *Lexical Semantics and Knowledge Representation.* Springer-Verlag, 1992.

[Byrd, 1983] R.J. Byrd, "Word Formation in Natural Language Processing Systems," In *COLING*, pages 704–706, 1983.

[Chafe, 1980] W. Chafe, *The Pear Stories: Cognitive, Cultural and Linguistic Aspects of Narrative Production*, Ablex Publishing Corp., Norwood, NJ, 1980.

[Chierchia and McConnell-Ginet, 1990] Gennaro Chierchia and Sally McConnell-Ginet, *Meaning and Grammar: an Introduction to Semantics*, MIT Press, Cambridge, 1990.

[Chinchor *et al.*, 1993] N. Chinchor, L. Hirschman, and D.D. Lewis, "Evaluating Message Understanding Systems: An Analysis of the Third Message Understanding Conference (MUC-3)," *Computational Linguistics*, 19(3):409–449, 1993.

[Chodorow *et al.*, 1985] M. S. Chodorow, R. J. Byrd, and G. E. Heidorn, "Extracting Semantic Hierarchies from a Large On-line Dictionary," In *Proceedings of the 23th Annual Meeting of the Association for Computational Linguistics*, 1985.

[Clark, 1982] E. V. Clark, "The Young Word Maker: a Case Study of Innovation in the Child's Lexicon," In E. Wanner and L. R. Gleitman, editors, *Language Acquisition: the State of the Art.* 1982.

[Clark and Cohen, 1984] E. V. Clark and S. R. Cohen, "Productivity and Memory for Newly Formed Words," *Journal of Child Language*, 2:611–625, 1984.

[Copestake, 1992] A Copestake, *The Representation of Lexical Semantic Information*, PhD thesis, University of Sussex, Brighton, UK, 1992.

[Copestake and Briscoe, 1992] A. Copestake and T. Briscoe, "Lexical Operations in a Unification-Based Framework," In J. Pustejovsky and S. Bergler, editors, *Lexical Semantics and Knowledge Representation.* Springer-Verlag, 1992.

[Copestake *et al.*, 1993] A. Copestake, Antonio Sanfilippo, T. Briscoe, and V. de Paiva, "The ACQUILEX LKB," In T. Briscoe, V. de Paiva, and A. Copestake, editors, *Inheritance, Defaults, and the Lexicon*, pages 148–163. Cambridge University Press, 1993.

[Cumming, 1986] S. Cumming, "The Lexicon in Text Generation," Technical Report ISI/RR-86-168, Information Sciences Institute, University of Southern California, 1986.

[Dahl *et al.*, 1987] D.A. Dahl, M. Palmer, and R.J. Passonneau, "Nominalizations in PUNDIT," In *Proceedings of the 25th Meeting of the Association for Computational Linguistics*, 1987.

[de Bruin and Scha, 1988] J. de Bruin and R. Scha, "The Interpretation of Relational Nouns," In *Proceedings of the 26th Meeting of the Association for Computational Linguistics*, 1988.

[Domenig, 1988] M. Domenig, "Word Manager: A System for the Definition, Access and Maintenance of Lexical Data bases," In *COLING*, pages 154–159, 1988.

[Dorr and Lee, 1992] B. J. Dorr and K. Lee, "Building a Lexicon for Machine Translation: Use of Corpora for Aspectual Classification of Verbs," Technical Report CS-TR-2876, University of Maryland, 1992.

[Dorr, 1993] B.J. Dorr, *Machine Translation: a View from the Lexicon*, MIT Press, 1993.

[Dorr, 1994] B.J. Dorr, "Machine Translation Divergences: a Formal Description and a Proposed Solution," *Computational Linguistics*, 20(4):597–633, 1994.

[Dowding *et al.*, 1993] J. Dowding, J.M. Gawron, D. Appelt, J. Bear, L. Cherny, R. Moore, and D. Moran, "Gemini: a Natural Language System for Spoken-Language Understanding," In *Proceedings of the 31th Meeting of the Association for Computational Linguistics*, 1993.

[Dowty, 1979] David Dowty, *Word Meaning and Montague Grammar*, D. Reidel Publishing, 1979.

[Dowty, 1991] David Dowty, "Thematic Proto-Roles and Argument Selection," *Language*, 67(3), 1991.

[Eberle, 1991] K. Eberle, "On Representing the Temporal Structure of Texts," In O. Herzog and C.-R Rollinger, editors, *Text Understanding in LILOG: Integrating computational linguistics and artificial Intelligence Final report on the IBM Germany LILOG-Project.* Springer-Verlag, 1991.

[Fisher *et al.*, 1991] C. Fisher, H. Gleitman, and L. R. Gleitman, "On the Semantic Content of Subcategorization Frames," *Cognitive Psychology*, 23:331–392, 1991.

[Fodor, 1970] J. A. Fodor, "Three Reasons for Not Deriving 'Kill' from 'Cause to Die'," *Linguistic Inquiry*, 1:429–438, 1970.

[Gleitman, 1990] L. Gleitman, "The Structural Sources of Verb Meanings," *Language Acquisition*, 1:1–55, 1990.

[Gleitman and Landau, 1994] L. Gleitman and B. Landau, *The Acquisition of the Lexicon*, MIT Press, 1994.

[Gomez, 1994] F. Gomez, "Acquiring Knowledge from encyclopedic texts," In *Proceedings of the 4th Conference on Applied Natural Language Processing (ANLP)*, pages 84–90, 1994.

[Gordon, 1989] P. Gordon, "Levels of Affixtion in the Acquisition of English Morphology," *Journal of Memory and Language*, 28:519–530, 1989.

[Granger, 1977] R. Granger, "Foulup: a Program that Figures out Meanings of Words from Context," In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, 1977.

[Grimshaw, 1981] J. Grimshaw, "Form, Function, and the Language Acquisition Device," In C. L. Baker and J. J. McCarthy, editors, *The Logical Problem of Language Acquisition.* MIT Press, 1981.

[Grimshaw, 1990] J. Grimshaw, *Argument structure*, MIT Press, Cambridge, 1990.

[Grishman and Sterling, 1992] R. Grishman and J. Sterling, "Acquisition of Selectional Patterns," In *Proceedings of COLING-92*, 1992.

[Grishman and Sterling, 1993] R. Grishman and J. Sterling, "Smoothing of Automatically Generated Selectional Constraints," In *Proceedings of the Human Language Technology Workshop ARPA*, 1993.

[Gross, 1993] D. Gross, "The TRAINS 91 Dialogues," Technical Report 92-1, Department of Computer Science, University of Rochester, 1993.

[Gross *et al.*, 1993] D. Gross, Allen J.F., and D.R. Traum, "The TRAINS 91 Dialogues," Technical Report 92-1, Department of Computer Science, University of Rochester, 1993.

[Haenelt and Wanner, 1992] Karin Haenelt and Leo Wanner, *International Workshop on The Meaning-Text Theory*, Gesellschaft für Mathematik und Datenverarbeitung MBH, 1992.

[Hastings, 1994] P. Hastings, *Automatic Acquistion of Word Meaning from Context*, PhD thesis, University of Michigan, 1994.

[Hayes, 1979] P.J. Hayes, "The Logic of Frames," In D. Metzing, editor, *Frame Conceptions and Text Understanding.* de Gruyter, 1979.

[Hearst, 1992] M. Hearst, "Automatic Acquisition of Hyponyms from Large Text Corpora," In *COLING 14 (Nantes)*, 1992.

[Hearst and Schuetze, 1994] M. A. Hearst and H. Schuetze, "Customizing a Lexicon to Better Suit a Computational Task," ms., 1994.

[Heemskerk, 1993] J. S. Heemskerk, "A Probabilistic Context-free Grammar for Disambiguation in Morphological Parsing," In *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, pages 183–192, 1993.

[Herzog and Rollinger, 1991] O. Herzog and C. R. Rollinger, *Text Understanding in LILOG: Integrating computational linguistics and artificial Intelligence Final report on the IBM Germany LILOG-Project*, Springer-Verlag, 1991.

[Hindle, 1990] D. Hindle, "Noun Classification from Predicate-Argument Structures," In *Proceedings of the 28th Annual Meeting of the Assocation of Computational Linguistics*, 1990.

[Hinrichs, 1981] E. W. Hinrichs, "Temporale Anaphora in Englischen," StaatsExamen Thesis, Universität Tubingen, 1981, StaatsExam.

[Hoeksema, 1985] J. Hoeksema, *Categorial Morphology*, Garland Publishing, Inc., 1985.

[Hutchins and Somers, 1992] W.J. Hutchins and H.L. Somers, *An Introduction to Machine Translation*, Academic Press, 1992.

[Hwang and Schubert, 1993a] C.H. Hwang and L. Schubert, "Episodic Logic: a Comprehensive Natural Representation for Language Understanding," *Mind and Machine*, 3(4):381–419, 1993.

[Hwang and Schubert, 1993b] C.H. Hwang and L. Schubert, "Episodic Logic: a Situational Logic for Natural Language Processing," In P. Aczel, D. Israel, Y. Katagiri, and S. Peters, editors, *Situation Theory and its Applications*, volume 3, pages 303–338. CSLI, Stanford University, 1993.

[Iordanskaja *et al.*, 1991] L. Iordanskaja, R. Kittredge, and A. Polguére, "Lexical Selection and Paraphrase in a Meaning-Text Generation Model," In C.L. Paris, W.R. Swartout, and W.C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Kluwer Academic Publisher, 1991.

[Jackendoff, 1990] Ray Jackendoff, *Semantic Structures*, MIT Press, Cambridge, 1990.

[Jacobs and Zernik, 1988] P. Jacobs and U. Zernik, "Acquiring Lexical Knowledge from Text: a Case Study," In *Proceedings of the Eighth National Conference of the American Association for Artificial Intelligence (AAAI-88)*, pages 739–744, 1988.

[Johnson, 1987] M. Johnson, *Attribute-Value Logic and the Theory of Grammar*, Center For The Study of Language And Information, 1987.

[Kamp, 1979] H. Kamp, "Events, Instant and Temporal Reference," In R. Bauerle, U. Egli, and A. von Stechow, editors, *Semantics from Different Points of View*, pages 376–417. Springer-Verlag, 1979.

[Karttunen, 1983] L. Karttunen, "KIMMO: A General Morphological Processor," *Texas Linguistic Forum*, 22:165–278, 1983.

[Kasper and Rounds, 1986] R. T. Kasper and W. C. Rounds, "A Logical Semantics for Feature Structures," In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, 1986.

[Koskenniemi, 1984] K. Koskenniemi, "A General Computational Model For Word-Form Recognition and Production," In *COLING*, pages 178–181, 1984.

[Kucera and Francis, 1967] H. Kucera and W. Francis, *Computational Analysis of Present-day American English*, Brown University Press: Providence, R.I., 1967.

[Kuhlen, 1983] R. Kuhlen, "Topic: a System for Automatic Text Condensation," *ACM SIGART Newsletter*, 83:20–21, 1983.

[Laubsch and Nerbonne, 1991] J. Laubsch and J. Nerbonne, "An overview of NLL," Technical report, DFKI, Saarbrücken, Germany, 1991.

[Levin, 1993] B. Levin, *English Verb Classes and Alternations: a Preliminary Investigation*, The University of Chicago Press, 1993.

[Levin and Rappaport Hovav, 1994] B. Levin and M. Rappaport Hovav, "A Preliminary Analysis of Causative Verbs in English," In L. Gleitman and B. Landau, editors, *The Acquisition of the Lexicon.* MIT Press, 1994.

[Light, 1992] M. Light, "Rehashing *Re-*," In *Proceedings of the Eastern States Conference on Linguistics.* Cornell University Linguistics Department Working Papers, 1992.

[Marchand, 1969] Hans Marchand, *The Categories and Types of Present-Day English Word-Formation. 2nd Ed.*, Beck, Muenchen, 1969.

[Marcus *et al.*, 1993] M. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a Large Annotated Corpus of English: The Penn Treebank," *Computational Linguistics*, 19(2):313–330, 1993.

[Markowitz *et al.*, 1986] J. Markowitz, T. Ahlswede, and M. Evens, "Semantically Significant Patterns in Dictionary Definitions," In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, 1986.

[Matthiessen, 1991] C. Matthiessen, "Lexico(grammatical) Choice in Text Generation," In C.L. Paris, W.R. Swartout, and W.C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics.* Kluwer Academic Publisher, 1991.

[McDonald, 1991] D.D. McDonald, "On the Place of Words in the Generation Process," In C.L. Paris, W.R. Swartout, and W.C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics.* Kluwer Academic Publisher, 1991.

[Mel'čuk and Polguère, 1987] I. Mel'čuk and A. Polguère, "A Formal Lexicon in the Meaning-Text Theory (or How to Do Lexica with Words)," *Computational Linguistics*, 13(3-4):261–275, 1987.

[Mel'čuk and Polguère, 1995] I. Mel'čuk and A. Polguère, "Lexical Functions: a Tool for the Description of Lexical Relations in a Lexicon," unpublished manuscript, 1995.

[Mercer, 1992] R. Mercer, "Presuppositions and Default Reasoning: A Study in Lexical Pragmatics," In J. Pustejovsky and S. Bergler, editors, *Lexical Semantics and Knowledge Representation.* Springer-Verlag, 1992.

[Merialdo, 1994] B. Merialdo, "Tagging English Text with a Probabilistic Model," *Computational Linguistics*, 20(2):155–172, 1994.

[Montague, 1974] R. Montague, "The Proper Treatment of Quantification in Ordinary English," In R. H. Thomason, editor, *Formal Philosophy. Selected Papers of Richard Montague.* Yale University Press, 1974.

[Moore, 1981] R.C. Moore, "Problems in Logical Form," In *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics*, 1981.

[Nagao, 1987] M. Nagao, "Role of Structural Transformation in a Machine Translation System," In S. Nirenburg, editor, *Machine Translation: Theoretical and Methodological Issues.* Cambridge University Press, 1987.

[Naigles *et al.*, 1989] L. Naigles, H Gleitman, and L.R. Gleitman, "Children Acquire Word Meaning Components from Syntactic Evidence," In E. Dromi, editor, *Linguistic and Conceptual Development.* 1989.

[Nebel and Smolka, 1991] B. Nebel and G. Smolka, "Attributive Description Formalisms ... and the Rest of the World," In O. Herzog and C. Rollinger, editors, *Text Understanding in LILOG.* Springer-Verlag, 1991.

[Nerbonne, 1992] J. Nerbonne, "NLL models," Technical report, DFKI, Saarbrücken, Germany, 1992.

[Nerbonne, 1993] John Nerbonne, "A Feature-based Syntax/Semantics Interface," *Annals of Mathematics and Artificial Intelligence*, 8:107–132, 1993.

[Nirenburg and Raskin, 1987] S. Nirenburg and V. Raskin, "The Subworld Concept Lexicon and the Lexicon Management System," *Computational Linguistics*, 13(3-4):276–289, 1987.

[Palmer, 1991] M. Palmer, "General Lexical Representation for an Effect Predicate," In J. Pustejovsky and S. Bergler, editors, *Proceedings of the Lexical Semantics and Knowledge Representation Workshop Sponsored by the SIGLex*, 1991.

[Palmer *et al.*, 1986] M. Palmer, D.A. Dahl, R.J. Schiffman, L. Hirschman, M. Linebarger, and J. Dowding, "Recovering Implicit Information," In *Proceedings of the Strategic Computing Natural Language Workshop sponsored by DARPA*, 1986.

[Palmer, 1990] M.S. Palmer, *Semantic Processing for Finite Domains*, Cambridge University Press, 1990.

[Partee, 1984] B. H. Partee, "Nominal and Temporal Anaphora," *Linguistics and Philosophy*, 7:243–286, 1984.

[Passonneau, 1988] R.J. Passonneau, "A Computational Model of the Semantics of Tense and Aspect," *Computational Linguistics*, 14(2):44–60, 1988.

[Pentheroudakis and Vanderwende, 1993] J. Pentheroudakis and L. Vanderwende, "Automatically Identifying Morphologial Relations in Machine-Readable Dictionaries," In *Proceedings of the Ninth Annual Conference of the UW Centre for the New OED and text research*, 1993.

[Pereira *et al.*, 1993] F Pereira, N. Tishby, and L. Lee, "Distributional Clustering of English Words," In *Proceedings of the 31st Annual Meeting of the Assocation of Computational Linguistics*, 1993.

[Pinker, 1989] S. Pinker, *Learnability and Cognition: The Acquisition of Argument Structure*, MIT Press, Cambridge, 1989.

[Pinker, 1994] S. Pinker, "How Could a Child Use Verb Syntax to Learn Verb Semantics?," In L. Gleitman and B. Landau, editors, *The Acquisition of the Lexicon.* MIT Press, 1994.

[Pletat, 1991] U. Pletat, "The Knowledge Representation Language L_{LILOG}," In O. Herzog and C. Rollinger, editors, *Text Understanding in LILOG.* Springer-Verlag, 1991.

[Pugeault, 1994] F. Pugeault, "Handling Thematic Role Distributions in Argument Structures," In H. Trost, editor, *Proceedings of KONVENS-94*, 1994.

[Pustejovsky, 1988] J. Pustejovsky, "Constraints on the Acquisition of Semantic Knowledge," *International Journal of Intelligent Systems*, 3:247–268, 1988.

[Pustejovsky and Anick, 1988] J. Pustejovsky and P. Anick, "On the Semantics of Nominals," In *COLING*, 1988.

[Pustejovsky *et al.*] J. Pustejovsky, S. Bergler, and P. Anick, "Lexical Semantic Techniques for Corpus Analysis," unpublished manuscript.

[Pustejovsky and Boguraev, 1993] J. Pustejovsky and B. Boguraev, "Lexical Knowledge Representation and Natural Language Processing," *Artificial intelligence*, 63:193–223, 1993.

[Pustejovsky and Nirenburg, 1987] J. Pustejovsky and S. Nirenburg, "Lexical Selection in the Process of Language Generation," In *Proceedings of the 25th Meeting of the Association for Computational Linguistics*, 1987.

[Pustejovsky, 1991] James Pustejovsky, "The Generative Lexicon," *Computational Linguistics*, 17:409–441, 1991.

[Reiter, 1991] E. Reiter, "A New Model of Lexical Choice for Nouns," *Computational Intelligence*, 7:240–251, 1991.

[Rickheit, 1991] M. Rickheit, "Sortal Information in lexical concepts," In O. Herzog and C.-R Rollinger, editors, *Text Understanding in LILOG: Integrating computational linguistics and artificial Intelligence Final report on the IBM Germany LILOG-Project.* Springer-Verlag, 1991.

[Ritchie *et al.*, 1992a] G.D. Ritchie, G.J. Russell, A.W. Black, and S.G. Pulman, *Computational Morphology: Practical Mechanisms for the English Lexicon*, MIT Press, 1992.

[Ritchie *et al.*, 1992b] Graeme D. Ritchie, Graham J. Russell, Alan W. Black, and Steve G. Pulman, *Computational Morphology: practical mechanisms for the English lexicon*, MIT press, 1992.

[Rooth, 1995] M. Rooth, "Latent Class Models for Grammatical Relations," unpublished manuscript, 1995.

[Rupp *et al.*, 1992] C.J. Rupp, R. Johnson, and M. Rosner, "Situation Schemata and Linguistic Representation," In M. Rosner and R. Johnson, editors, *Computational linguistics and formal semantics.* Cambridge University Press, 1992.

[Schaeffer *et al.*, 1993] S. Schaeffer, C.H. Hwang, J. de Haan, and L.K. Schubert, "EPILOG: The Computational System for Episodic Logic (User's Guide, Quick Reference Manual, and Programmer's Guide)," Technical report, Dept. of Computing Science, Univ. of Alberta, Edmonton, Canada T6G 2H1, 1993.

[Schank and Abelson, 1977] R. Schank and R. Abelson, *Scripts, plans, goals, and understanding*, Lawrence Erlbaum Associates, 1977.

[Schmolze and Israel, 1983] J. G. Schmolze and D. J. Israel, "KL-ONE: Semantics and classification," In *Research in Knowledge Representation and Natural Language Understanding*, pages 27–39. Bolt, Beranek, and Newman Inc., 1983, BBN Technical Report No. 5421.

[Schubert, 1982] L. Schubert, "An Approach to the Syntax and Semantics of Affixes in Conventionalized Phrase Structure Grammar," In *Proceedings of the Fourth Bienn. Conference of the CSCSI/SCEIO*, pages 189–195, 1982.

[Schubert, 1990] L. K. Schubert, "Semantic Nets are in the Eye of the Beholder," Technical Report 346, University of Rochester, 1990.

[Schubert and Hwang, 1990] L. K. Schubert and C. H. Hwang, "An Episodic Knowledge Representation for Narrative Texts," Technical Report 345, Department of Computer Science, University of Rochester, 1990.

[Schuetze, 1992a] H. Schuetze, "Dimensions of Meaning," In *Proceedings of Supercomputing 92*, 1992.

[Schuetze, 1992b] H. Schuetze, "Word Sense Disambiguation With Sublexical Representations," In *Statistically-Based NLP Techniques (AAAI Workshop, July 12-16, 1992, San Jose, CA.)*, pages 109–113, 1992.

[Schuetze, 1993] H. Schuetze, "Word Space," In S.J. Hanson, J.D. Cowan, and C.L. Giles, editors, *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann, San Mateo CA, 1993.

[Selfridge, 1980] M. G. R. Selfridge, *A Process Model of Language Acquisition*, PhD thesis, Yale, 1980.

[Siskind, 1990] J. M. Siskind, "Acquiring Core Meanings of Words, Represented as Jackendoff-style Conceptual Structures, from Correlated Streams of Linguistic and Non-linguistic Input," In *Proceedings of the 28th Meeting of the Association for Computational Linguistics*, 1990.

[Smith, 1970] C. S. Smith, "Jespersen's 'move' and 'change' class and causative verbs in English," In M. A. Jazayery, E. C. Palome, and W. Winter, editors, *Linguistic and literary studies in honor of Archibald A. Hill, Vol. 2: Descriptive linguistics.* The Hague: Mouton, 1970.

[Smolka, 1991] G. Smolka, "A Feature Logic with Subsorts," In J. Wedekind and C. Rohrer, editors, *Unification in Grammar.* MIT Press, 1991.

[Sondheimer *et al.*, 1989] N.K. Sondheimer, S. Cumming, and R. Albano, "How to Realize a Concept: Lexical Selection and the Conceptual Network in Text

Generation," Technical Report ISI/RR-89-248, Information Sciences Institute, University of Southern California, 1989.

[Sproat, 1992] R. Sproat, *Morphology and Computation*, MIT Press, 1992.

[Tenny, 1992] C. Tenny, "The Aspectual Interface Hypothesis," In I. A. Sag and A. Szabolcsi, editors, *Lexical Matters*. CSLI, 1992.

[Trost, 1993] H. Trost, "Coping with Derivation in a Morphological Component," In *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, pages 368–376, 1993.

[Tyler, 1989] A. Tyler, "The acquisition of English derivational morphology," *Journal of Memory and Language*, 28:649–667, 1989.

[Velardi *et al.*, 1991] P. Velardi, M. T. Pazienza, and M. Fasolo, "How to Encode Semantic Knowledge: a Method for Meaning Representation and Computer-aided Acquisition," *Computational Linguistics*, 17(2), 1991.

[Vendler, 1967] Zeno Vendler, *Linguistics in Philosophy*, Cornell University Press, Ithaca, New York, 1967.

[Webber, 1988] B.L. Webber, "Tense as Discourse Anaphora," *Computational Linguistics*, 14(2):61–73, 1988.

[Webster and Marcus, 1989] M Webster and M. Marcus, "Automatic Acquisition of the Lexical Semantics of Verbs from Sentence Frames," In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, 1989.

[Wechsler, 1989] S. Wechsler, "Accomplishments and the Prefix *re-*," In *Proceedings of the North East Linguistic Society 19*, 1989.

[Wu and Palmer, 1994] Z. Wu and M. Palmer, "Verb Semantics and Lexical Selection," In *Proceedings of the 32th Meeting of the Association for Computational Linguistics*, 1994.

[Yarowsky, 1993] D. Yarowsky, "One Sense per Collocation," In *Proceedings of the ARPA Workshop on Human Language Technology*. Morgan Kaufmann, 1993.

[Zucchi, 1989] Alessandro Zucchi, *The Language of Propositions and Events: Issues in the Syntax and the Semantics of Nominalization*, PhD thesis, University of Massachusetts, Amherst, MA, 1989.

A A Lexicon Produced by Applying Morphological Cueing to the Brown Corpus

This appendix contains the results of applying the process of morphological cueing described in Chapter 4 to the Brown corpus: a lists of words marked with semantic features. In addition, the appendix contains information relevant to the process: lists of axioms, tables of results, etc. This information will be presented first followed by the lists of words.

A.1 Axioms and tables

Below is a list of axioms collected from the previous appendix.

COMPLEX

```
For all predicates D with features COMPLEX and DYADIC and
and for their corresponding base predicate B with
feature DYADIC
(∀x,y [[x (base D) y] -> [x B y]])
```

TELIC

```
For all predicates P with features TELIC and DYADIC:
(∀x,y,e [[[x P y]**e] -> (∃e1:  [[e1 at-end-of e] ∧
                                  [e cause e1]]
                                 [[y (rstate P)]**e1])])
```

ACTIVITY

```
For all predicates P with features ACTIVITY and DYADIC:
(∀x,y,e [[[x P y]**e] -> (∀t:  [[t time] ∧ [t during e] ∧
                                ((appr-grain P) t)]
                               [[x P y]*t])])
```

BASE-RSTATE

```
For all predicates P with features BASE-RSTATE
(∀y [[y (rstate P)] -> [y (rstate (base P))]])
```

ENTAIL-BASE

```
For all predicates P with features ENTAIL-BASE and DYADIC:
(∀x,y [[x P y] -> [x (base P) y]])
```

REPRESUP

```
For all predicates P with features REPRESUP and DYADIC:
(∀x,y,e [[[x P y]**e] ->
        (∃e1:  [e1 before e] [[y (rstate P)]*e1])])
```

COS

```
For all predicates P with features COS and DYADIC:
(∀x,y,e [[[x P y]**e] -> [(∃e1:  [[e1 at-end-of e] ∧
                                   [e cause e1]]
                                  [[y (rstate P)]**e1]) ∧
                          (∃e2:  [e2 at-beginning-of e]
                                  [(¬[y (rstate P)])**e2])]])
```

BASE-NEG-RSTATE

```
For all predicates P with feature BASE-NEG-RSTATE:
(∀y [((rstate P) y) -> ¬((rstate (base P)) y)])
```

IZE-DEP

```
For all predicates P with features IZE-DEP and DYADIC:
(∀y [((base P) y)*e -> (∃e1 [[(∃x [x P y])**e1 ∧ [e1 before e]]
                              ∨ (∀e2:[e2 before e]
                                     ((base P) y)*e2)]]))
```

RSTATE-EQ-BASE

```
For all predicates P with features RESTATE-EQ-BASE and DYADIC:
(∀x ((rstate P) x) -> ((base P) x))
```

PART-IN-E

```
For all predicates P with features PART-IN-E and DYADIC-BASE:
(∀x (P x) -> (∃e (∃y,z [y (base P) z]*e) ∧ [x part e]))
```

SENTIENT

```
For all predicates P with features SENTIENT and DYADIC-BASE:
(∀x (P x) -> (∃e (∃y,z [y (base P) z]*e) ∧ [x sentient-of e]))
```

NON-VOLITIONAL

```
For all predicates P with features NON-VOLITIONAL and
DYADIC-BASE:
```

```
(∀x (P x) -> (∃e (∃y,z [y (base P) z]*e) ∧
                  ¬[x volitional-involved-in e]))
```

E-AND-R

For all predicates P with features E-AND-R and DYADIC-BASE:

```
(∀x (P x) -> (∃e [(∃y,z [y (base P) z]**e) ∧
                  [x part e] ∧
                  [(∀y1 z1,e1 (¬([y1 (base P) z1]**e1))) ->
                   ¬(P x)]]))
```

E-OR-P-OR-R

For all predicates P with features E-OR-P-OR-R and DYADIC-BASE:

```
(∀x (P x) -> (∃e [(∃y,z [y (base P) z]**e) ∧
                  [[x = e] ∨
                   [x = (that [y P z]*e)] ∨
                   [[x part e] ∧
                    [(∀y1,z1,e1 (¬([y1 (base P) z1]**e1))) ->
                     ¬(P x)]]]]))
```

INCOR

For all predicates P with features INCOR and DYADIC and DYADIC-BASE:

```
(∀x,y [x P y] -> [x ((adv-m incorrect) ((base P) y))])
```

ABLE

For all predicates P with features ABLE and DYADIC-BASE:

```
(∀y (P y) -> ◇(∃x,e [x (base P) y]**e))
```

NESS

For all predicates P with the feature NESS:

```
(∀x (P x) -> (∀e: [e instance-of x] (∃y ((base P) y)**e)))
```

ABST

For all predicates P with the feature ABST:

```
(∀x (P x) -> (no-physical-extent x))
```

LESSANT

For all predicates P with the feature LESSANT:

```
(∀x (P x) -> ¬((less (base P)) x))
```

FULANT

For all predicates P with the feature FULANT:

```
(∀x (P x) -> ¬((ful (base P)) x))
```

LESS

For all predicates P with the feature LESS:

```
(∀x (P x) <-> (less (base P)) x)
```

FUL

```
For all predicates P with the feature FUL:
(∀x (P x) <-> (ful (base P)) x)
```

Next, a number of tables are presented that are relevant to the extraction of prospective tokens for the morphological cueing process. The first table lists the tags that were used in the version of the Brown corpus used here [Marcus *et al.*, 1993]. The second table lists the regular expressions used to extract prospective tokens from the corpus. The token regular expressions were matched against the words of the corpus and the tag regular expressions were matched against the tags. The next table lists, for each affix, the number of types found and the number of types that contain stems that were not in the Alvey dictionary. The final two tables were presented in Chapter 4 and contain the precision statistics for the semantic features defined by the axioms above. These statistics are derived from the words listed below ignoring types where the base was not in the Alvey dictionary (see the discussion of collection features at the beginning of the next section).

Brown Corpus Tags

1.	CC	Coordinating conjunction
2.	CD	Cardinal number
3.	DT	Determiner
4.	EX	Existential *there*
5.	FW	Foreign word
6.	IN	Preposition or subordinating conjunction
7.	JJ	Adjective
8.	JJR	Adjective, comparative
9.	JJS	Adjective, superlative
10.	LS	List item marker
11.	MD	Modal
12.	NN	Noun, singular or mass
13.	NNS	Noun, plural
14.	NP	Proper noun, singular
15.	NPS	Proper noun, plural
16.	PDT	Predeterminer
17.	POS	Possessive ending
18.	PP	Personal pronoun
19.	PP$	Possessive pronoun
20.	RB	Adverb
21.	RBR	Adverb, comparative
22.	RBS	Adverb, superlative
23.	RP	Particle
24.	SYM	Symbol
25.	TO	*to*
26.	UH	Interjection
27.	VB	Verb, base form
28.	VBD	Verb, past tense
29.	VBG	Verb, gerund or present participle
30.	VBN	Verb, past participle
31.	VBP	Verb, non-3rd person singular present
32.	VBZ	Verb, 3rd person singular present
33.	WDT	Wh-determiner
34.	WP	Wh-pronoun
35.	WP$	Possessive wh-pronoun
36.	WRB	Wh-adverb

Regular expressions used to extract prospective tokens

Affix	Token Regular Exp.	Tag Regular Exp.
Vun-	^UN.+	^VB[^N]
Vde-	^DE.+	^VB[^N]
re-	^RE.+	^V.*
-Aize	.*IZ(E|ING|ES|ED)$	^V.*
-Nize	.*IZ(E|ING|ES|ED)$	^V.*
-Aify	.*IF(Y|YING|IES|IED)$	^V.*
-Nify	.*IF(Y|YING|IES|IED)$	^V.*
-ate	.*AT(E|ING|ES|ED)$	^V.*
-AenV	.*EN(ING|S|ED)$	^V.*
-leV	^[^]*[^AEIOU]LE$	^V.*
-ee	.+EE(S|)$	^NN.*
-VerN	.+(E|O)R(S|)$	^NN.*
-ant	.+(E|A)NT(S|)$	^NN.*
-ment	.+MENT(S|)$	^NN.*
-age	.+AGE(S|)$	^NN.*
mis-	^MIS.+	^V.*
-able	.+ABLE$	^JJ.*
-ness	.+NESS$	^NN.*
-ful	.*FUL$	^J.*
-less	.*LESS$	^J.*

The number of types and new stems per affix

Affix	Types Total	New Stems
Vun-	24	1
Vde-	38	3
re-	179	15
-Aize	75	7
-Nize	91	5
-Aify	19	1
-Nify	22	1
-AenV	40	3
-ee	31	9
-VerN	634	147
-ant	30	5
-ment	182	11
-age	44	0
mis-	26	3
-able	236	16
-ness	318	9
-ful	79	2
-less	112	0

Precision for derived words

Feature	Affix	Types	Precision	Feature	Affix	Types	Precision
TELIC	*re-*	164	91%	RSTATE-EQ-BASE	*-Aify*	17	58%
BASE-RSTATE	*re-*	164	65%	COS	*-Nify*	21	67%
ENTAIL-BASE	*re-*	164	65%	COS	*-ate*	365	48%
REPRESUP	*re-*	164	65%	PART-IN-E	*-ee*	22	91%
COS	*un-*	23	100%	SENTIENT	*-ee*	22	82%
BASE-NEG-RSTATE	*un-*	23	91%	NON-VOL	*-ee*	22	68%
COS	*de-*	35	34%	PART-IN-E	*-er*	471	85%
BASE-NEG-RSTATE	*de-*	35	20%	PART-IN-E	*-ant*	21	81%
COS	*-Aize*	63	78%	E-AND-R	*-age*	43	58%
RSTATE-EQ-BASE	*-Aize*	63	75%	E-OR-P-OR-R	*-ment*	166	88%
COS	*-Nize*	86	56%	INCOR	*mis-*	21	86%
ACTIVITY	*-le*	71	55%	ABLE	*-able*	148	84%
COS	*-en*	36	100%	NESS	*-ness*	307	97%
RSTATE-EQ-BASE	*-en*	36	97%	FULANT	*-less*	22	77%
COS	*-Aify*	17	94%	LESSANT	*-ful*	22	77%

Precision for base words

Feature	Affix	Types	Precision	Feature	Affix	Types	Precision
TELIC	*re-*	164	91%	IZE-DEP	*-AenV*	36	72%
COS	*Vun-*	23	91%	IZE-DEP	*-Aify*	15	40%
COS	*Vde-*	33	36%	ABST	*-ful*	76	93%
IZE-DEP	*-Aize*	64	80%				

A.2 Word lists

The word lists have been organized by affix: for each affix all of its derived forms and/or bases are listed. (Some affixes do not provide any semantic information about one or the other of their bases or derived forms and these words are not listed.) For each word, its citation form is listed followed by a collection feature and a set of semantic features. The collection features give information about the word in relation to the Alvey dictionary, *e.g.*, was it in the lexicon, was its stem in the lexicon, etc. Below are the possible collection features for derived forms.

NEWDER: The word is a derived form and was not in the dictionary but its base was.

NEWSTEMDER: The word is a derived form and was constructed from a stem that was not in the dictionary.

INLEX: The word and its base, if there is one, were in the dictionary.

A number of additional collection features were needed to mark words which involve particular affixes. The DENOM feature was used to mark words ending in *-le* with a corresponding homonym noun that is not eventive. The NOTINLEX was used to mark *-ate* words that were not in the dictionary. The collection features for bases are listed below.

NEWDERSTEM: The word was in the dictionary but the corresponding derived form was not.

INLEX: The word and its derived form were in the dictionary.

NEWSTEM: The word was not in the dictionary.

The semantic features are defined by the axioms listed in the previous section. They are suffixed by the affix which cued them. A semantic feature may be prefixed by a "*" or a "?". A star signifies that the feature was posited by the morphological cueing process but in fact the word does not conform to the features axiom. A question mark signifies that it was indeterminate whether the word conformed. A feature holds if and only if it is true of the particular sense used in the majority of the tokens.

A.2.1 Words derived using *re-*

reache	INLEX	*TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
reacquaint	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
react	INLEX	*TELICre- *BASE-RSTATEre- BASEre- *REPRESUPre-
reactivate	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
readapt	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
readjust	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reaffirm	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
realign	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
realize	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
reape	INLEX	*TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
reappear	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reapportion	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
rearm	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
rearrange	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reassemble	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reassert	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reassign	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reassure	INLEX	TELICre- BASE-RSTATEre- BASEre- *REPRESUPre-
reawaken	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
rebell	INLEX	*TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
rebound	INLEX	*TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
rebuff	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
rebuild	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
rebutt	NEWDER	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recalculate	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
recall	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recant	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recapitulate	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recapture	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
receave	NEWSTEMDER	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recede	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
recheck	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
recite	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
reclaim	INLEX	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
reclassify	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
recoil	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recollect	INLEX	*TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recommence	NEWDER	TELICre- BASE-RSTATEre- BASEre- REPRESUPre-
recommend	INLEX	TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-
recond	NEWSTEMDER	*TELICre- *BASE-RSTATEre- *BASEre- *REPRESUPre-

recondition	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- REPRESUP*re*-
reconsider	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reconstruct	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reconvene	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reconvert	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
recoon	NEWSTEMDER	*TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
recooned	NEWSTEMDER	*TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
recopy	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
record	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
recount	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
recover	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
recreate	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redecorate	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
rededicate	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redeem	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
redefine	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redirect	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
rediscover	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redistribute	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redistrict	NEWSTEMDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redistricting	NEWSTEMDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redo	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redouble	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
redress	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
reek	INLEX	*TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
reelect	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reemerge	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reenact	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reenter	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reestablish	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reexamine	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refashion	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refill	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refinance	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refine	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
refine	NEWDER	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
refold	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reform	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reformulate	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refuel	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refund	INLEX	TELIC*re*- BASE-RSTATE*re*- *BASE*re*- REPRESUP*re*-
refurbish	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
refuse	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
regain	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
regenerate	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reground	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
regroup	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
rehash	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
rehear	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reincarnate	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reinstall	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reinterpret	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reinterpret	NEWSTEMDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reinterpreted	NEWSTEMDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reintroduce	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reiterate	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
rejoin	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
relay	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
relearn	NEWDER	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
release	INLEX	TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-
relie	INLEX	*TELIC*re*- *BASE-RSTATE*re*- **BASE*re*- *REPRESUP*re*-
relive	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
reload	INLEX	TELIC*re*- BASE-RSTATE*re*- BASE*re*- REPRESUP*re*-
rely	INLEX	*TELIC*re*- *BASE-RSTATE*re*- *BASE*re*- *REPRESUP*re*-

relyric	NEWSTEMDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
relyriced	NEWSTEMDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
remake	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
remark	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
remarry	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
remind	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
remodel	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	REPRESUPre-
remold	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
remount	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
remove	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
rename	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reopen	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reorder	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reorganize	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reorient	NEWSTEMDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reoriented	NEWSTEMDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
repaint	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
repair	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
repay	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
repeal	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
rephrase	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
replace	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
replant	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reply	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
report	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
repose	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
represent	INLEX	*TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
repress	INLEX	*TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
reprint	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reprobate	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reproduce	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
repulse	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
request	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
reread	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reseal	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
research	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
reserve	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
resettle	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reshape	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
reside	INLEX	*TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
resift	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
resign	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
resolve	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
resort	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
resound	INLEX	*TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
restate	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
resting	INLEX	*TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
restock	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
restore	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
restrain	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
restructure	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
restudy	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
resublime	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
resuspend	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
retail	INLEX	*TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
retell	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
rethink	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
retie	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
retire	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
retrace	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
retrain	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
retranslate	NEWDER	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-
retreat	INLEX	TELICre-	*BASE-RSTATEre-	*BASEre-	*REPRESUPre-
retrench	INLEX	TELICre-	BASE-RSTATEre-	BASEre-	REPRESUPre-

return	INLEX	TELIC*re-* *BASE-RSTATE*re-* *BASE*re-* *REPRESUP*re-*
reunite	INLEX	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*
reupholster	NEWDER	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*
revamp	INLEX	TELIC*re-* *BASE-RSTATE*re-* *BASE*re-* *REPRESUP*re-*
review	INLEX	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*
revisit	NEWSTEMDER	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*
revisited	NEWSTEMDER	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*
revitalize	INLEX	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*
revved	NEWSTEMDER	*TELIC*re-* *BASE-RSTATE*re-* *BASE*re-* *REPRESUP*re-*
reward	INLEX	TELIC*re-* *BASE-RSTATE*re-* *BASE*re-* *REPRESUP*re-*
rewrite	INLEX	TELIC*re-* BASE-RSTATE*re-* BASE*re-* REPRESUP*re-*

A.2.2 Words serving as bases for *re-*

ache	INLEX	*TELIC*re-*
acquaint	NEWDERSTEM	TELIC*re-*
act	INLEX	TELIC*re-*
activate	INLEX	TELIC*re-*
adapt	NEWDERSTEM	TELIC*re-*
adjust	INLEX	TELIC*re-*
affirm	NEWDERSTEM	TELIC*re-*
align	INLEX	TELIC*re-*
alize	INLEX	TELIC*re-*
ape	INLEX	*TELIC*re-*
appear	INLEX	*TELIC*re-*
apportion	NEWDERSTEM	TELIC*re-*
arm	INLEX	TELIC*re-*
arrange	INLEX	TELIC*re-*
assemble	NEWDERSTEM	TELIC*re-*
assert	NEWDERSTEM	TELIC*re-*
assign	NEWDERSTEM	TELIC*re-*
assure	INLEX	TELIC*re-*
awaken	NEWDERSTEM	TELIC*re-*
bell	INLEX	TELIC*re-*
bound	INLEX	*TELIC*re-*
buff	INLEX	*TELIC*re-*
build	INLEX	TELIC*re-*
butt	NEWDERSTEM	*TELIC*re-*
calculate	NEWDERSTEM	TELIC*re-*
call	INLEX	TELIC*re-*
cant	INLEX	TELIC*re-*
capitulate	INLEX	TELIC*re-*
capture	INLEX	TELIC*re-*
ceave	NEWSTEM	TELIC*re-*
cede	INLEX	TELIC*re-*
check	NEWDERSTEM	TELIC*re-*
cite	INLEX	TELIC*re-*
claim	INLEX	TELIC*re-*
classify	NEWDERSTEM	TELIC*re-*
coil	INLEX	TELIC*re-*
collect	INLEX	TELIC*re-*
commence	NEWDERSTEM	TELIC*re-*
commend	INLEX	TELIC*re-*
cond	NEWSTEM	TELIC*re-*
condition	INLEX	TELIC*re-*
consider	INLEX	TELIC*re-*
construct	INLEX	TELIC*re-*
convene	NEWDERSTEM	TELIC*re-*
convert	NEWDERSTEM	TELIC*re-*
coon	NEWSTEM	TELIC*re-*
cooned	NEWSTEM	TELIC*re-*
copy	NEWDERSTEM	TELIC*re-*

cord	INLEX	TELIC*re-*
count	INLEX	TELIC*re-*
cover	INLEX	TELIC*re-*
create	INLEX	TELIC*re-*
decorate	INLEX	TELIC*re-*
dedicate	NEWDERSTEM	TELIC*re-*
deem	INLEX	*TELIC*re-*
define	NEWDERSTEM	TELIC*re-*
direct	INLEX	TELIC*re-*
discover	NEWDERSTEM	TELIC*re-*
distribute	INLEX	TELIC*re-*
district	NEWSTEM	TELIC*re-*
districting	NEWSTEM	TELIC*re-*
do	INLEX	TELIC*re-*
double	INLEX	TELIC*re-*
dress	INLEX	TELIC*re-*
eke	INLEX	TELIC*re-*
elect	NEWDERSTEM	TELIC*re-*
emerge	NEWDERSTEM	TELIC*re-*
enact	NEWDERSTEM	TELIC*re-*
enter	NEWDERSTEM	TELIC*re-*
establish	NEWDERSTEM	TELIC*re-*
examine	NEWDERSTEM	TELIC*re-*
fashion	INLEX	TELIC*re-*
fill	INLEX	TELIC*re-*
fine	INLEX	TELIC*re-*
fine	NEWDERSTEM	TELIC*re-*
finance	NEWDERSTEM	TELIC*re-*
fold	NEWDERSTEM	TELIC*re-*
form	INLEX	TELIC*re-*
formulate	NEWDERSTEM	TELIC*re-*
fuel	INLEX	TELIC*re-*
fund	INLEX	TELIC*re-*
furbish	INLEX	TELIC*re-*
fuse	INLEX	TELIC*re-*
gain	INLEX	TELIC*re-*
generate	INLEX	TELIC*re-*
ground	NEWDERSTEM	TELIC*re-*
group	INLEX	TELIC*re-*
hash	INLEX	TELIC*re-*
hear	INLEX	TELIC*re-*
incarnate	INLEX	TELIC*re-*
install	NEWDERSTEM	TELIC*re-*
interpret	NEWDERSTEM	TELIC*re-*
interpret	NEWSTEM	TELIC*re-*
interpreted	NEWSTEM	TELIC*re-*
introduce	NEWDERSTEM	TELIC*re-*
iterate	INLEX	TELIC*re-*
join	INLEX	TELIC*re-*
lay	INLEX	TELIC*re-*
learn	NEWDERSTEM	TELIC*re-*
lease	INLEX	TELIC*re-*
lie	INLEX	*TELIC*re-*
live	INLEX	TELIC*re-*
load	INLEX	TELIC*re-*
ly	INLEX	TELIC*re-*
lyric	NEWSTEM	TELIC*re-*
lyriced	NEWSTEM	TELIC*re-*
make	INLEX	TELIC*re-*
mark	INLEX	TELIC*re-*
marry	INLEX	TELIC*re-*
mind	INLEX	*TELIC*re-*
model	INLEX	*TELIC*re-*
mold	INLEX	TELIC*re-*

mount	INLEX	TELIC*re-*
move	INLEX	TELIC*re-*
name	INLEX	TELIC*re-*
open	INLEX	TELIC*re-*
order	NEWDERSTEM	TELIC*re-*
organize	INLEX	TELIC*re-*
orient	NEWSTEM	TELIC*re-*
oriented	NEWSTEM	TELIC*re-*
paint	NEWDERSTEM	TELIC*re-*
pair	INLEX	TELIC*re-*
pay	INLEX	TELIC*re-*
peal	INLEX	TELIC*re-*
phrase	NEWDERSTEM	TELIC*re-*
place	INLEX	TELIC*re-*
plant	NEWDERSTEM	TELIC*re-*
ply	INLEX	TELIC*re-*
port	INLEX	TELIC*re-*
pose	INLEX	TELIC*re-*
present	INLEX	TELIC*re-*
press	INLEX	TELIC*re-*
print	INLEX	TELIC*re-*
probate	INLEX	TELIC*re-*
produce	INLEX	TELIC*re-*
pulse	INLEX	*TELIC*re-*
quest	INLEX	*TELIC*re-*
read	NEWDERSTEM	TELIC*re-*
seal	NEWDERSTEM	TELIC*re-*
search	INLEX	*TELIC*re-*
serve	INLEX	TELIC*re-*
settle	INLEX	TELIC*re-*
shape	NEWDERSTEM	TELIC*re-*
side	INLEX	TELIC*re-*
sift	NEWDERSTEM	TELIC*re-*
sign	INLEX	TELIC*re-*
solve	INLEX	TELIC*re-*
sort	INLEX	TELIC*re-*
sound	INLEX	TELIC*re-*
state	INLEX	TELIC*re-*
sting	INLEX	TELIC*re-*
stock	INLEX	TELIC*re-*
store	INLEX	*TELIC*re-*
strain	INLEX	*TELIC*re-*
structure	INLEX	TELIC*re-*
study	NEWDERSTEM	TELIC*re-*
sublime	NEWDERSTEM	TELIC*re-*
suspend	NEWDERSTEM	TELIC*re-*
tail	INLEX	TELIC*re-*
tell	INLEX	TELIC*re-*
think	INLEX	TELIC*re-*
tie	NEWDERSTEM	TELIC*re-*
tire	INLEX	TELIC*re-*
trace	INLEX	TELIC*re-*
train	NEWDERSTEM	TELIC*re-*
translate	NEWDERSTEM	TELIC*re-*
treat	INLEX	TELIC*re-*
trench	INLEX	TELIC*re-*
turn	INLEX	TELIC*re-*
unite	INLEX	TELIC*re-*
upholster	NEWDERSTEM	TELIC*re-*
vamp	INLEX	TELIC*re-*
view	INLEX	TELIC*re-*
visit	NEWSTEM	TELIC*re-*
visited	NEWSTEM	TELIC*re-*
vitalize	INLEX	TELIC*re-*

vved	NEWSTEM	TELIC *re-*
ward	INLEX	TELIC *re-*
write	INLEX	TELIC *re-*

A.2.3 Words derived using *un-*

unclasp	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
uncoil	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
uncork	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
uncurl	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
undo	INLEX	COS *Vun-* *BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
undress	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unearth	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unfasten	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unfold	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* *DYADIC *Vun-*
unhitch	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unlace	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unlash	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unload	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unlock	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unnerve	INLEX	COS *Vun-* *BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unpack	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unreel	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unscrew	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unsettle	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unsheath	NEWSTEMDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
untie	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unveil	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unwind	INLEX	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*
unwire	NEWDER	COS *Vun-* BASE-NEG-RSTATE *Vun-* DYADIC *Vun-*

A.2.4 Words serving as bases for *un-*

clasp	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
coil	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
cork	INLEX	COS *Vun-* DYADIC *Vun-*
curl	NEWDERSTEM	COS *Vun-* *DYADIC *Vun-*
do	INLEX	*COS *Vun-* DYADIC *Vun-*
dress	INLEX	COS *Vun-* DYADIC *Vun-*
earth	INLEX	COS *Vun-* DYADIC *Vun-*
fasten	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
fold	INLEX	COS *Vun-* DYADIC *Vun-*
hitch	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
lace	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
lash	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
load	INLEX	COS *Vun-* DYADIC *Vun-*
lock	INLEX	COS *Vun-* DYADIC *Vun-*
nerve	INLEX	*COS *Vun-* DYADIC *Vun-*
pack	INLEX	COS *Vun-* DYADIC *Vun-*
reel	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*
screw	INLEX	COS *Vun-* DYADIC *Vun-*
settle	INLEX	COS *Vun-* *DYADIC *Vun-*
sheath	NEWSTEM	COS *Vun-* DYADIC *Vun-*
tie	INLEX	COS *Vun-* DYADIC *Vun-*
veil	INLEX	COS *Vun-* DYADIC *Vun-*
wind	INLEX	COS *Vun-* DYADIC *Vun-*
wire	NEWDERSTEM	COS *Vun-* DYADIC *Vun-*

A.2.5 Words derived using *de-*

debunk	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
debut	NEWSTEMDER	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
decant	INLEX	COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
decentralize	INLEX	COS *Vde-* BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
declaim	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
decompose	INLEX	COS *Vde-* BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
decrease	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
decry	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
defend	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
define	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
defocus	NEWSTEMDER	COS *Vde-* BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
degenerate	INLEX	COS *Vde-* BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
degrade	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
delay	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
delight	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
delimit	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
delive	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
demean	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
demoralize	NEWDER	COS *Vde-* BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
demythologize	NEWDER	COS *Vde-* BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
denote	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
depart	INLEX	COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
depose	INLEX	COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
depress	INLEX	COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
derive	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
describe	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
desegregate	INLEX	COS *Vde-* BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
deserve	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
design	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
desire	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
despise	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
despoil	INLEX	COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*
desynchronize	NEWDER	COS *Vde-* BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
determ (typo)	NEWDER	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
detest	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
detour	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
develop	NEWSTEMDER	COS *Vde-* *BASE-NEG-RSTATE *Vde-* *DYADIC *Vde-*
devote	INLEX	*COS *Vde-* *BASE-NEG-RSTATE *Vde-* DYADIC *Vde-*

A.2.6 Words serving as bases for *de-*

bunk	INLEX	*COS *Vde-* *DYADIC *Vde-*
but	NEWSTEM	*COS *Vde-* *DYADIC *Vde-*
cant	INLEX	COS *Vde-* *DYADIC *Vde-*
centralize	INLEX	COS *Vde-* DYADIC *Vde-*
claim	INLEX	*COS *Vde-* *DYADIC *Vde-*
compose	INLEX	COS *Vde-* DYADIC *Vde-*
crease	INLEX	*COS *Vde-* *DYADIC *Vde-*
cry	INLEX	*COS *Vde-* *DYADIC *Vde-*
fend	INLEX	*COS *Vde-* *DYADIC *Vde-*
fine	INLEX	COS *Vde-* ?DYADIC *Vde-*
focus	NEWSTEM	COS *Vde-* DYADIC *Vde-*
generate	INLEX	COS *Vde-* DYADIC *Vde-*
grade	INLEX	COS *Vde-* DYADIC *Vde-*
lay	INLEX	*COS *Vde-* *DYADIC *Vde-*
light	INLEX	COS *Vde-* DYADIC *Vde-*
limit	INLEX	*COS *Vde-* *DYADIC *Vde-*
live	INLEX	*COS *Vde-* *DYADIC *Vde-*
mean	INLEX	*COS *Vde-* *DYADIC *Vde-*
moralize	NEWDERSTEM	COS *Vde-* DYADIC *Vde-*

mythologize	NEWDERSTEM	COS *Vde-*	DYADIC *Vde-*
note	INLEX	*COS *Vde-*	*DYADIC *Vde-*
part	INLEX	*COS *Vde-*	*DYADIC *Vde-*
pose	INLEX	*COS *Vde-*	*DYADIC *Vde-*
press	INLEX	*COS *Vde-*	*DYADIC *Vde-*
rive	INLEX	COS *Vde-*	?DYADIC *Vde-*
scribe	INLEX	*COS *Vde-*	*DYADIC *Vde-*
segregate	INLEX	COS *Vde-*	DYADIC *Vde-*
serve	INLEX	*COS *Vde-*	*DYADIC *Vde-*
sign	INLEX	*COS *Vde-*	*DYADIC *Vde-*
sire	INLEX	COS *Vde-*	DYADIC *Vde-*
spise (spy ize)	INLEX	*COS *Vde-*	*DYADIC *Vde-*
spoil	INLEX	COS *Vde-*	*DYADIC *Vde-*
synchronize	NEWDERSTEM	COS *Vde-*	*DYADIC *Vde-*
term (typo)	NEWDERSTEM	*COS *Vde-*	*DYADIC *Vde-*
test	INLEX	*COS *Vde-*	*DYADIC *Vde-*
tour	INLEX	*COS *Vde-*	*DYADIC *Vde-*
velop	NEWSTEM	*COS *Vde-*	*DYADIC *Vde-*
vote	INLEX	*COS *Vde-*	*DYADIC *Vde-*

A.2.7 Words derived using -*Aize*

balkanize	NEWDER	COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
brutalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
capitalize	INLEX	*COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
centralize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
characterize	INLEX	*COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
civilize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
communize	NEWSTEMDER	COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
conventionalize	NEWDER	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
demythologize	NEWSTEMDER	COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
economize	INLEX	?COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
equalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
federalize	NEWDER	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
fertilize	INLEX	*COS-*Aize*	?RSTATE-EQ-BASE-*Aize*
fetishize	NEWSTEMDER	?COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
formalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
fossilize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
generalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
germanize	NEWDER	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
humanize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
hypothesize	NEWSTEMDER	*COS-*Aize*	RSTATE-EQ-BASE-*Aize*
idealize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
immortalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
impersonalize	NEWDER	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
individualize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
industrialize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
internalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
internationalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
italicize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
legalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
liberalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
localize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
materialize	INLEX	COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
memorize	NEWSTEMDER	*COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
mineralize	NEWDER	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
mobilize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
modernize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*
moralize	NEWDER	COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
mythologize	NEWSTEMDER	*COS-*Aize*	RSTATE-EQ-BASE-*Aize*
narcotize	NEWSTEMDER	COS-*Aize*	*RSTATE-EQ-BASE-*Aize*
nationalize	INLEX	COS-*Aize*	RSTATE-EQ-BASE-*Aize*

naturalize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
neutralize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
normalize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
novelize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
penalize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
personalize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
polarize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
publicize	INLEX	*COS-*Aize* RSTATE-EQ-BASE-*Aize*
rationalize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
realize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
ritualize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
romanticize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
rubberize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
sectionalize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
secularize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
sentimentalize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
sexualize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
sidize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
signalize	INLEX	?COS-*Aize* RSTATE-EQ-BASE-*Aize*
socialize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
specialize	INLEX	?COS-*Aize* *RSTATE-EQ-BASE-*Aize*
stabilize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
standardize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
sterilize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
subsidize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
suburbanize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
summarize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
traditionalize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
tribalize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
universalize	NEWDER	COS-*Aize* RSTATE-EQ-BASE-*Aize*
urbanize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
visualize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*
vitalize	INLEX	COS-*Aize* RSTATE-EQ-BASE-*Aize*
vocalize	INLEX	*COS-*Aize* *RSTATE-EQ-BASE-*Aize*

A.2.8 Words serving as bases for -*Aize*

balkan (balk an)	NEWDERSTEM	IZE-DEP-*Aize*
brutal	INLEX	*IZE-DEP-*Aize*
capital	INLEX	*IZE-DEP-*Aize*
central	INLEX	IZE-DEP-*Aize*
character	INLEX	*IZE-DEP-*Aize*
civil	INLEX	IZE-DEP-*Aize*
commun	NEWSTEM	IZE-DEP-*Aize*
conventional	NEWDERSTEM	IZE-DEP-*Aize*
demytholog	NEWSTEM	?IZE-DEP-*Aize*
economy	INLEX	IZE-DEP-*Aize*
equal	INLEX	IZE-DEP-*Aize*
federal	NEWDERSTEM	IZE-DEP-*Aize*
fertil	INLEX	?IZE-DEP-*Aize*
fetish	NEWSTEM	IZE-DEP-*Aize*
formal	INLEX	IZE-DEP-*Aize*
fossil	INLEX	IZE-DEP-*Aize*
general	INLEX	IZE-DEP-*Aize*
german	NEWDERSTEM	IZE-DEP-*Aize*
human	INLEX	IZE-DEP-*Aize*
hypothes	NEWSTEM	IZE-DEP-*Aize*
ideal	INLEX	IZE-DEP-*Aize*
immortal	INLEX	IZE-DEP-*Aize*
impersonal	NEWDERSTEM	IZE-DEP-*Aize*
individual	INLEX	IZE-DEP-*Aize*
industrial	INLEX	IZE-DEP-*Aize*

internal	INLEX	IZE-DEP-*Aize*
international	INLEX	IZE-DEP-*Aize*
italic	INLEX	IZE-DEP-*Aize*
legal	INLEX	IZE-DEP-*Aize*
liberal	INLEX	IZE-DEP-*Aize*
local	INLEX	IZE-DEP-*Aize*
material	INLEX	IZE-DEP-*Aize*
memor	NEWSTEM	IZE-DEP-*Aize*
mineral	NEWDERSTEM	IZE-DEP-*Aize*
mobil	INLEX	IZE-DEP-*Aize*
modern	INLEX	IZE-DEP-*Aize*
moral	NEWDERSTEM	IZE-DEP-*Aize*
mytholog	NEWSTEM	IZE-DEP-*Aize*
narcot	NEWSTEM	IZE-DEP-*Aize*
national	INLEX	IZE-DEP-*Aize*
natural	INLEX	IZE-DEP-*Aize*
neutral	INLEX	IZE-DEP-*Aize*
normal	INLEX	IZE-DEP-*Aize*
novel	NEWDERSTEM	IZE-DEP-*Aize*
penal	INLEX	*IZE-DEP-*Aize*
personal	INLEX	IZE-DEP-*Aize*
polar	INLEX	IZE-DEP-*Aize*
public	INLEX	*IZE-DEP-*Aize*
rational	INLEX	*IZE-DEP-*Aize*
real	INLEX	*IZE-DEP-*Aize*
ritual	NEWDERSTEM	IZE-DEP-*Aize*
romantic	INLEX	*IZE-DEP-*Aize*
rubbery	INLEX	IZE-DEP-*Aize*
sectional	NEWDERSTEM	IZE-DEP-*Aize*
secular	INLEX	IZE-DEP-*Aize*
sentimental	INLEX	*IZE-DEP-*Aize*
sexual	NEWDERSTEM	IZE-DEP-*Aize*
side	INLEX	*IZE-DEP-*Aize*
signal	INLEX	IZE-DEP-*Aize*
social	INLEX	*IZE-DEP-*Aize*
special	INLEX	IZE-DEP-*Aize*
stabile	INLEX	*IZE-DEP-*Aize*
standard	INLEX	*IZE-DEP-*Aize*
steril	INLEX	*IZE-DEP-*Aize*
subside	INLEX	*IZE-DEP-*Aize*
suburban	NEWDERSTEM	IZE-DEP-*Aize*
summary	INLEX	*IZE-DEP-*Aize*
traditional	NEWDERSTEM	IZE-DEP-*Aize*
tribal	NEWDERSTEM	IZE-DEP-*Aize*
universal	NEWDERSTEM	IZE-DEP-*Aize*
urban	INLEX	IZE-DEP-*Aize*
visual	INLEX	*IZE-DEP-*Aize*
vital	INLEX	IZE-DEP-*Aize*
vocal	INLEX	IZE-DEP-*Aize*

A.2.9 Words derived using -*Nize*

aerosolize	NEWDER	COS-*Nize*
agonize	INLEX	*COS-*Nize*
alize	INLEX	*COS-*Nize*
apologize	INLEX	*COS-*Nize*
authorize	INLEX	COS-*Nize*
balkanize	NEWDER	COS-*Nize*
canonize	INLEX	COS-*Nize*
capitalize	INLEX	*COS-*Nize*
categorize	INLEX	COS-*Nize*
centralize	INLEX	COS-*Nize*

characterize	INLEX	*COS-*Nize*
colonize	INLEX	COS-*Nize*
communize	NEWDER	COS-*Nize*
criticize	INLEX	*COS-*Nize*
deputize	INLEX	COS-*Nize*
economize	INLEX	COS-*Nize*
energize	INLEX	COS-*Nize*
epitomize	INLEX	*COS-*Nize*
equalize	INLEX	COS-*Nize*
eulogize	INLEX	*COS-*Nize*
federalize	NEWSTEMDER	COS-*Nize*
fetishize	NEWDER	COS-*Nize*
fossilize	INLEX	COS-*Nize*
generalize	INLEX	*COS-*Nize*
germanize	NEWDER	COS-*Nize*
glamorize	INLEX	*COS-*Nize*
hospitalize	INLEX	COS-*Nize*
hypothesize	NEWSTEMDER	*COS-*Nize*
idealize	INLEX	*COS-*Nize*
idolize	INLEX	*COS-*Nize*
individualize	INLEX	COS-*Nize*
internationalize	INLEX	COS-*Nize*
ionize	INLEX	COS-*Nize*
italicize	INLEX	COS-*Nize*
itize	NEWDER	*COS-*Nize*
iversalize	NEWSTEMDER	*COS-*Nize*
jeopardize	INLEX	*COS-*Nize*
liberalize	INLEX	COS-*Nize*
lionize	INLEX	*COS-*Nize*
localize	INLEX	COS-*Nize*
materialize	INLEX	COS-*Nize*
maximize	INLEX	COS-*Nize*
memorialize	NEWDER	*COS-*Nize*
memorize	NEWDER	*COS-*Nize*
mineralize	NEWDER	COS-*Nize*
mobilize	INLEX	COS-*Nize*
modernize	INLEX	COS-*Nize*
monopolize	INLEX	COS-*Nize*
moralize	NEWDER	COS-*Nize*
mythologize	NEWDER	COS-*Nize*
narcotize	NEWSTEMDER	COS-*Nize*
nationalize	INLEX	COS-*Nize*
naturalize	INLEX	COS-*Nize*
neutralize	INLEX	COS-*Nize*
notarize	INLEX	COS-*Nize*
novelize	NEWDER	COS-*Nize*
organize	INLEX	*COS-*Nize*
panelize	NEWDER	COS-*Nize*
patronize	INLEX	*COS-*Nize*
philosophize	INLEX	*COS-*Nize*
poetize	NEWDER	COS-*Nize*
proselytize	INLEX	COS-*Nize*
publicize	INLEX	COS-*Nize*
rationalize	INLEX	*COS-*Nize*
realize	INLEX	*COS-*Nize*
revolutionize	INLEX	COS-*Nize*
ritualize	NEWDER	COS-*Nize*
romanticize	INLEX	*COS-*Nize*
rubberize	INLEX	COS-*Nize*
satirize	INLEX	*COS-*Nize*
scandalize	INLEX	*COS-*Nize*
scrutinize	INLEX	*COS-*Nize*
sidize	INLEX	*COS-*Nize*
signalize	INLEX	*COS-*Nize*

socialize	INLEX	*COS-*Nize*
specialize	INLEX	*COS-*Nize*
stabilize	INLEX	COS-*Nize*
standardize	INLEX	COS-*Nize*
stylize	INLEX	COS-*Nize*
subsidize	INLEX	*COS-*Nize*
suburbanize	NEWDER	COS-*Nize*
summarize	INLEX	*COS-*Nize*
symbolize	INLEX	*COS-*Nize*
sympathize	INLEX	*COS-*Nize*
terrorize	INLEX	*COS-*Nize*
theorize	INLEX	*COS-*Nize*
tribalize	NEWDER	COS-*Nize*
tyrannize	INLEX	*COS-*Nize*
unitize	NEWDER	COS-*Nize*
universalize	NEWSTEMDER	COS-*Nize*
vocalize	INLEX	*COS-*Nize*

A.2.10 Words derived using -*Aify*

amplify	INLEX	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
beautify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
classify	INLEX	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
diversify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
falsify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
fortify	INLEX	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
intensify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
justify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
nullify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
oversimplify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
purify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
rarify	NEWDER	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
scarify	INLEX	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
simplify	INLEX	COS-*Aify*	RSTATE-EQ-BASE-*Aify*
testify	INLEX	*COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
verify	INLEX	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
vilify	INLEX	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*
vivify	NEWSTEMDER	COS-*Aify*	*RSTATE-EQ-BASE-*Aify*

A.2.11 Words serving as bases for -*Aify*

ample	INLEX	*IZE-DEP-*Aify*
beaut	INLEX	?IZE-DEP-*Aify*
classy	INLEX	*IZE-DEP-*Aify*
diverse	INLEX	IZE-DEP-*Aify*
false	INLEX	*IZE-DEP-*Aify*
forte	INLEX	*IZE-DEP-*Aify*
intense	INLEX	IZE-DEP-*Aify*
just	INLEX	*IZE-DEP-*Aify*
null	INLEX	IZE-DEP-*Aify*
oversimple	INLEX	IZE-DEP-*Aify*
pure	INLEX	IZE-DEP-*Aify*
rare	NEWDERSTEM	*IZE-DEP-*Aify*
scary	INLEX	*IZE-DEP-*Aify*
simple	INLEX	IZE-DEP-*Aify*
testy	INLEX	*IZE-DEP-*Aify*
very	INLEX	*IZE-DEP-*Aify*
vile	INLEX	*IZE-DEP-*Aify*
viv	NEWSTEM	*IZE-DEP-*Aify*

A.2.12 Words derived using *-Nify*

beautify	INLEX	COS-*Nify*
boobify	NEWDER	COS-*Nify*
certify	INLEX	COS-*Nify*
classify	INLEX	*COS-*Nify*
codify	INLEX	*COS-*Nify*
fortify	INLEX	COS-*Nify*
glorify	INLEX	*COS-*Nify*
gratify	INLEX	COS-*Nify*
minify	NEWDER	COS-*Nify*
modify	INLEX	*COS-*Nify*
mollify	INLEX	COS-*Nify*
mummify	INLEX	COS-*Nify*
notify	INLEX	*COS-*Nify*
pacify	INLEX	COS-*Nify*
personify	INLEX	*COS-*Nify*
ratify	INLEX	COS-*Nify*
scarify	INLEX	COS-*Nify*
signify	INLEX	*COS-*Nify*
simplify	INLEX	COS-*Nify*
testify	INLEX	COS-*Nify*
typify	INLEX	COS-*Nify*
vivify	NEWSTEMDER	COS-*Nify*

A.2.13 Words derived using *-en*

awaken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
bemadden	NEWSTEMDER	COS-*AenV*	*RSTATE-EQ-BASE-*AenV*
blacken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
brighten	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
broaden	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
coarsen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
dampen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
darken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
deaden	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
deafen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
deepen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
fasten	INLEX	COS-*AenV*	*RSTATE-EQ-BASE-*AenV*
freshen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
green	NEWSTEMDER	COS-*AenV*	*RSTATE-EQ-BASE-*AenV*
harden	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
harshen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
lessen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
lighten	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
liken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
loosen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
moisten	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
quicken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
ripen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
roughen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
sharpen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
shorten	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
sicken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
slacken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
soften	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
stiffen	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
straighten	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
strengten	NEWSTEMDER	*COS-*AenV*	*RSTATE-EQ-BASE-*AenV*
thicken	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
tighten	INLEX	COS-*AenV*	RSTATE-EQ-BASE-*AenV*
unfasten	NEWDER	COS-*AenV*	RSTATE-EQ-BASE-*AenV*

weaken	INLEX	COS-*AenV* RSTATE-EQ-BASE-*AenV*
whiten	INLEX	COS-*AenV* RSTATE-EQ-BASE-*AenV*
widen	INLEX	COS-*AenV* RSTATE-EQ-BASE-*AenV*
worsen	INLEX	COS-*AenV* RSTATE-EQ-BASE-*AenV*

A.2.14 Words serving as bases for *-en*

awake	INLEX	*IZE-DEP-*AenV*
bemadd	NEWSTEM	*IZE-DEP-*AenV*
black	INLEX	*IZE-DEP-*AenV*
bright	INLEX	IZE-DEP-*AenV*
broad	INLEX	IZE-DEP-*AenV*
coarse	INLEX	IZE-DEP-*AenV*
damp	INLEX	IZE-DEP-*AenV*
dark	INLEX	IZE-DEP-*AenV*
dead	INLEX	*IZE-DEP-*AenV*
deaf	INLEX	*IZE-DEP-*AenV*
deep	INLEX	IZE-DEP-*AenV*
fast	INLEX	*IZE-DEP-*AenV*
fresh	INLEX	IZE-DEP-*AenV*
gre	NEWSTEM	*IZE-DEP-*AenV*
hard	INLEX	IZE-DEP-*AenV*
harsh	INLEX	IZE-DEP-*AenV*
less	INLEX	*IZE-DEP-*AenV*
light	INLEX	IZE-DEP-*AenV*
like	INLEX	*IZE-DEP-*AenV*
loose	INLEX	IZE-DEP-*AenV*
moist	INLEX	IZE-DEP-*AenV*
quick	INLEX	IZE-DEP-*AenV*
ripe	INLEX	IZE-DEP-*AenV*
rough	INLEX	IZE-DEP-*AenV*
sharp	INLEX	IZE-DEP-*AenV*
short	INLEX	IZE-DEP-*AenV*
sick	INLEX	*IZE-DEP-*AenV*
slack	INLEX	IZE-DEP-*AenV*
soft	INLEX	IZE-DEP-*AenV*
stiff	INLEX	IZE-DEP-*AenV*
straight	INLEX	IZE-DEP-*AenV*
strength	NEWSTEM	*IZE-DEP-*AenV*
thick	INLEX	IZE-DEP-*AenV*
tight	INLEX	IZE-DEP-*AenV*
unfast	NEWDERSTEM	*IZE-DEP-*AenV*
weak	INLEX	IZE-DEP-*AenV*
white	INLEX	IZE-DEP-*AenV*
wide	INLEX	IZE-DEP-*AenV*
worse	INLEX	*IZE-DEP-*AenV*

A.2.15 Words derived using *-ate*

abate	INLEX	COS-*ate*
abbreviate	INLEX	COS-*ate*
ablate	NOTINLEX	COS-*ate*
abrogate	INLEX	COS-*ate*
accelerate	INLEX	COS-*ate*
accentuate	INLEX	COS-*ate*
accommodate	INLEX	*COS-*ate*
acculturate	NOTINLEX	COS-*ate*
accumulate	INLEX	*COS-*ate*
activate	INLEX	COS-*ate*

actuate	INLEX	COS-*ate*
adjudicate	INLEX	*COS-*ate*
adulterate	INLEX	COS-*ate*
advocate	INLEX	*COS-*ate*
aerate	INLEX	*COS-*ate*
affiliate	INLEX	*COS-*ate*
agglomerate	INLEX	COS-*ate*
agglutinate	NOTINLEX	COS-*ate*
aggravate	INLEX	*COS-*ate*
agitate	INLEX	*COS-*ate*
alienate	INLEX	*COS-*ate*
alleviate	INLEX	COS-*ate*
allocate	INLEX	COS-*ate*
alternate	INLEX	*COS-*ate*
amalgamate	INLEX	COS-*ate*
amputate	INLEX	COS-*ate*
animate	INLEX	COS-*ate*
annihilate	INLEX	COS-*ate*
annunciate	NOTINLEX	*COS-*ate*
anticipate	INLEX	*COS-*ate*
antiquate	NOTINLEX	COS-*ate*
appreciate	INLEX	*COS-*ate*
appropriate	INLEX	COS-*ate*
approximate	INLEX	*COS-*ate*
arbitrate	INLEX	*COS-*ate*
arrogate	INLEX	COS-*ate*
articulate	INLEX	*COS-*ate*
assassinate	INLEX	COS-*ate*
assimilate	INLEX	COS-*ate*
associate	INLEX	*COS-*ate*
ate	INLEX	*COS-*ate*
authenticate	INLEX	*COS-*ate*
automate	INLEX	COS-*ate*
beat	INLEX	*COS-*ate*
berate	INLEX	*COS-*ate*
bleat	INLEX	*COS-*ate*
bloat	INLEX	*COS-*ate*
boat	INLEX	*COS-*ate*
calculate	INLEX	*COS-*ate*
calibrate	INLEX	COS-*ate*
calumniate	INLEX	*COS-*ate*
capitulate	INLEX	COS-*ate*
captivate	INLEX	COS-*ate*
castigate	INLEX	*COS-*ate*
celebrate	INLEX	*COS-*ate*
cheat	INLEX	*COS-*ate*
circulate	INLEX	*COS-*ate*
co-operate	NOTINLEX	*COS-*ate*
co-ordinate	NOTINLEX	*COS-*ate*
coagulate	INLEX	COS-*ate*
coat	INLEX	*COS-*ate*
collaborate	INLEX	*COS-*ate*
collate	INLEX	COS-*ate*
collimate	NOTINLEX	COS-*ate*
combat	INLEX	*COS-*ate*
commemorate	INLEX	*COS-*ate*
commiserate	INLEX	*COS-*ate*
communicate	INLEX	*COS-*ate*
compensate	INLEX	*COS-*ate*
compleat	NOTINLEX	COS-*ate*
complicate	INLEX	*COS-*ate*
concentrate	INLEX	*COS-*ate*
conciliate	INLEX	COS-*ate*
confabulate	INLEX	*COS-*ate*

confiscate	INLEX	COS-*ate*
congratulate	INLEX	*COS-*ate*
congregate	INLEX	COS-*ate*
conjugate	INLEX	COS-*ate*
consolidate	INLEX	COS-*ate*
consummate	INLEX	COS-*ate*
contaminate	INLEX	COS-*ate*
contemplate	INLEX	*COS-*ate*
cooperate	INLEX	*COS-*ate*
coordinate	INLEX	*COS-*ate*
correlate	INLEX	COS-*ate*
corroborate	INLEX	*COS-*ate*
corrugate	INLEX	COS-*ate*
create	INLEX	COS-*ate*
cremate	INLEX	COS-*ate*
culminate	INLEX	COS-*ate*
cultivate	INLEX	*COS-*ate*
cumulate	NOTINLEX	COS-*ate*
date	INLEX	*COS-*ate*
de-iodinate	NOTINLEX	COS-*ate*
deactivate	NEWDER	COS-*ate*
debate	INLEX	*COS-*ate*
debilitate	INLEX	COS-*ate*
decelerate	INLEX	COS-*ate*
decorate	INLEX	*COS-*ate*
decorticate	NOTINLEX	COS-*ate*
dedicate	INLEX	*COS-*ate*
dedifferentiate	NEWDER	COS-*ate*
defeat	INLEX	*COS-*ate*
defecate	INLEX	*COS-*ate*
deflate	INLEX	COS-*ate*
degenerate	INLEX	COS-*ate*
dehydrate	INLEX	COS-*ate*
delegate	INLEX	*COS-*ate*
delineate	INLEX	*COS-*ate*
demarcate	INLEX	*COS-*ate*
demonstrate	INLEX	*COS-*ate*
denominate	INLEX	*COS-*ate*
derogate	INLEX	COS-*ate*
desecrate	INLEX	*COS-*ate*
desegregate	INLEX	COS-*ate*
designate	INLEX	COS-*ate*
deteriorate	INLEX	COS-*ate*
detonate	INLEX	COS-*ate*
deuterate	NOTINLEX	COS-*ate*
devastate	INLEX	COS-*ate*
deviate	INLEX	*COS-*ate*
dictate	INLEX	*COS-*ate*
differentiate	INLEX	*COS-*ate*
dilapidate	INLEX	COS-*ate*
dilate	INLEX	COS-*ate*
disaffiliate	INLEX	COS-*ate*
discorporate	NOTINLEX	COS-*ate*
discriminate	INLEX	*COS-*ate*
disintegrate	INLEX	COS-*ate*
dislocate	INLEX	COS-*ate*
disseminate	INLEX	COS-*ate*
dissipate	INLEX	COS-*ate*
dissociate	INLEX	COS-*ate*
dominate	INLEX	*COS-*ate*
donate	INLEX	COS-*ate*
duplicate	INLEX	*COS-*ate*
eat	INLEX	*COS-*ate*
educate	INLEX	COS-*ate*

effectuate	INLEX	COS-*ate*
ejaculate	INLEX	COS-*ate*
elaborate	INLEX	*COS-*ate*
elate	INLEX	COS-*ate*
elevate	INLEX	COS-*ate*
eliminate	INLEX	COS-*ate*
elongate	INLEX	COS-*ate*
elucidate	INLEX	COS-*ate*
emaciate	INLEX	COS-*ate*
emanate	INLEX	*COS-*ate*
emancipate	INLEX	COS-*ate*
emasculate	INLEX	COS-*ate*
emigrate	INLEX	COS-*ate*
emulate	INLEX	*COS-*ate*
enervate	INLEX	COS-*ate*
entreat	INLEX	*COS-*ate*
enumerate	INLEX	*COS-*ate*
enunciate	INLEX	*COS-*ate*
equate	INLEX	*COS-*ate*
equilibrate	NOTINLEX	COS-*ate*
eradicate	INLEX	COS-*ate*
estimate	INLEX	*COS-*ate*
evacuate	INLEX	COS-*ate*
evaporate	INLEX	COS-*ate*
eventuate	INLEX	COS-*ate*
exacerbate	INLEX	COS-*ate*
exaggerate	INLEX	*COS-*ate*
exasperate	INLEX	COS-*ate*
excommunicate	INLEX	COS-*ate*
excoriate	INLEX	*COS-*ate*
exhilarate	INLEX	COS-*ate*
exonerate	INLEX	COS-*ate*
expiate	INLEX	*COS-*ate*
expropriate	INLEX	COS-*ate*
extenuate	INLEX	COS-*ate*
exterminate	INLEX	COS-*ate*
extirpate	INLEX	COS-*ate*
extrapolate	INLEX	*COS-*ate*
extricate	INLEX	COS-*ate*
fabricate	INLEX	COS-*ate*
facilitate	INLEX	*COS-*ate*
fascinate	INLEX	*COS-*ate*
flagellate	INLEX	*COS-*ate*
float	INLEX	*COS-*ate*
flocculate	NOTINLEX	COS-*ate*
fluctuate	INLEX	*COS-*ate*
fluorinate	NOTINLEX	COS-*ate*
formulate	INLEX	*COS-*ate*
fractionate	NOTINLEX	COS-*ate*
frustrate	INLEX	*COS-*ate*
fulminate	INLEX	*COS-*ate*
generate	INLEX	COS-*ate*
germinate	INLEX	COS-*ate*
gesticulate	INLEX	COS-*ate*
gloat	INLEX	*COS-*ate*
glycerinate	NOTINLEX	COS-*ate*
graduate	INLEX	COS-*ate*
grate	INLEX	COS-*ate*
hallucinate	INLEX	*COS-*ate*
hate	INLEX	*COS-*ate*
heat	INLEX	COS-*ate*
hesitate	INLEX	*COS-*ate*
hibernate	INLEX	*COS-*ate*
humiliate	INLEX	*COS-*ate*

hydrate	NOTINLEX	COS-*ate*
hyphenate	INLEX	COS-*ate*
illuminate	INLEX	COS-*ate*
illustrate	INLEX	*COS-*ate*
imitate	INLEX	*COS-*ate*
impersonate	INLEX	*COS-*ate*
implicate	INLEX	*COS-*ate*
imprecate	NOTINLEX	*COS-*ate*
inactivate	NOTINLEX	COS-*ate*
inaugurate	INLEX	COS-*ate*
incapacitate	INLEX	COS-*ate*
incarcerate	INLEX	COS-*ate*
incarnate	INLEX	COS-*ate*
incorporate	INLEX	COS-*ate*
incriminate	INLEX	*COS-*ate*
incubate	INLEX	COS-*ate*
inculcate	INLEX	COS-*ate*
indicate	INLEX	*COS-*ate*
indoctrinate	INLEX	COS-*ate*
infiltrate	INLEX	COS-*ate*
inflate	INLEX	COS-*ate*
infuriate	INLEX	COS-*ate*
initiate	INLEX	COS-*ate*
innovate	INLEX	COS-*ate*
insinuate	INLEX	*COS-*ate*
instigate	INLEX	COS-*ate*
insulate	INLEX	COS-*ate*
integrate	INLEX	COS-*ate*
interpenetrate	NEWDER	COS-*ate*
interpolate	INLEX	COS-*ate*
interrelate	INLEX	*COS-*ate*
intimate	INLEX	*COS-*ate*
intoxicate	INLEX	COS-*ate*
inundate	INLEX	COS-*ate*
invalidate	INLEX	COS-*ate*
investigate	INLEX	*COS-*ate*
invigorate	INLEX	COS-*ate*
iodinate	NOTINLEX	COS-*ate*
irradiate	INLEX	*COS-*ate*
irrigate	INLEX	COS-*ate*
irritate	INLEX	*COS-*ate*
isolate	INLEX	COS-*ate*
lacerate	INLEX	COS-*ate*
lactate	NOTINLEX	*COS-*ate*
laminated	INLEX	COS-*ate*
legislate	INLEX	COS-*ate*
liberate	INLEX	COS-*ate*
liquidate	INLEX	COS-*ate*
locate	INLEX	*COS-*ate*
lubricate	INLEX	COS-*ate*
mandate	INLEX	COS-*ate*
manipulate	INLEX	*COS-*ate*
marinate	INLEX	COS-*ate*
mate	INLEX	*COS-*ate*
matriculate	INLEX	COS-*ate*
mediate	INLEX	*COS-*ate*
meditate	INLEX	*COS-*ate*
migrate	INLEX	COS-*ate*
militate	INLEX	*COS-*ate*
miscalculate	INLEX	*COS-*ate*
misrelate	NEWDER	*COS-*ate*
mitigate	INLEX	COS-*ate*
moderate	INLEX	COS-*ate*
modulate	INLEX	*COS-*ate*

motivate	INLEX	COS-*ate*
mutilate	INLEX	COS-*ate*
narrate	INLEX	*COS-*ate*
nauseate	INLEX	COS-*ate*
navigate	INLEX	*COS-*ate*
necessitate	INLEX	*COS-*ate*
negate	INLEX	COS-*ate*
negotiate	INLEX	*COS-*ate*
nominate	INLEX	*COS-*ate*
nucleate	NOTINLEX	COS-*ate*
obligate	INLEX	*COS-*ate*
obliterate	INLEX	COS-*ate*
officiate	INLEX	*COS-*ate*
operate	INLEX	*COS-*ate*
orate	NOTINLEX	*COS-*ate*
originate	INLEX	*COS-*ate*
oscillate	INLEX	*COS-*ate*
outdate	NEWDER	*COS-*ate*
overeat	NEWDER	*COS-*ate*
overestimate	INLEX	*COS-*ate*
overheat	INLEX	COS-*ate*
overpopulate	NEWDER	COS-*ate*
overrate	INLEX	*COS-*ate*
paginate	NOTINLEX	COS-*ate*
participate	INLEX	*COS-*ate*
penetrate	INLEX	COS-*ate*
perforate	INLEX	COS-*ate*
permeate	INLEX	COS-*ate*
perpetrate	INLEX	*COS-*ate*
perpetuate	INLEX	COS-*ate*
placate	INLEX	COS-*ate*
plate	INLEX	COS-*ate*
pontificate	INLEX	*COS-*ate*
populate	INLEX	COS-*ate*
postulate	INLEX	*COS-*ate*
precipitate	INLEX	*COS-*ate*
predominate	INLEX	*COS-*ate*
prefabricate	INLEX	COS-*ate*
preisolate	NEWDER	COS-*ate*
preponderate	INLEX	*COS-*ate*
procrastinate	INLEX	*COS-*ate*
proliferate	INLEX	*COS-*ate*
promulgate	INLEX	COS-*ate*
propagate	INLEX	COS-*ate*
propitiate	INLEX	*COS-*ate*
prorate	NOTINLEX	*COS-*ate*
pulsate	INLEX	*COS-*ate*
punctuate	INLEX	COS-*ate*
pupate	INLEX	COS-*ate*
radiate	INLEX	*COS-*ate*
rate	INLEX	*COS-*ate*
ratiocinate	NOTINLEX	COS-*ate*
re-activate	NEWDER	COS-*ate*
re-create	NEWDER	*COS-*ate*
re-evaluate	NEWDER	*COS-*ate*
re-incorporate	NEWDER	COS-*ate*
reactivate	INLEX	COS-*ate*
recalculate	NEWDER	*COS-*ate*
recapitulate	INLEX	*COS-*ate*
reciprocate	INLEX	?COS-*ate* ?
recreate	INLEX	*COS-*ate*
recuperate	INLEX	COS-*ate*
redecorate	INLEX	*COS-*ate*
rededicate	NEWDER	*COS-*ate*

reformulate	NEWDER	*COS-*ate*
refrigerate	INLEX	COS-*ate*
regenerate	INLEX	*COS-*ate*
regulate	INLEX	*COS-*ate*
rehabilitate	INLEX	*COS-*ate*
reincarnate	INLEX	*COS-*ate*
reinstate	INLEX	*COS-*ate*
reiterate	INLEX	*COS-*ate*
relate	INLEX	*COS-*ate*
relegate	INLEX	COS-*ate*
remonstrate	INLEX	*COS-*ate*
renovate	INLEX	*COS-*ate*
repeat	INLEX	*COS-*ate*
reprobate	INLEX	*COS-*ate*
repudiate	INLEX	*COS-*ate*
restate	INLEX	*COS-*ate*
retaliate	INLEX	*COS-*ate*
retranslate	NEWDER	*COS-*ate*
retreat	INLEX	*COS-*ate*
reverberate	INLEX	*COS-*ate*
rotate	INLEX	*COS-*ate*
salivate	INLEX	*COS-*ate*
satiate	INLEX	COS-*ate*
saturate	INLEX	COS-*ate*
scintillate	INLEX	*COS-*ate*
seat	INLEX	COS-*ate*
segregate	INLEX	COS-*ate*
separate	INLEX	COS-*ate*
simulate	INLEX	*COS-*ate*
situate	NOTINLEX	*COS-*ate*
skate	INLEX	*COS-*ate*
slate	INLEX	*COS-*ate*
solvate	NOTINLEX	COS-*ate*
speculate	INLEX	*COS-*ate*
state	INLEX	*COS-*ate*
stimulate	INLEX	COS-*ate*
stipulate	INLEX	*COS-*ate*
subjugate	INLEX	COS-*ate*
sublimate	INLEX	COS-*ate*
subordinate	INLEX	COS-*ate*
substantiate	INLEX	*COS-*ate*
suffocate	INLEX	COS-*ate*
summate	NOTINLEX	COS-*ate*
supplicate	INLEX	*COS-*ate*
sweat	INLEX	*COS-*ate*
syndicate	INLEX	COS-*ate*
tabulate	INLEX	*COS-*ate*
terminate	INLEX	COS-*ate*
thoriate	NOTINLEX	COS-*ate*
tinplate	NOTINLEX	COS-*ate*
titillate	INLEX	COS-*ate*
tolerate	INLEX	*COS-*ate*
translate	INLEX	*COS-*ate*
transpirate	NOTINLEX	COS-*ate*
treat	INLEX	*COS-*ate*
truncate	INLEX	COS-*ate*
ulcerate	INLEX	COS-*ate*
underestimate	INLEX	*COS-*ate*
underrate	INLEX	*COS-*ate*
understate	INLEX	*COS-*ate*
undulate	INLEX	*COS-*ate*
update	INLEX	COS-*ate*
vacate	INLEX	COS-*ate*
vaccinate	INLEX	COS-*ate*

vacuolate	NOTINLEX	COS-*ate*
validate	INLEX	COS-*ate*
variegate	NOTINLEX	COS-*ate*
venerate	INLEX	*COS-*ate*
ventilate	INLEX	*COS-*ate*
vibrate	INLEX	*COS-*ate*
vindicate	INLEX	*COS-*ate*
violate	INLEX	*COS-*ate*
vitiate	INLEX	COS-*ate*

A.2.16 Words derived using *-le*

angle	INLEX DENOM	ACTIVITY-*leV*
assemble	INLEX	*ACTIVITY-*leV*
battle	INLEX DENOM	ACTIVITY-*leV*
buckle	INLEX DENOM	*ACTIVITY-*leV*
bundle	INLEX DENOM	*ACTIVITY-*leV*
cable	INLEX DENOM	*ACTIVITY-*leV*
chandelle		*ACTIVITY-*leV*
chuckle	INLEX DENOM	ACTIVITY-*leV*
circle	INLEX DENOM	ACTIVITY-*leV*
crumble	INLEX DENOM	*ACTIVITY-*leV*
dangle	INLEX	ACTIVITY-*leV*
dazzle	INLEX DENOM	ACTIVITY-*leV*
disable	INLEX	*ACTIVITY-*leV*
disassemble	INLEX	*ACTIVITY-*leV*
disentangle	INLEX	*ACTIVITY-*leV*
double	INLEX DENOM	*ACTIVITY-*leV*
dwindle	INLEX	ACTIVITY-*leV*
embezzle	INLEX	ACTIVITY-*leV*
enable	INLEX	*ACTIVITY-*leV*
encircle	INLEX	ACTIVITY-*leV*
entitle	INLEX	ACTIVITY-*leV*
gamble	INLEX DENOM	ACTIVITY-*leV*
grapple	INLEX	ACTIVITY-*leV*
grumble	INLEX DENOM	ACTIVITY-*leV*
guzzle	INLEX	*ACTIVITY-*leV*
handle	INLEX DENOM	ACTIVITY-*leV*
hobble	INLEX	ACTIVITY-*leV*
humble	INLEX	*ACTIVITY-*leV*
hurdle	INLEX DENOM	*ACTIVITY-*leV*
hustle	INLEX DENOM	ACTIVITY-*leV*
jostle	INLEX	ACTIVITY-*leV*
knuckle	INLEX DENOM	ACTIVITY-*leV*
meddle	INLEX	ACTIVITY-*leV*
mingle	INLEX	ACTIVITY-*leV*
nolle	INLEX	*ACTIVITY-*leV*
paddle	INLEX DENOM	ACTIVITY-*leV*
peddle	INLEX	ACTIVITY-*leV*
people	INLEX DENOM	*ACTIVITY-*leV*
prevayle		*ACTIVITY-*leV*
puzzle	INLEX DENOM	ACTIVITY-*leV*
quadruple	INLEX	*ACTIVITY-*leV*
quibble	INLEX DENOM	ACTIVITY-*leV*
ramble	INLEX DENOM	ACTIVITY-*leV*
reassemble	INLEX DENOM	*ACTIVITY-*leV*
resemble	INLEX	*ACTIVITY-*leV*
riffle	INLEX	ACTIVITY-*leV*
ripple	INLEX DENOM	ACTIVITY-*leV*
rustle	INLEX DENOM	ACTIVITY-*leV*
saddle	INLEX DENOM	*ACTIVITY-*leV*
sample	INLEX DENOM	*ACTIVITY-*leV*

settle	INLEX DENOM	*ACTIVITY-*le V*
shuffle	INLEX DENOM	ACTIVITY-*le V*
smuggle	INLEX	*ACTIVITY-*le V*
sprinkle	INLEX DENOM	ACTIVITY-*le V*
startle	INLEX	*ACTIVITY-*le V*
stifle	INLEX	*ACTIVITY-*le V*
straggle	INLEX	*ACTIVITY-*le V*
struggle	INLEX DENOM	ACTIVITY-*le V*
stumble	INLEX DENOM	ACTIVITY-*le V*
table	INLEX DENOM	*ACTIVITY-*le V*
tackle	INLEX DENOM	*ACTIVITY-*le V*
tangle	INLEX DENOM	ACTIVITY-*le V*
topple	INLEX	*ACTIVITY-*le V*
trample	INLEX	*ACTIVITY-*le V*
tremble	INLEX DENOM	ACTIVITY-*le V*
trouble	INLEX DENOM	ACTIVITY-*le V*
tumble	INLEX DENOM	ACTIVITY-*le V*
unscramble	INLEX	*ACTIVITY-*le V*
whistle	INLEX DENOM	ACTIVITY-*le V*
wobble	INLEX DENOM	ACTIVITY-*le V*
wrestle	INLEX	ACTIVITY-*le V*

A.2.17 Words derived using *-ee*

absentee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* *NON-VOL-*ee*
addressee	INLEX	PART-IN-E-*ee* *SENTIENT-*ee* NON-VOL-*ee*
appointee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
assignee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
conferee	NEWSTEMDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
deportee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
devisee	NEWDER	PART-IN-E-*ee* *SENTIENT-*ee* NON-VOL-*ee*
devotee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* *NON-VOL-*ee*
divorcee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* *NON-VOL-*ee*
draftee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
emcee	NEWSTEMDER	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
employee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
enrollee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* *NON-VOL-*ee*
escapee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* *NON-VOL-*ee*
filagree	NEWSTEMDER	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
honoree	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
inductee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
interviewee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
invitee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
licensee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
millidegree	NEWSTEMDER	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
nferee	NEWSTEMDER	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
parolee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
patentee	INLEX	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
repartee	INLEX	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
sangaree	NEWSTEMDER	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
three	NEWSTEMDER	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
transferee	NEWSTEMDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*
trustee	INLEX	*PART-IN-E-*ee* *SENTIENT-*ee* *NON-VOL-*ee*
viewee	NEWDER	PART-IN-E-*ee* SENTIENT-*ee* NON-VOL-*ee*

A.2.18 Words derived using *-er*

absorber	NEWDER	PART-IN-E-*VerN*
accelerometer	NEWSTEMDER	*PART-IN-E-*VerN*

accelerometer	NEWSTEMDER	*PART-IN-E-*VerN*
achiever	NEWDER	PART-IN-E-*VerN*
actinometer	NEWSTEMDER	*PART-IN-E-*VerN*
adapter	INLEX	PART-IN-E-*VerN*
admirer	INLEX	PART-IN-E-*VerN*
advertiser	NEWDER	PART-IN-E-*VerN*
adviser	INLEX	PART-IN-E-*VerN*
amplifier	INLEX	PART-IN-E-*VerN*
analyzer	NEWSTEMDER	PART-IN-E-*VerN*
announcer	INLEX	PART-IN-E-*VerN*
appraiser	NEWDER	PART-IN-E-*VerN*
arranger	NEWDER	PART-IN-E-*VerN*
attacker	NEWDER	PART-IN-E-*VerN*
autoloader	NEWSTEMDER	PART-IN-E-*VerN*
backer	INLEX	PART-IN-E-*VerN*
baffler	NEWDER	PART-IN-E-*VerN*
baker	INLEX	PART-IN-E-*VerN*
ballplayer	NEWSTEMDER	*PART-IN-E-*VerN*
banker	INLEX	PART-IN-E-*VerN*
barnstormer	NEWDER	PART-IN-E-*VerN*
bather	NEWDER	PART-IN-E-*VerN*
bearer	INLEX	PART-IN-E-*VerN*
believer	INLEX	PART-IN-E-*VerN*
bellwether	NEWSTEMDER	*PART-IN-E-*VerN*
besieger	NEWDER	PART-IN-E-*VerN*
betrayer	NEWDER	PART-IN-E-*VerN*
binder	INLEX	PART-IN-E-*VerN*
biter	NEWDER	PART-IN-E-*VerN*
blackmailer	NEWDER	PART-IN-E-*VerN*
blazer	INLEX	*PART-IN-E-*VerN*
bleacher	INLEX	*PART-IN-E-*VerN*
blinker	INLEX	PART-IN-E-*VerN*
blower	INLEX	PART-IN-E-*VerN*
boarder	INLEX	PART-IN-E-*VerN*
boater	INLEX	PART-IN-E-*VerN*
bodybuilder	NEWSTEMDER	PART-IN-E-*VerN*
boiler	INLEX	PART-IN-E-*VerN*
bomber	INLEX	PART-IN-E-*VerN*
booker	NEWDER	PART-IN-E-*VerN*
booster	INLEX	PART-IN-E-*VerN*
borer	INLEX	PART-IN-E-*VerN*
borrower	NEWDER	PART-IN-E-*VerN*
bower	INLEX	*PART-IN-E-*VerN*
boxer	INLEX	PART-IN-E-*VerN*
breaker	INLEX	PART-IN-E-*VerN*
brewer	INLEX	PART-IN-E-*VerN*
briber	NEWDER	PART-IN-E-*VerN*
broadcaster	NEWDER	PART-IN-E-*VerN*
broiler	INLEX	PART-IN-E-*VerN*
buffer	INLEX	*PART-IN-E-*VerN*
bugler	NEWSTEMDER	PART-IN-E-*VerN*
builder	INLEX	PART-IN-E-*VerN*
bullwhacker	NEWSTEMDER	?PART-IN-E-*VerN*
bumper	INLEX	PART-IN-E-*VerN*
bunker	INLEX	*PART-IN-E-*VerN*
bunter	NEWSTEMDER	PART-IN-E-*VerN*
burner	INLEX	PART-IN-E-*VerN*
butter	INLEX	*PART-IN-E-*VerN*
buyer	INLEX	PART-IN-E-*VerN*
caller	INLEX	PART-IN-E-*VerN*
calligrapher	NEWSTEMDER	*PART-IN-E-*VerN*
calorimeter	NEWSTEMDER	*PART-IN-E-*VerN*
campaigner	NEWDER	PART-IN-E-*VerN*
camper	INLEX	PART-IN-E-*VerN*

canter	INLEX	*PART-IN-E-*VerN*
canvasser	NEWDER	PART-IN-E-*VerN*
carreer	NEWSTEMDER	*PART-IN-E-*VerN*
carrier	INLEX	PART-IN-E-*VerN*
carver	INLEX	PART-IN-E-*VerN*
caseworker	NEWSTEMDER	*PART-IN-E-*VerN*
caster	INLEX	PART-IN-E-*VerN*
catcher	INLEX	PART-IN-E-*VerN*
challenger	NEWDER	PART-IN-E-*VerN*
chamfer	NEWSTEMDER	*PART-IN-E-*VerN*
chanter	NEWDER	PART-IN-E-*VerN*
charmer	INLEX	PART-IN-E-*VerN*
charter	INLEX	*PART-IN-E-*VerN*
checker	INLEX	PART-IN-E-*VerN*
chowder	NEWSTEMDER	*PART-IN-E-*VerN*
chronicler	NEWDER	PART-IN-E-*VerN*
classifier	NEWDER	PART-IN-E-*VerN*
cleaner	INLEX	PART-IN-E-*VerN*
clincher	INLEX	PART-IN-E-*VerN*
co-signer	NEWDER	PART-IN-E-*VerN*
co-worker	INLEX	PART-IN-E-*VerN*
comer	INLEX	*PART-IN-E-*VerN*
commander	INLEX	PART-IN-E-*VerN*
commissioner	INLEX	*PART-IN-E-*VerN*
commuter	INLEX	PART-IN-E-*VerN*
compiler	NEWDER	PART-IN-E-*VerN*
composer	INLEX	PART-IN-E-*VerN*
computer	INLEX	PART-IN-E-*VerN*
condenser	INLEX	PART-IN-E-*VerN*
conditioner	NEWDER	PART-IN-E-*VerN*
conniver	NEWDER	PART-IN-E-*VerN*
consumer	INLEX	PART-IN-E-*VerN*
container	INLEX	PART-IN-E-*VerN*
contender	INLEX	PART-IN-E-*VerN*
cooler	INLEX	PART-IN-E-*VerN*
corker	INLEX	PART-IN-E-*VerN*
corner	INLEX	*PART-IN-E-*VerN*
corrupter	INLEX	PART-IN-E-*VerN*
counter	INLEX	*PART-IN-E-*VerN*
coupler	NEWDER	PART-IN-E-*VerN*
coworker	INLEX	PART-IN-E-*VerN*
cowpuncher	NEWSTEMDER	*PART-IN-E-*VerN*
cowrtier	NEWSTEMDER	*PART-IN-E-*VerN*
cracker	INLEX	*PART-IN-E-*VerN*
crafter	NEWDER	PART-IN-E-*VerN*
crasher	INLEX	PART-IN-E-*VerN*
crater	INLEX	*PART-IN-E-*VerN*
creamer	INLEX	*PART-IN-E-*VerN*
creeper	INLEX	PART-IN-E-*VerN*
cruiser	INLEX	PART-IN-E-*VerN*
crusader	NEWDER	PART-IN-E-*VerN*
crusher	NEWDER	PART-IN-E-*VerN*
crystallographer	NEWSTEMDER	*PART-IN-E-*VerN*
cuirassier	NEWSTEMDER	*PART-IN-E-*VerN*
dabbler	NEWDER	PART-IN-E-*VerN*
dager	NEWSTEMDER	*PART-IN-E-*VerN*
dancer	NEWDER	PART-IN-E-*VerN*
dazzler	NEWDER	PART-IN-E-*VerN*
dealer	INLEX	PART-IN-E-*VerN*
defender	NEWDER	PART-IN-E-*VerN*
deluxer	NEWSTEMDER	*PART-IN-E-*VerN*
demander	NEWDER	PART-IN-E-*VerN*
designer	INLEX	PART-IN-E-*VerN*
despoiler	NEWDER	PART-IN-E-*VerN*

destroyer	INLEX	PART-IN-E-*VerN*
diagnometer	NEWSTEMDER	*PART-IN-E-*VerN*
dieter	NEWDER	PART-IN-E-*VerN*
diffuser	NEWDER	PART-IN-E-*VerN*
dimer	NEWSTEMDER	*PART-IN-E-*VerN*
disclaimer	INLEX	PART-IN-E-*VerN*
discoverer	NEWDER	PART-IN-E-*VerN*
dispenser	INLEX	PART-IN-E-*VerN*
dissenter	INLEX	PART-IN-E-*VerN*
distiller	INLEX	PART-IN-E-*VerN*
disturber	NEWDER	PART-IN-E-*VerN*
ditcher	NEWDER	*PART-IN-E-*VerN*
diver	INLEX	PART-IN-E-*VerN*
divider	NEWDER	PART-IN-E-*VerN*
docter	NEWSTEMDER	*PART-IN-E-*VerN*
doer	INLEX	PART-IN-E-*VerN*
double-crosser	NEWDER	PART-IN-E-*VerN*
doubleheader	NEWSTEMDER	?PART-IN-E-*VerN*
drafter	NEWDER	PART-IN-E-*VerN*
draper	INLEX	*PART-IN-E-*VerN*
drawer	INLEX	*PART-IN-E-*VerN*
dreamer	INLEX	PART-IN-E-*VerN*
dresser	INLEX	PART-IN-E-*VerN*
drier	INLEX	PART-IN-E-*VerN*
drinker	INLEX	PART-IN-E-*VerN*
driver	INLEX	PART-IN-E-*VerN*
ducer	NEWSTEMDER	PART-IN-E-*VerN*
dweller	NEWDER	PART-IN-E-*VerN*
easterner	NEWSTEMDER	*PART-IN-E-*VerN*
eater	INLEX	PART-IN-E-*VerN*
employer	INLEX	PART-IN-E-*VerN*
enforcer	NEWDER	PART-IN-E-*VerN*
engraver	NEWDER	PART-IN-E-*VerN*
enjoinder	NEWSTEMDER	*PART-IN-E-*VerN*
entertainer	INLEX	PART-IN-E-*VerN*
equalizer	NEWDER	PART-IN-E-*VerN*
eraser	INLEX	PART-IN-E-*VerN*
ester	NEWSTEMDER	*PART-IN-E-*VerN*
ether	NEWSTEMDER	*PART-IN-E-*VerN*
eulogizer	NEWDER	PART-IN-E-*VerN*
ex-schoolteacher	NEWSTEMDER	*PART-IN-E-*VerN*
examiner	NEWDER	PART-IN-E-*VerN*
experimenter	NEWDER	PART-IN-E-*VerN*
exploiter	NEWDER	PART-IN-E-*VerN*
explorer	NEWDER	PART-IN-E-*VerN*
exporter	INLEX	PART-IN-E-*VerN*
extruder	NEWDER	PART-IN-E-*VerN*
fairgoer	NEWSTEMDER	*PART-IN-E-*VerN*
faker	INLEX	PART-IN-E-*VerN*
farmer	INLEX	PART-IN-E-*VerN*
favorer	NEWDER	PART-IN-E-*VerN*
feeder	INLEX	PART-IN-E-*VerN*
feeler	INLEX	PART-IN-E-*VerN*
feler	NEWSTEMDER	*PART-IN-E-*VerN*
feller	INLEX	*PART-IN-E-*VerN*
fender	INLEX	*PART-IN-E-*VerN*
ferometer	NEWSTEMDER	*PART-IN-E-*VerN*
fertilizer	INLEX	PART-IN-E-*VerN*
fielder	INLEX	PART-IN-E-*VerN*
fighter	INLEX	PART-IN-E-*VerN*
filler	INLEX	PART-IN-E-*VerN*
finder	INLEX	PART-IN-E-*VerN*
finisher	NEWDER	PART-IN-E-*VerN*
fisher	NEWDER	PART-IN-E-*VerN*

fixer	NEWDER	PART-IN-E-*VerN*
flier	INLEX	PART-IN-E-*VerN*
floater	NEWDER	PART-IN-E-*VerN*
flower	INLEX	*PART-IN-E-*VerN*
folder	INLEX	*PART-IN-E-*VerN*
follower	INLEX	PART-IN-E-*VerN*
forecaster	NEWDER	PART-IN-E-*VerN*
founder	INLEX	PART-IN-E-*VerN*
framer	NEWDER	PART-IN-E-*VerN*
freewheeler	NEWDER	PART-IN-E-*VerN*
freezer	INLEX	PART-IN-E-*VerN*
freighter	INLEX	PART-IN-E-*VerN*
gagwriter	NEWSTEMDER	*PART-IN-E-*VerN*
gainer	NEWDER	PART-IN-E-*VerN*
gallbladder	NEWSTEMDER	*PART-IN-E-*VerN*
gambler	NEWDER	PART-IN-E-*VerN*
gardener	NEWDER	PART-IN-E-*VerN*
gazer	NEWDER	PART-IN-E-*VerN*
girder	INLEX	*PART-IN-E-*VerN*
giver	NEWDER	PART-IN-E-*VerN*
glander	NEWSTEMDER	*PART-IN-E-*VerN*
glider	INLEX	PART-IN-E-*VerN*
glover	NEWSTEMDER	*PART-IN-E-*VerN*
gobbler	INLEX	PART-IN-E-*VerN*
golfer	NEWDER	PART-IN-E-*VerN*
grader	NEWDER	PART-IN-E-*VerN*
grasser	NEWDER	*PART-IN-E-*VerN*
grazer	NEWDER	PART-IN-E-*VerN*
grinder	INLEX	PART-IN-E-*VerN*
grounder	NEWDER	*PART-IN-E-*VerN*
grower	INLEX	PART-IN-E-*VerN*
gunfighter	NEWSTEMDER	*PART-IN-E-*VerN*
gunslinger	NEWSTEMDER	*PART-IN-E-*VerN*
gusher	INLEX	PART-IN-E-*VerN*
hacker	NEWDER	PART-IN-E-*VerN*
halter	INLEX	*PART-IN-E-*VerN*
hander	NEWDER	*PART-IN-E-*VerN*
handler	INLEX	PART-IN-E-*VerN*
hanger	INLEX	PART-IN-E-*VerN*
hardener	NEWDER	PART-IN-E-*VerN*
hasher	NEWDER	*PART-IN-E-*VerN*
hawker	INLEX	PART-IN-E-*VerN*
header	INLEX	*PART-IN-E-*VerN*
headquarter	NEWSTEMDER	*PART-IN-E-*VerN*
headwater	NEWSTEMDER	*PART-IN-E-*VerN*
healer	NEWDER	PART-IN-E-*VerN*
hearer	INLEX	PART-IN-E-*VerN*
heater	INLEX	PART-IN-E-*VerN*
heaver	NEWDER	?PART-IN-E-*VerN*
heeler	NEWDER	PART-IN-E-*VerN*
helper	NEWDER	PART-IN-E-*VerN*
hesiometer	NEWSTEMDER	*PART-IN-E-*VerN*
hijacker	NEWDER	PART-IN-E-*VerN*
holder	INLEX	PART-IN-E-*VerN*
homebuilder	NEWSTEMDER	*PART-IN-E-*VerN*
homemaker	NEWSTEMDER	*PART-IN-E-*VerN*
homeowner	NEWSTEMDER	*PART-IN-E-*VerN*
homer	INLEX	*PART-IN-E-*VerN*
homesteader	NEWSTEMDER	PART-IN-E-*VerN*
homopolymer	NEWSTEMDER	*PART-IN-E-*VerN*
honeymooner	NEWDER	PART-IN-E-*VerN*
hunter	INLEX	PART-IN-E-*VerN*
hurler	NEWDER	PART-IN-E-*VerN*
hustler	INLEX	PART-IN-E-*VerN*

idler	INLEX	PART-IN-E-*VerN*
impresser	NEWDER	PART-IN-E-*VerN*
improviser	NEWDER	PART-IN-E-*VerN*
intensifier	INLEX	PART-IN-E-*VerN*
interferometer	NEWSTEMDER	*PART-IN-E-*VerN*
interviewer	NEWDER	PART-IN-E-*VerN*
intruder	INLEX	PART-IN-E-*VerN*
invader	NEWDER	PART-IN-E-*VerN*
isomer	NEWSTEMDER	*PART-IN-E-*VerN*
jetliner	NEWSTEMDER	*PART-IN-E-*VerN*
joiner	INLEX	PART-IN-E-*VerN*
joker	INLEX	PART-IN-E-*VerN*
jumper	INLEX	PART-IN-E-*VerN*
keeper	INLEX	PART-IN-E-*VerN*
kidnaper	NEWSTEMDER	PART-IN-E-*VerN*
killer	INLEX	PART-IN-E-*VerN*
laborer	INLEX	PART-IN-E-*VerN*
landowner	NEWSTEMDER	*PART-IN-E-*VerN*
larder	INLEX	*PART-IN-E-*VerN*
launcher	NEWDER	PART-IN-E-*VerN*
lawmaker	NEWSTEMDER	*PART-IN-E-*VerN*
layer	INLEX	*PART-IN-E-*VerN*
leader	INLEX	PART-IN-E-*VerN*
leafhopper	NEWSTEMDER	*PART-IN-E-*VerN*
leaguer	NEWDER	*PART-IN-E-*VerN*
learner	INLEX	PART-IN-E-*VerN*
lecher	INLEX	*PART-IN-E-*VerN*
lecturer	INLEX	PART-IN-E-*VerN*
lifter	NEWDER	PART-IN-E-*VerN*
lighter	INLEX	PART-IN-E-*VerN*
linebacker	NEWSTEMDER	*PART-IN-E-*VerN*
liner	INLEX	*PART-IN-E-*VerN*
listener	INLEX	PART-IN-E-*VerN*
liver	INLEX	*PART-IN-E-*VerN*
loader	NEWDER	PART-IN-E-*VerN*
locker	INLEX	PART-IN-E-*VerN*
loser	INLEX	PART-IN-E-*VerN*
lover	INLEX	PART-IN-E-*VerN*
lower	INLEX	*PART-IN-E-*VerN*
luster	INLEX	*PART-IN-E-*VerN*
maker	INLEX	PART-IN-E-*VerN*
manager	INLEX	PART-IN-E-*VerN*
manufacturer	INLEX	PART-IN-E-*VerN*
marauder	NEWDER	PART-IN-E-*VerN*
marker	INLEX	PART-IN-E-*VerN*
mauler	NEWDER	PART-IN-E-*VerN*
merger	INLEX	PART-IN-E-*VerN*
midwesterner	NEWSTEMDER	*PART-IN-E-*VerN*
millivoltmeter	NEWSTEMDER	*PART-IN-E-*VerN*
miner	INLEX	PART-IN-E-*VerN*
minter	NEWDER	PART-IN-E-*VerN*
misinterpreter	NEWSTEMDER	PART-IN-E-*VerN*
misunderstander	NEWDER	PART-IN-E-*VerN*
mixer	INLEX	PART-IN-E-*VerN*
modifier	INLEX	PART-IN-E-*VerN*
monomer	NEWSTEMDER	*PART-IN-E-*VerN*
mooncurser	NEWSTEMDER	*PART-IN-E-*VerN*
motorscooter	NEWSTEMDER	*PART-IN-E-*VerN*
mourner	INLEX	PART-IN-E-*VerN*
mover	INLEX	PART-IN-E-*VerN*
mucker	NEWDER	*PART-IN-E-*VerN*
muffler	INLEX	PART-IN-E-*VerN*
murderer	NEWDER	PART-IN-E-*VerN*
mutterer	NEWDER	PART-IN-E-*VerN*

nester	NEWDER	PART-IN-E-*VerN*
nibbler	NEWDER	PART-IN-E-*VerN*
nighter	NEWSTEMDER	*PART-IN-E-*VerN*
noisemaker	NEWSTEMDER	*PART-IN-E-*VerN*
northener	NEWSTEMDER	*PART-IN-E-*VerN*
norther	NEWSTEMDER	*PART-IN-E-*VerN*
northerner	NEWSTEMDER	*PART-IN-E-*VerN*
nullifier	NEWDER	PART-IN-E-*VerN*
number	INLEX	*PART-IN-E-*VerN*
observer	INLEX	PART-IN-E-*VerN*
offender	INLEX	PART-IN-E-*VerN*
opener	INLEX	PART-IN-E-*VerN*
operagoer	NEWSTEMDER	*PART-IN-E-*VerN*
organizer	NEWDER	PART-IN-E-*VerN*
other	NEWSTEMDER	*PART-IN-E-*VerN*
ouster	NEWDER	PART-IN-E-*VerN*
outfielder	NEWDER	*PART-IN-E-*VerN*
outlander	NEWDER	*PART-IN-E-*VerN*
outsider	INLEX	*PART-IN-E-*VerN*
over-achiever	NEWDER	PART-IN-E-*VerN*
overnighter	NEWSTEMDER	*PART-IN-E-*VerN*
overseer	INLEX	PART-IN-E-*VerN*
owner	INLEX	PART-IN-E-*VerN*
oystcher	NEWSTEMDER	*PART-IN-E-*VerN*
pacer	NEWDER	PART-IN-E-*VerN*
pacifier	INLEX	PART-IN-E-*VerN*
painter	INLEX	PART-IN-E-*VerN*
parameter	NEWSTEMDER	*PART-IN-E-*VerN*
parameter	NEWSTEMDER	*PART-IN-E-*VerN*
partaker	NEWDER	PART-IN-E-*VerN*
pateroller	NEWSTEMDER	*PART-IN-E-*VerN*
peddler	INLEX	PART-IN-E-*VerN*
peer	INLEX	*PART-IN-E-*VerN*
pensioner	INLEX	*PART-IN-E-*VerN*
performer	INLEX	PART-IN-E-*VerN*
persuader	NEWDER	PART-IN-E-*VerN*
petitioner	INLEX	PART-IN-E-*VerN*
photographer	INLEX	PART-IN-E-*VerN*
picker	INLEX	PART-IN-E-*VerN*
piper	INLEX	PART-IN-E-*VerN*
pistoleer	NEWSTEMDER	*PART-IN-E-*VerN*
pitcher	INLEX	PART-IN-E-*VerN*
planer	INLEX	PART-IN-E-*VerN*
planter	INLEX	PART-IN-E-*VerN*
plasterer	INLEX	PART-IN-E-*VerN*
player	INLEX	PART-IN-E-*VerN*
pleader	NEWDER	PART-IN-E-*VerN*
plier	INLEX	*PART-IN-E-*VerN*
plumber	INLEX	*PART-IN-E-*VerN*
plunderer	NEWDER	PART-IN-E-*VerN*
plunker	NEWDER	?PART-IN-E-*VerN*
pointer	INLEX	PART-IN-E-*VerN*
poker	INLEX	PART-IN-E-*VerN*
polyester	NEWSTEMDER	*PART-IN-E-*VerN*
polyether	NEWSTEMDER	*PART-IN-E-*VerN*
polyether	NEWSTEMDER	*PART-IN-E-*VerN*
pornographer	NEWSTEMDER	*PART-IN-E-*VerN*
porter	INLEX	*PART-IN-E-*VerN*
portwatcher	NEWSTEMDER	*PART-IN-E-*VerN*
poster	INLEX	PART-IN-E-*VerN*
potentiometer	NEWSTEMDER	*PART-IN-E-*VerN*
prayer	INLEX	PART-IN-E-*VerN*
preacher	NEWDER	PART-IN-E-*VerN*
prepolymer	NEWSTEMDER	*PART-IN-E-*VerN*

presenter	INLEX	PART-IN-E-*VerN*
presser	NEWDER	PART-IN-E-*VerN*
pretender	INLEX	PART-IN-E-*VerN*
primer	INLEX	PART-IN-E-*VerN*
printer	INLEX	PART-IN-E-*VerN*
procurer	INLEX	PART-IN-E-*VerN*
producer	INLEX	PART-IN-E-*VerN*
programmer	INLEX	PART-IN-E-*VerN*
promoter	INLEX	PART-IN-E-*VerN*
proprieter	NEWSTEMDER	*PART-IN-E-*VerN*
prowler	INLEX	PART-IN-E-*VerN*
publisher	INLEX	PART-IN-E-*VerN*
puncher	NEWDER	PART-IN-E-*VerN*
purchaser	INLEX	PART-IN-E-*VerN*
pursuer	INLEX	PART-IN-E-*VerN*
pusher	INLEX	PART-IN-E-*VerN*
putter	INLEX	PART-IN-E-*VerN*
puzzler	INLEX	PART-IN-E-*VerN*
pyrometer	NEWSTEMDER	*PART-IN-E-*VerN*
pyrometer	NEWSTEMDER	*PART-IN-E-*VerN*
questioner	INLEX	PART-IN-E-*VerN*
racer	INLEX	PART-IN-E-*VerN*
rafter	INLEX	PART-IN-E-*VerN*
raider	INLEX	PART-IN-E-*VerN*
railroader	NEWDER	PART-IN-E-*VerN*
raiser	NEWDER	PART-IN-E-*VerN*
ranger	INLEX	PART-IN-E-*VerN*
rattler	NEWDER	PART-IN-E-*VerN*
reader	INLEX	PART-IN-E-*VerN*
receiver	INLEX	PART-IN-E-*VerN*
recorder	INLEX	PART-IN-E-*VerN*
recruiter	NEWDER	PART-IN-E-*VerN*
rectifier	INLEX	PART-IN-E-*VerN*
redeveloper	NEWSTEMDER	PART-IN-E-*VerN*
redheader	NEWSTEMDER	*PART-IN-E-*VerN*
reducer	NEWDER	PART-IN-E-*VerN*
reformer	NEWDER	PART-IN-E-*VerN*
refresher	INLEX	PART-IN-E-*VerN*
reminder	INLEX	PART-IN-E-*VerN*
repeater	INLEX	PART-IN-E-*VerN*
reporter	INLEX	PART-IN-E-*VerN*
requester	NEWDER	PART-IN-E-*VerN*
researcher	NEWDER	PART-IN-E-*VerN*
restorer	INLEX	PART-IN-E-*VerN*
retailer	INLEX	PART-IN-E-*VerN*
retainer	INLEX	PART-IN-E-*VerN*
retriever	INLEX	PART-IN-E-*VerN*
revenuer	NEWSTEMDER	*PART-IN-E-*VerN*
reviewer	INLEX	PART-IN-E-*VerN*
revolver	INLEX	*PART-IN-E-*VerN*
rider	INLEX	PART-IN-E-*VerN*
ringer	INLEX	*PART-IN-E-*VerN*
ringsider	NEWSTEMDER	*PART-IN-E-*VerN*
rioter	NEWDER	PART-IN-E-*VerN*
river	INLEX	*PART-IN-E-*VerN*
rocker	INLEX	PART-IN-E-*VerN*
rodder	NEWSTEMDER	*PART-IN-E-*VerN*
roemer	NEWSTEMDER	?PART-IN-E-*VerN*
roller	INLEX	PART-IN-E-*VerN*
romancer	NEWDER	PART-IN-E-*VerN*
roofer	NEWDER	PART-IN-E-*VerN*
rooster	INLEX	*PART-IN-E-*VerN*
roper	NEWDER	PART-IN-E-*VerN*
rover	INLEX	*PART-IN-E-*VerN*

ruler	INLEX	PART-IN-E-*VerN*
rustler	INLEX	PART-IN-E-*VerN*
sacker	NEWDER	PART-IN-E-*VerN*
saloonkeeper	NEWSTEMDER	*PART-IN-E-*VerN*
sampler	INLEX	PART-IN-E-*VerN*
sander	INLEX	PART-IN-E-*VerN*
saucer	INLEX	*PART-IN-E-*VerN*
saver	INLEX	PART-IN-E-*VerN*
sawtimber	NEWSTEMDER	*PART-IN-E-*VerN*
scavenger	INLEX	PART-IN-E-*VerN*
schooler	NEWDER	PART-IN-E-*VerN*
schoolteacher	NEWSTEMDER	PART-IN-E-*VerN*
scorcher	INLEX	PART-IN-E-*VerN*
scriber	INLEX	PART-IN-E-*VerN*
seafarer	NEWSTEMDER	*PART-IN-E-*VerN*
seducer	NEWDER	PART-IN-E-*VerN*
seeker	NEWDER	PART-IN-E-*VerN*
seer	INLEX	*PART-IN-E-*VerN*
seller	INLEX	PART-IN-E-*VerN*
sender	INLEX	PART-IN-E-*VerN*
settler	INLEX	PART-IN-E-*VerN*
sewer	INLEX	*PART-IN-E-*VerN*
shaker	INLEX	PART-IN-E-*VerN*
sharer	NEWDER	PART-IN-E-*VerN*
shifter	NEWDER	PART-IN-E-*VerN*
shocker	INLEX	PART-IN-E-*VerN*
shooter	NEWDER	PART-IN-E-*VerN*
shouder	NEWSTEMDER	*PART-IN-E-*VerN*
shower	INLEX	*PART-IN-E-*VerN*
sidewinder	NEWSTEMDER	*PART-IN-E-*VerN*
sightseer	INLEX	PART-IN-E-*VerN*
signer	INLEX	PART-IN-E-*VerN*
singer	NEWDER	PART-IN-E-*VerN*
skirmisher	NEWDER	PART-IN-E-*VerN*
skyjacker	NEWDER	PART-IN-E-*VerN*
slanderer	NEWDER	PART-IN-E-*VerN*
sleeper	INLEX	PART-IN-E-*VerN*
slicker	INLEX	*PART-IN-E-*VerN*
smoker	INLEX	PART-IN-E-*VerN*
smuggler	NEWDER	PART-IN-E-*VerN*
sneaker	INLEX	*PART-IN-E-*VerN*
sniper	NEWDER	*PART-IN-E-*VerN*
snuffer	INLEX	PART-IN-E-*VerN*
softener	NEWDER	PART-IN-E-*VerN*
sojourner	NEWDER	PART-IN-E-*VerN*
sommelier	NEWSTEMDER	*PART-IN-E-*VerN*
southerner	NEWSTEMDER	*PART-IN-E-*VerN*
soyaburger	NEWSTEMDER	*PART-IN-E-*VerN*
spacer	NEWDER	PART-IN-E-*VerN*
speaker	INLEX	PART-IN-E-*VerN*
spectrometer	NEWSTEMDER	*PART-IN-E-*VerN*
spectrophotometer	NEWSTEMDER	*PART-IN-E-*VerN*
spender	INLEX	PART-IN-E-*VerN*
spoiler	NEWDER	PART-IN-E-*VerN*
sportswriter	NEWSTEMDER	*PART-IN-E-*VerN*
spreader	NEWDER	PART-IN-E-*VerN*
stabilizer	INLEX	PART-IN-E-*VerN*
stager	INLEX	PART-IN-E-*VerN*
starter	INLEX	PART-IN-E-*VerN*
stealer	NEWDER	PART-IN-E-*VerN*
steamer	INLEX	PART-IN-E-*VerN*
steelmaker	NEWSTEMDER	*PART-IN-E-*VerN*
stepmother	NEWSTEMDER	*PART-IN-E-*VerN*
stinkpotter	NEWSTEMDER	*PART-IN-E-*VerN*

stoker	NEWDER	PART-IN-E-*VerN*
straggler	NEWDER	PART-IN-E-*VerN*
streamer	INLEX	*PART-IN-E-*VerN*
streamliner	NEWDER	*PART-IN-E-*VerN*
stretcher	INLEX	PART-IN-E-*VerN*
subscriber	INLEX	PART-IN-E-*VerN*
sucker	INLEX	*PART-IN-E-*VerN*
sufferer	INLEX	PART-IN-E-*VerN*
supplier	INLEX	PART-IN-E-*VerN*
supporter	INLEX	PART-IN-E-*VerN*
suspender	INLEX	PART-IN-E-*VerN*
sweater	INLEX	*PART-IN-E-*VerN*
synchronizer	NEWDER	PART-IN-E-*VerN*
taker	INLEX	PART-IN-E-*VerN*
talker	INLEX	PART-IN-E-*VerN*
taper	INLEX	*PART-IN-E-*VerN*
taxpayer	NEWSTEMDER	*PART-IN-E-*VerN*
teacher	INLEX	PART-IN-E-*VerN*
telegrapher	INLEX	PART-IN-E-*VerN*
teller	INLEX	PART-IN-E-*VerN*
tempter	NEWDER	PART-IN-E-*VerN*
theatergoer	NEWSTEMDER	*PART-IN-E-*VerN*
thickener	INLEX	PART-IN-E-*VerN*
thinker	NEWDER	PART-IN-E-*VerN*
thriller	INLEX	PART-IN-E-*VerN*
thrower	NEWDER	PART-IN-E-*VerN*
tier	INLEX	*PART-IN-E-*VerN*
tiller	INLEX	PART-IN-E-*VerN*
timer	INLEX	PART-IN-E-*VerN*
titer	NEWSTEMDER	*PART-IN-E-*VerN*
toddler	INLEX	PART-IN-E-*VerN*
toner	NEWDER	PART-IN-E-*VerN*
toolmaker	NEWSTEMDER	*PART-IN-E-*VerN*
tormenter	NEWDER	PART-IN-E-*VerN*
torquer	NEWSTEMDER	PART-IN-E-*VerN*
tower	INLEX	*PART-IN-E-*VerN*
tracer	INLEX	PART-IN-E-*VerN*
trader	INLEX	PART-IN-E-*VerN*
trailer	INLEX	PART-IN-E-*VerN*
tranquilizer	NEWDER	PART-IN-E-*VerN*
transducer	NEWSTEMDER	PART-IN-E-*VerN*
transformer	INLEX	PART-IN-E-*VerN*
traveler	INLEX	PART-IN-E-*VerN*
trawler	INLEX	PART-IN-E-*VerN*
treasurer	INLEX	*PART-IN-E-*VerN*
trooper	INLEX	*PART-IN-E-*VerN*
truckdriver	NEWSTEMDER	*PART-IN-E-*VerN*
trucker	NEWSTEMDER	PART-IN-E-*VerN*
trumpeter	NEWSTEMDER	PART-IN-E-*VerN*
tumbler	INLEX	PART-IN-E-*VerN*
twirler	NEWDER	PART-IN-E-*VerN*
twister	INLEX	PART-IN-E-*VerN*
under-achiever	NEWDER	PART-IN-E-*VerN*
underachiever	NEWDER	PART-IN-E-*VerN*
undertaker	INLEX	*PART-IN-E-*VerN*
underwriter	INLEX	PART-IN-E-*VerN*
upholder	NEWDER	PART-IN-E-*VerN*
user	INLEX	PART-IN-E-*VerN*
vacationer	NEWDER	PART-IN-E-*VerN*
vernier	NEWSTEMDER	*PART-IN-E-*VerN*
viewer	INLEX	PART-IN-E-*VerN*
viscometer	NEWSTEMDER	*PART-IN-E-*VerN*
voltmeter	NEWSTEMDER	*PART-IN-E-*VerN*
voter	INLEX	PART-IN-E-*VerN*

voucher	INLEX	PART-IN-E-*VerN*
voyager	INLEX	PART-IN-E-*VerN*
wager	INLEX	*PART-IN-E-*VerN*
waiter	INLEX	PART-IN-E-*VerN*
walker	INLEX	PART-IN-E-*VerN*
wanderer	INLEX	PART-IN-E-*VerN*
washer	INLEX	PART-IN-E-*VerN*
wastewater	NEWSTEMDER	*PART-IN-E-*VerN*
watcher	NEWDER	PART-IN-E-*VerN*
waver	INLEX	PART-IN-E-*VerN*
westerner	NEWSTEMDER	*PART-IN-E-*VerN*
westerner	NEWSTEMDER	*PART-IN-E-*VerN*
whisker	INLEX	*PART-IN-E-*VerN*
wielder	NEWDER	PART-IN-E-*VerN*
wiener	NEWSTEMDER	*PART-IN-E-*VerN*
wigmaker	NEWSTEMDER	*PART-IN-E-*VerN*
winder	NEWDER	PART-IN-E-*VerN*
wisenheimer	NEWSTEMDER	*PART-IN-E-*VerN*
woodcarver	NEWSTEMDER	*PART-IN-E-*VerN*
woolworker	NEWSTEMDER	*PART-IN-E-*VerN*
worker	INLEX	PART-IN-E-*VerN*
worlder	NEWSTEMDER	*PART-IN-E-*VerN*
wpuncher	NEWSTEMDER	*PART-IN-E-*VerN*
wrangler	INLEX	PART-IN-E-*VerN*
writer	INLEX	PART-IN-E-*VerN*
wrongdoer	NEWSTEMDER	*PART-IN-E-*VerN*
wrtier	NEWSTEMDER	*PART-IN-E-*VerN*
yachter	NEWSTEMDER	PART-IN-E-*VerN*

A.2.19 Words derived using *-ant*

accountant	INLEX	PART-IN-E-*ant*
aspirant	INLEX	PART-IN-E-*ant*
assailant	INLEX	PART-IN-E-*ant*
assistant	INLEX	PART-IN-E-*ant*
attendant	INLEX	PART-IN-E-*ant*
claimant	INLEX	PART-IN-E-*ant*
commandant	INLEX	PART-IN-E-*ant*
consultant	INLEX	PART-IN-E-*ant*
contestant	INLEX	PART-IN-E-*ant*
coolant	INLEX	PART-IN-E-*ant*
defendant	INLEX	PART-IN-E-*ant*
depressant	NEWDER	PART-IN-E-*ant*
descendant	INLEX	PART-IN-E-*ant*
determinant	INLEX	?PART-IN-E-*ant*
discussant	NEWDER	PART-IN-E-*ant*
habitant	NEWSTEMDER	*PART-IN-E-*ant*
informant	INLEX	PART-IN-E-*ant*
intendant	NEWDER	*PART-IN-E-*ant*
ironpant	NEWSTEMDER	*PART-IN-E-*ant*
malfeasant	NEWSTEMDER	*PART-IN-E-*ant*
misdemeanant	NEWDER	*PART-IN-E-*ant*
pageant	INLEX	*PART-IN-E-*ant*
pleasant	INLEX	*PART-IN-E-*ant*
powerplant	NEWSTEMDER	*PART-IN-E-*ant*
resultant	INLEX	PART-IN-E-*ant*
servant	INLEX	PART-IN-E-*ant*
significant	NEWSTEMDER	*PART-IN-E-*ant*
variant	INLEX	?PART-IN-E-*ant*

A.2.20 Words derived using *-ment*

abandonment	NEWDER	E-OR-P-OR-R-*ment*
abasement	NEWDER	E-OR-P-OR-R-*ment*
abutment	INLEX	E-OR-P-OR-R-*ment*
accompaniment	INLEX	E-OR-P-OR-R-*ment*
accomplishment	INLEX	E-OR-P-OR-R-*ment*
accouterment	NEWSTEMDER	E-OR-P-OR-R-*ment*
achievement	INLEX	E-OR-P-OR-R-*ment*
acknowledgement	NEWDER	E-OR-P-OR-R-*ment*
adjournment	NEWDER	E-OR-P-OR-R-*ment*
adjustment	NEWDER	E-OR-P-OR-R-*ment*
admonishment	NEWDER	E-OR-P-OR-R-*ment*
advancement	INLEX	E-OR-P-OR-R-*ment*
advertisement	INLEX	E-OR-P-OR-R-*ment*
advisement	NEWDER	E-OR-P-OR-R-*ment*
agreement	INLEX	E-OR-P-OR-R-*ment*
ailment	INLEX	E-OR-P-OR-R-*ment*
alignment	INLEX	E-OR-P-OR-R-*ment*
allotment	INLEX	E-OR-P-OR-R-*ment*
allurement	INLEX	E-OR-P-OR-R-*ment*
amazement	NEWDER	E-OR-P-OR-R-*ment*
amendment	INLEX	E-OR-P-OR-R-*ment*
amusement	INLEX	E-OR-P-OR-R-*ment*
announcement	INLEX	E-OR-P-OR-R-*ment*
appeasement	INLEX	E-OR-P-OR-R-*ment*
appointment	INLEX	E-OR-P-OR-R-*ment*
apportionment	NEWDER	E-OR-P-OR-R-*ment*
arrangement	INLEX	E-OR-P-OR-R-*ment*
assesment	NEWSTEMDER	E-OR-P-OR-R-*ment*
assessment	INLEX	E-OR-P-OR-R-*ment*
assignment	INLEX	E-OR-P-OR-R-*ment*
assortment	INLEX	E-OR-P-OR-R-*ment*
astonishment	INLEX	E-OR-P-OR-R-*ment*
atonement	NEWDER	E-OR-P-OR-R-*ment*
attachment	INLEX	E-OR-P-OR-R-*ment*
attainment	INLEX	E-OR-P-OR-R-*ment*
banishment	NEWDER	E-OR-P-OR-R-*ment*
basement	INLEX	*E-OR-P-OR-R-*ment*
battlement	INLEX	*E-OR-P-OR-R-*ment*
bedazzlement	NEWSTEMDER	E-OR-P-OR-R-*ment*
bereavement	INLEX	E-OR-P-OR-R-*ment*
betterment	INLEX	E-OR-P-OR-R-*ment*
bewilderment	NEWDER	E-OR-P-OR-R-*ment*
bombardment	INLEX	E-OR-P-OR-R-*ment*
chastisement	INLEX	E-OR-P-OR-R-*ment*
citement	INLEX	*E-OR-P-OR-R-*ment*
commandment	INLEX	E-OR-P-OR-R-*ment*
commencement	INLEX	*E-OR-P-OR-R-*ment*
commitment	INLEX	E-OR-P-OR-R-*ment*
committment	NEWSTEMDER	E-OR-P-OR-R-*ment*
compliment	INLEX	*E-OR-P-OR-R-*ment*
comportment	INLEX	E-OR-P-OR-R-*ment*
concealment	INLEX	E-OR-P-OR-R-*ment*
confinement	INLEX	E-OR-P-OR-R-*ment*
containment	INLEX	E-OR-P-OR-R-*ment*
contentment	INLEX	E-OR-P-OR-R-*ment*
curement	NEWDER	*E-OR-P-OR-R-*ment*
decrement	NEWSTEMDER	*E-OR-P-OR-R-*ment*
deferment	NEWDER	E-OR-P-OR-R-*ment*
delineament	NEWSTEMDER	E-OR-P-OR-R-*ment*
department	INLEX	*E-OR-P-OR-R-*ment*
deployment	NEWDER	E-OR-P-OR-R-*ment*
derangement	NEWDER	E-OR-P-OR-R-*ment*

detachment	INLEX	E-OR-P-OR-R-*ment*
detriment	INLEX	*E-OR-P-OR-R-*ment*
development	INLEX	E-OR-P-OR-R-*ment*
disagreement	INLEX	E-OR-P-OR-R-*ment*
disappointment	INLEX	E-OR-P-OR-R-*ment*
disbursement	INLEX	E-OR-P-OR-R-*ment*
discernment	INLEX	E-OR-P-OR-R-*ment*
discouragement	INLEX	E-OR-P-OR-R-*ment*
disenfranchisement	NEWDER	E-OR-P-OR-R-*ment*
disengagement	NEWDER	E-OR-P-OR-R-*ment*
disillusionment	INLEX	E-OR-P-OR-R-*ment*
dismemberment	NEWDER	E-OR-P-OR-R-*ment*
disparagement	NEWDER	E-OR-P-OR-R-*ment*
dispersement	NEWDER	E-OR-P-OR-R-*ment*
displacement	INLEX	E-OR-P-OR-R-*ment*
downpayment	NEWSTEMDER	E-OR-P-OR-R-*ment*
easement	NEWDER	*E-OR-P-OR-R-*ment*
embarrassment	INLEX	E-OR-P-OR-R-*ment*
embezzlement	NEWDER	E-OR-P-OR-R-*ment*
embodiment	INLEX	E-OR-P-OR-R-*ment*
employment	INLEX	E-OR-P-OR-R-*ment*
enactment	INLEX	E-OR-P-OR-R-*ment*
encampment	INLEX	E-OR-P-OR-R-*ment*
enchantment	INLEX	E-OR-P-OR-R-*ment*
encouragement	INLEX	E-OR-P-OR-R-*ment*
encroachment	INLEX	E-OR-P-OR-R-*ment*
endearment	INLEX	E-OR-P-OR-R-*ment*
endorsement	NEWDER	E-OR-P-OR-R-*ment*
endowment	INLEX	E-OR-P-OR-R-*ment*
enforcement	NEWDER	E-OR-P-OR-R-*ment*
engagement	INLEX	*E-OR-P-OR-R-*ment*
enjoyment	INLEX	E-OR-P-OR-R-*ment*
enlargement	INLEX	E-OR-P-OR-R-*ment*
enlightenment	INLEX	E-OR-P-OR-R-*ment*
enlistment	NEWDER	E-OR-P-OR-R-*ment*
enrichment	NEWDER	E-OR-P-OR-R-*ment*
enrollment	NEWDER	E-OR-P-OR-R-*ment*
enslavement	NEWDER	E-OR-P-OR-R-*ment*
entanglement	INLEX	E-OR-P-OR-R-*ment*
entertainment	INLEX	E-OR-P-OR-R-*ment*
enticement	INLEX	E-OR-P-OR-R-*ment*
equipment	INLEX	E-OR-P-OR-R-*ment*
establishment	INLEX	E-OR-P-OR-R-*ment*
estrangement	INLEX	E-OR-P-OR-R-*ment*
excitement	INLEX	E-OR-P-OR-R-*ment*
fulfillment	NEWDER	E-OR-P-OR-R-*ment*
government	INLEX	*E-OR-P-OR-R-*ment*
harrassment	NEWSTEMDER	E-OR-P-OR-R-*ment*
impairment	NEWDER	E-OR-P-OR-R-*ment*
impoundment	NEWDER	E-OR-P-OR-R-*ment*
imprisonment	NEWDER	E-OR-P-OR-R-*ment*
improvement	INLEX	E-OR-P-OR-R-*ment*
incitement	NEWDER	E-OR-P-OR-R-*ment*
indictment	NEWDER	E-OR-P-OR-R-*ment*
inducement	INLEX	E-OR-P-OR-R-*ment*
infringement	NEWDER	E-OR-P-OR-R-*ment*
installment	NEWDER	*E-OR-P-OR-R-*ment*
interment	INLEX	E-OR-P-OR-R-*ment*
investment	INLEX	E-OR-P-OR-R-*ment*
involvement	NEWDER	E-OR-P-OR-R-*ment*
judgement	INLEX	E-OR-P-OR-R-*ment*
maladjustment	NEWSTEMDER	E-OR-P-OR-R-*ment*
maladjustment	NEWSTEMDER	E-OR-P-OR-R-*ment*
management	INLEX	E-OR-P-OR-R-*ment*

management	NEWDER	E-OR-P-OR-R-*ment*
measurement	INLEX	E-OR-P-OR-R-*ment*
misalignment	NEWDER	E-OR-P-OR-R-*ment*
misplacement	NEWDER	E-OR-P-OR-R-*ment*
mmittment	NEWSTEMDER	*E-OR-P-OR-R-*ment*
movement	INLEX	E-OR-P-OR-R-*ment*
nourishment	INLEX	E-OR-P-OR-R-*ment*
over-achievement	NEWDER	E-OR-P-OR-R-*ment*
overpayment	NEWDER	E-OR-P-OR-R-*ment*
parchment	INLEX	*E-OR-P-OR-R-*ment*
pavement	INLEX	E-OR-P-OR-R-*ment*
payment	INLEX	E-OR-P-OR-R-*ment*
pigment	INLEX	*E-OR-P-OR-R-*ment*
placement	INLEX	E-OR-P-OR-R-*ment*
postponement	NEWDER	E-OR-P-OR-R-*ment*
preferment	INLEX	*E-OR-P-OR-R-*ment*
preordainment	NEWDER	E-OR-P-OR-R-*ment*
prepayment	NEWDER	E-OR-P-OR-R-*ment*
presentment	NEWDER	*E-OR-P-OR-R-*ment*
procurement	NEWDER	E-OR-P-OR-R-*ment*
pronouncement	INLEX	E-OR-P-OR-R-*ment*
punishment	INLEX	E-OR-P-OR-R-*ment*
puzzlement	INLEX	E-OR-P-OR-R-*ment*
re-enactment	NEWDER	E-OR-P-OR-R-*ment*
readjustment	NEWDER	E-OR-P-OR-R-*ment*
reapportionment	NEWDER	E-OR-P-OR-R-*ment*
recruitment	NEWDER	E-OR-P-OR-R-*ment*
redevelopment	NEWDER	E-OR-P-OR-R-*ment*
refinement	INLEX	E-OR-P-OR-R-*ment*
refreshment	INLEX	*E-OR-P-OR-R-*ment*
reimbursement	INLEX	E-OR-P-OR-R-*ment*
reinforcement	INLEX	E-OR-P-OR-R-*ment*
repayment	INLEX	E-OR-P-OR-R-*ment*
replacement	INLEX	E-OR-P-OR-R-*ment*
replenishment	NEWDER	E-OR-P-OR-R-*ment*
requirement	INLEX	E-OR-P-OR-R-*ment*
resentment	INLEX	E-OR-P-OR-R-*ment*
resettlement	NEWDER	E-OR-P-OR-R-*ment*
restatement	NEWDER	E-OR-P-OR-R-*ment*
retirement	INLEX	E-OR-P-OR-R-*ment*
revetment	INLEX	*E-OR-P-OR-R-*ment*
settlement	INLEX	E-OR-P-OR-R-*ment*
shipment	INLEX	E-OR-P-OR-R-*ment*
statement	INLEX	E-OR-P-OR-R-*ment*
transshipment	NEWDER	E-OR-P-OR-R-*ment*
treatment	INLEX	E-OR-P-OR-R-*ment*
under-achievement	NEWDER	E-OR-P-OR-R-*ment*
understatement	INLEX	E-OR-P-OR-R-*ment*
unemployment	INLEX	*E-OR-P-OR-R-*ment*
unfoldment	NEWDER	E-OR-P-OR-R-*ment*
vestment	INLEX	*E-OR-P-OR-R-*ment*

A.2.21 Words derived using *-age*

anchorage	INLEX	*E-AND-R-*age*
appendage	INLEX	E-AND-R-*age*
assemblage	INLEX	E-AND-R-*age*
average	INLEX	*E-AND-R-*age*
bandage	INLEX	*E-AND-R-*age*
blockage	INLEX	E-AND-R-*age*
bondage	INLEX	E-AND-R-*age*
breakage	INLEX	E-AND-R-*age*

carriage	INLEX	*E-AND-R-*age*
cleavage	INLEX	E-AND-R-*age*
coverage	INLEX	E-AND-R-*age*
dosage	INLEX	*E-AND-R-*age*
drainage	INLEX	E-AND-R-*age*
footage	INLEX	*E-AND-R-*age*
frontage	INLEX	*E-AND-R-*age*
garbage	INLEX	*E-AND-R-*age*
haulage	INLEX	E-AND-R-*age*
homage	INLEX	*E-AND-R-*age*
hostage	INLEX	*E-AND-R-*age*
intermarriage	INLEX	E-AND-R-*age*
leakage	INLEX	E-AND-R-*age*
leverage	INLEX	E-AND-R-*age*
linkage	INLEX	E-AND-R-*age*
marriage	INLEX	E-AND-R-*age*
massage	INLEX	*E-AND-R-*age*
message	INLEX	*E-AND-R-*age*
orphanage	INLEX	*E-AND-R-*age*
package	INLEX	E-AND-R-*age*
passage	INLEX	E-AND-R-*age*
rampage	INLEX	*E-AND-R-*age*
ravage	INLEX	*E-AND-R-*age*
reportage	INLEX	E-AND-R-*age*
salvage	INLEX	*E-AND-R-*age*
savage	INLEX	*E-AND-R-*age*
seepage	INLEX	E-AND-R-*age*
sewage	INLEX	*E-AND-R-*age*
shrinkage	INLEX	E-AND-R-*age*
spoilage	INLEX	E-AND-R-*age*
storage	INLEX	E-AND-R-*age*
stumpage	NEWDER	E-AND-R-*age*
usage	INLEX	E-AND-R-*age*
wastage	INLEX	E-AND-R-*age*
wreckage	INLEX	E-AND-R-*age*

A.2.22 Words derived using *mis-*

misbrand	NEWDER	INCOR*mis-*
miscalculate	INLEX	INCOR*mis-*
miscarry	INLEX	?INCOR*mis-*
misconstrue	INLEX	INCOR*mis-*
misfire	INLEX	*INCOR*mis-*
misgauge	NEWDER	INCOR*mis-*
misguide	INLEX	INCOR*mis-*
misinterpret	INLEX	INCOR*mis-*
misjudge	INLEX	INCOR*mis-*
mislead	INLEX	INCOR*mis-*
mismanage	INLEX	INCOR*mis-*
misname	INLEX	INCOR*mis-*
misperceive	NEWDER	INCOR*mis-*
misplace	INLEX	INCOR*mis-*
misquote	INLEX	INCOR*mis-*
misrelate	NEWDER	INCOR*mis-*
misrepresent	INLEX	INCOR*mis-*
missing	INLEX	*INCOR*mis-*
mistake	INLEX	INCOR*mis-*
mistrust	INLEX	*INCOR*mis-*
misunderstand	INLEX	INCOR*mis-*
misuse	INLEX	INCOR*mis-*
miswritten	NEWSTEMDER	*INCOR*mis-*

A.2.23 Words derived using *-able*

acceptable	INLEX	ABLE-*able*
accountable	INLEX	ABLE-*able*
achievable	NEWDER	ABLE-*able*
adaptable	INLEX	ABLE-*able*
adjustable	NEWDER	ABLE-*able*
admirable	INLEX	ABLE-*able*
adorable	INLEX	ABLE-*able*
advisable	INLEX	ABLE-*able*
agreeable	INLEX	ABLE-*able*
alienable	INLEX	ABLE-*able*
allocable	NEWSTEMDER	*ABLE-*able*
allowable	INLEX	ABLE-*able*
alterable	NEWDER	ABLE-*able*
analyzable	NEWSTEMDER	ABLE-*able*
answerable	INLEX	*ABLE-*able*
appeasable	NEWDER	ABLE-*able*
approachable	INLEX	ABLE-*able*
ascertainable	NEWDER	ABLE-*able*
attainable	NEWDER	ABLE-*able*
attributable	INLEX	ABLE-*able*
available	INLEX	*ABLE-*able*
avaliable	NEWSTEMDER	*ABLE-*able*
avoidable	NEWDER	ABLE-*able*
bearable	INLEX	ABLE-*able*
believable	INLEX	ABLE-*able*
breakable	NEWDER	ABLE-*able*
callable	NEWDER	ABLE-*able*
changeable	INLEX	ABLE-*able*
chargeable	INLEX	ABLE-*able*
combable	NEWDER	ABLE-*able*
combinable	NEWDER	ABLE-*able*
comfortable	INLEX	*ABLE-*able*
commendable	INLEX	ABLE-*able*
comparable	INLEX	ABLE-*able*
conceivable	INLEX	ABLE-*able*
conquerable	NEWDER	ABLE-*able*
conscionable	NEWSTEMDER	*ABLE-*able*
considerable	INLEX	*ABLE-*able*
contestable	INLEX	ABLE-*able*
curable	INLEX	ABLE-*able*
debatable	INLEX	ABLE-*able*
decipherable	INLEX	ABLE-*able*
deductable	NEWDER	ABLE-*able*
definable	INLEX	ABLE-*able*
deniable	INLEX	ABLE-*able*
dependable	INLEX	ABLE-*able*
deplorable	INLEX	ABLE-*able*
describable	INLEX	ABLE-*able*
desirable	INLEX	ABLE-*able*
detachable	NEWDER	ABLE-*able*
detectable	NEWDER	ABLE-*able*
determinable	NEWDER	ABLE-*able*
detestable	NEWDER	ABLE-*able*
diagnosable	NEWDER	ABLE-*able*
diagonalizable	NEWDER	ABLE-*able*
differentiable	NEWSTEMDER	*ABLE-*able*
disagreeable	INLEX	ABLE-*able*
discernable	NEWDER	ABLE-*able*
dispensable	INLEX	ABLE-*able*
disputable	INLEX	ABLE-*able*
distinguishable	INLEX	ABLE-*able*
distortable	NEWDER	ABLE-*able*

drinkable	NEWDER	ABLE-*able*
duplicable	NEWSTEMDER	*ABLE-*able*
eatable	INLEX	ABLE-*able*
endurable	NEWDER	ABLE-*able*
enforceable	NEWDER	ABLE-*able*
enjoyable	INLEX	ABLE-*able*
enviable	INLEX	ABLE-*able*
escapable	INLEX	ABLE-*able*
excusable	INLEX	ABLE-*able*
expandable	NEWDER	ABLE-*able*
expectable	NEWDER	ABLE-*able*
expendable	INLEX	ABLE-*able*
explainable	NEWDER	ABLE-*able*
fashionable	INLEX	*ABLE-*able*
favorable	INLEX	*ABLE-*able*
foreseeable	INLEX	ABLE-*able*
forgivable	NEWDER	ABLE-*able*
friable	INLEX	*ABLE-*able*
honorable	INLEX	*ABLE-*able*
identifiable	NEWDER	ABLE-*able*
imaginable	NEWDER	ABLE-*able*
imcomparable	NEWSTEMDER	*ABLE-*able*
impeachable	INLEX	ABLE-*able*
injectable	NEWDER	ABLE-*able*
interchangeable	INLEX	ABLE-*able*
interminable	INLEX	*ABLE-*able*
irresolvable	NEWSTEMDER	*ABLE-*able*
justifiable	INLEX	ABLE-*able*
killable	NEWDER	ABLE-*able*
livable	INLEX	ABLE-*able*
lovable	INLEX	ABLE-*able*
manageable	NEWDER	ABLE-*able*
marketable	NEWSTEMDER	ABLE-*able*
measurable	INLEX	ABLE-*able*
minable	INLEX	*ABLE-*able*
mistakable	INLEX	ABLE-*able*
movable	INLEX	ABLE-*able*
nameable	NEWSTEMDER	ABLE-*able*
notable	INLEX	ABLE-*able*
noticeable	INLEX	ABLE-*able*
observable	INLEX	ABLE-*able*
obtainable	INLEX	ABLE-*able*
paintable	NEWDER	ABLE-*able*
pardonable	INLEX	ABLE-*able*
passable	INLEX	ABLE-*able*
payable	INLEX	ABLE-*able*
perishable	INLEX	*ABLE-*able*
pitiable	INLEX	ABLE-*able*
plantable	NEWDER	ABLE-*able*
playable	INLEX	ABLE-*able*
pliable	INLEX	*ABLE-*able*
portable	INLEX	*ABLE-*able*
predictable	INLEX	ABLE-*able*
preferable	INLEX	ABLE-*able*
presentable	INLEX	ABLE-*able*
printable	INLEX	ABLE-*able*
probable	INLEX	*ABLE-*able*
procurable	NEWDER	ABLE-*able*
punishable	INLEX	ABLE-*able*
questionable	INLEX	ABLE-*able*
ratable	INLEX	ABLE-*able*
readable	INLEX	ABLE-*able*
reasonable	INLEX	*ABLE-*able*
recognizable	NEWDER	ABLE-*able*

recoverable	NEWDER	ABLE-*able*
reimburseable	NEWSTEMDER	ABLE-*able*
reliable	INLEX	ABLE-*able*
remarkable	INLEX	*ABLE-*able*
removable	NEWDER	ABLE-*able*
renewable	INLEX	ABLE-*able*
repayable	INLEX	ABLE-*able*
researchable	NEWDER	ABLE-*able*
resonable	NEWSTEMDER	*ABLE-*able*
respectable	INLEX	ABLE-*able*
seasonable	NEWDER	*ABLE-*able*
serviceable	INLEX	*ABLE-*able*
shakable	INLEX	ABLE-*able*
sinkable	NEWDER	ABLE-*able*
sizable	INLEX	*ABLE-*able*
speakable	INLEX	ABLE-*able*
standable	NEWDER	*ABLE-*able*
suable	NEWDER	ABLE-*able*
suitable	INLEX	*ABLE-*able*
superable	INLEX	*ABLE-*able*
supportable	NEWDER	ABLE-*able*
surmountable	INLEX	ABLE-*able*
taxable	NEWDER	ABLE-*able*
tellable	NEWDER	ABLE-*able*
thinkable	INLEX	ABLE-*able*
tintable	NEWDER	ABLE-*able*
traceable	NEWDER	ABLE-*able*
transplantable	NEWDER	ABLE-*able*
understandable	NEWDER	ABLE-*able*
usable	NEWDER	ABLE-*able*
useable	NEWSTEMDER	ABLE-*able*
valuable	INLEX	ABLE-*able*
variable	INLEX	ABLE-*able*
warrantable	NEWDER	ABLE-*able*
workable	INLEX	*ABLE-*able*

A.2.24 Words derived using *-ness*

abruptness	NEWDER	NESS-*ness*
absoluteness	NEWDER	NESS-*ness*
abstractedness	NEWDER	NESS-*ness*
acquisitiveness	NEWDER	NESS-*ness*
adroitness	NEWDER	NESS-*ness*
aggressiveness	NEWDER	NESS-*ness*
agreeableness	NEWDER	NESS-*ness*
alertness	NEWDER	NESS-*ness*
allusiveness	NEWSTEMDER	NESS-*ness*
aloneness	NEWDER	NESS-*ness*
aloofness	NEWDER	NESS-*ness*
amateurishness	NEWDER	NESS-*ness*
appropriateness	NEWDER	NESS-*ness*
aptness	NEWDER	NESS-*ness*
artfulness	NEWDER	NESS-*ness*
assertiveness	NEWDER	NESS-*ness*
astuteness	NEWDER	NESS-*ness*
awareness	NEWDER	NESS-*ness*
awfulness	NEWDER	NESS-*ness*
awkwardness	NEWDER	NESS-*ness*
badness	NEWDER	NESS-*ness*
baldness	NEWDER	NESS-*ness*
balkiness	NEWDER	NESS-*ness*
bitterness	NEWDER	NESS-*ness*

blackness	NEWDER	NESS-*ness*
blandness	NEWDER	NESS-*ness*
blindness	NEWDER	NESS-*ness*
bluntness	NEWDER	NESS-*ness*
boldness	NEWDER	NESS-*ness*
brashness	NEWDER	NESS-*ness*
brazenness	NEWDER	NESS-*ness*
brightness	NEWDER	NESS-*ness*
briskness	NEWDER	NESS-*ness*
business	INLEX	*NESS-*ness*
busyness	NEWSTEMDER	NESS-*ness*
callousness	NEWDER	NESS-*ness*
calmness	NEWDER	NESS-*ness*
carefulness	NEWDER	NESS-*ness*
carelessness	NEWDER	NESS-*ness*
cheerfulness	NEWDER	NESS-*ness*
childishness	NEWDER	NESS-*ness*
chumminess	NEWDER	NESS-*ness*
clannishness	NEWDER	NESS-*ness*
clearness	NEWDER	NESS-*ness*
cleverness	NEWDER	NESS-*ness*
cloddishness	NEWDER	NESS-*ness*
closeness	NEWDER	NESS-*ness*
coarseness	NEWDER	NESS-*ness*
cohesiveness	NEWDER	NESS-*ness*
coldness	NEWDER	NESS-*ness*
commonness	NEWDER	NESS-*ness*
completeness	NEWDER	NESS-*ness*
conciseness	NEWDER	NESS-*ness*
connectedness	NEWDER	NESS-*ness*
consciousness	INLEX	NESS-*ness*
coolness	NEWDER	NESS-*ness*
correctness	NEWDER	NESS-*ness*
courtliness	NEWDER	NESS-*ness*
covetousness	NEWDER	NESS-*ness*
coyness	NEWDER	NESS-*ness*
crassness	NEWDER	NESS-*ness*
creativeness	NEWDER	NESS-*ness*
credulousness	NEWDER	NESS-*ness*
crispness	NEWDER	NESS-*ness*
curtness	NEWDER	NESS-*ness*
dampness	NEWDER	NESS-*ness*
darkness	NEWDER	NESS-*ness*
deadliness	NEWDER	NESS-*ness*
deadness	NEWDER	NESS-*ness*
decisiveness	NEWDER	NESS-*ness*
decorativeness	NEWDER	NESS-*ness*
defensiveness	NEWDER	NESS-*ness*
deftness	NEWDER	NESS-*ness*
directness	NEWDER	NESS-*ness*
discursiveness	NEWDER	NESS-*ness*
disorderliness	NEWDER	NESS-*ness*
dizziness	NEWDER	NESS-*ness*
dreariness	NEWDER	NESS-*ness*
drunkenness	NEWDER	*NESS-*ness*
dryness	NEWSTEMDER	NESS-*ness*
dullness	NEWDER	NESS-*ness*
eagerness	NEWDER	NESS-*ness*
earnestness	NEWDER	NESS-*ness*
effectiveness	INLEX	NESS-*ness*
elusiveness	NEWDER	NESS-*ness*
emptiness	NEWDER	NESS-*ness*
exclusiveness	NEWDER	NESS-*ness*
expansiveness	NEWDER	NESS-*ness*

explicitness	NEWDER	NESS-*ness*
expressiveness	NEWDER	NESS-*ness*
expressivness	NEWSTEMDER	NESS-*ness*
exquisiteness	NEWDER	NESS-*ness*
extraneousness	NEWDER	NESS-*ness*
fairness	NEWDER	NESS-*ness*
familiarness	NEWDER	NESS-*ness*
fierceness	NEWDER	NESS-*ness*
fineness	NEWDER	NESS-*ness*
firmness	NEWDER	NESS-*ness*
fitness	INLEX	NESS-*ness*
flatness	NEWDER	NESS-*ness*
fondness	NEWDER	NESS-*ness*
foolishness	NEWDER	NESS-*ness*
forcefulness	NEWDER	NESS-*ness*
forgetfulness	NEWDER	NESS-*ness*
forthrightness	NEWDER	NESS-*ness*
frankness	NEWDER	NESS-*ness*
freshness	NEWDER	NESS-*ness*
friendliness	NEWDER	NESS-*ness*
fruitfulness	NEWDER	NESS-*ness*
fullness	INLEX	NESS-*ness*
garishness	NEWDER	NESS-*ness*
gentleness	NEWDER	NESS-*ness*
giddiness	NEWDER	NESS-*ness*
givenness	NEWDER	NESS-*ness*
gladness	NEWDER	NESS-*ness*
godliness	NEWDER	NESS-*ness*
goodness	INLEX	NESS-*ness*
greatness	NEWDER	NESS-*ness*
greenness	NEWDER	NESS-*ness*
grimness	NEWDER	NESS-*ness*
guardedness	NEWDER	NESS-*ness*
guiltiness	NEWDER	NESS-*ness*
happiness	INLEX	NESS-*ness*
hardness	INLEX	NESS-*ness*
harshness	NEWDER	NESS-*ness*
haughtiness	NEWDER	NESS-*ness*
heaviness	NEWDER	NESS-*ness*
helpfulness	NEWDER	NESS-*ness*
helplessness	NEWDER	NESS-*ness*
highness	NEWDER	*NESS-*ness*
hoarseness	NEWDER	NESS-*ness*
holiness	INLEX	NESS-*ness*
hollowness	NEWDER	NESS-*ness*
homesickness	NEWDER	NESS-*ness*
hopelessness	NEWDER	NESS-*ness*
humanness	NEWDER	NESS-*ness*
huskiness	NEWDER	NESS-*ness*
idleness	NEWDER	NESS-*ness*
illness	INLEX	NESS-*ness*
inappropriateness	NEWDER	NESS-*ness*
incisiveness	NEWDER	NESS-*ness*
inclusiveness	NEWDER	NESS-*ness*
incompleteness	NEWDER	NESS-*ness*
indecisiveness	NEWDER	NESS-*ness*
indefiniteness	NEWDER	NESS-*ness*
ineffectiveness	NEWDER	NESS-*ness*
ineptness	NEWDER	NESS-*ness*
interconnectedness	NEWDER	NESS-*ness*
inwardness	INLEX	*NESS-*ness*
jewishness	NEWSTEMDER	NESS-*ness*
joblessness	NEWDER	NESS-*ness*
justness	NEWDER	NESS-*ness*

kindliness	NEWDER	NESS-*ness*
kindness	INLEX	NESS-*ness*
laxness	NEWDER	NESS-*ness*
light-headedness	NEWDER	NESS-*ness*
light-mindedness	NEWDER	NESS-*ness*
lightness	INLEX	NESS-*ness*
likeness	INLEX	*NESS-*ness*
literalness	NEWDER	NESS-*ness*
liveliness	NEWDER	NESS-*ness*
loneliness	NEWDER	NESS-*ness*
looseness	NEWDER	NESS-*ness*
lousiness	NEWDER	NESS-*ness*
loveliness	NEWDER	NESS-*ness*
ludicrousness	NEWDER	NESS-*ness*
madness	INLEX	NESS-*ness*
maleness	NEWDER	NESS-*ness*
manliness	NEWDER	NESS-*ness*
matter-of-factness	NEWDER	NESS-*ness*
meaningfulness	NEWDER	NESS-*ness*
meanness	NEWDER	NESS-*ness*
mustiness	NEWDER	NESS-*ness*
nakedness	NEWDER	NESS-*ness*
narrowness	NEWDER	NESS-*ness*
naturalness	INLEX	NESS-*ness*
nearness	NEWSTEMDER	NESS-*ness*
neatness	NEWDER	NESS-*ness*
neighborliness	NEWDER	NESS-*ness*
nervousness	NEWDER	NESS-*ness*
numbness	NEWDER	NESS-*ness*
objectiveness	NEWDER	NESS-*ness*
obtrusiveness	NEWDER	NESS-*ness*
obviousness	NEWDER	NESS-*ness*
oneness	NEWDER	NESS-*ness*
orderliness	NEWDER	NESS-*ness*
oversoftness	NEWDER	NESS-*ness*
paleness	NEWDER	NESS-*ness*
passiveness	NEWDER	NESS-*ness*
pastness	NEWDER	NESS-*ness*
pettiness	NEWDER	NESS-*ness*
physicalness	NEWDER	NESS-*ness*
pleasantness	NEWDER	NESS-*ness*
plumpness	NEWDER	NESS-*ness*
politeness	NEWDER	NESS-*ness*
pompousness	NEWDER	NESS-*ness*
powerfulness	NEWDER	NESS-*ness*
preparedness	INLEX	NESS-*ness*
presentness	NEWDER	NESS-*ness*
pressivness	NEWSTEMDER	*NESS-*ness*
prettiness	NEWDER	NESS-*ness*
proneness	NEWDER	NESS-*ness*
pseudo-happiness	NEWDER	NESS-*ness*
queasiness	NEWDER	NESS-*ness*
quickness	NEWDER	NESS-*ness*
quietness	NEWDER	NESS-*ness*
raggedness	NEWDER	NESS-*ness*
readiness	INLEX	NESS-*ness*
realness	NEWDER	NESS-*ness*
recklessness	NEWDER	NESS-*ness*
relatedness	NEWDER	NESS-*ness*
relentlessness	NEWDER	NESS-*ness*
religiousness	NEWDER	NESS-*ness*
remoteness	NEWDER	NESS-*ness*
resourcefulness	NEWDER	NESS-*ness*
responsiveness	NEWDER	NESS-*ness*

restlessness	NEWDER	NESS-*ness*
retentiveness	NEWDER	NESS-*ness*
richness	INLEX	NESS-*ness*
righteousness	NEWDER	NESS-*ness*
rightness	NEWDER	NESS-*ness*
robustness	NEWDER	NESS-*ness*
roughness	INLEX	NESS-*ness*
roundness	NEWDER	NESS-*ness*
ruddiness	NEWDER	NESS-*ness*
rudeness	NEWDER	NESS-*ness*
ruefulness	NEWDER	NESS-*ness*
ruthlessness	NEWDER	NESS-*ness*
sacredness	NEWDER	NESS-*ness*
sadness	NEWDER	NESS-*ness*
saintliness	NEWDER	NESS-*ness*
sameness	INLEX	*NESS-*ness*
scratchiness	NEWDER	NESS-*ness*
self-consciousness	NEWDER	NESS-*ness*
self-righteousness	NEWDER	NESS-*ness*
selfishness	NEWDER	NESS-*ness*
selflessness	NEWDER	NESS-*ness*
separateness	NEWDER	?NESS-*ness*
seriousness	NEWDER	NESS-*ness*
shallowness	NEWDER	NESS-*ness*
sharpness	NEWDER	NESS-*ness*
shortness	NEWDER	NESS-*ness*
shortsightedness	NEWDER	NESS-*ness*
shrillness	NEWDER	NESS-*ness*
sickness	INLEX	*NESS-*ness*
sinfulness	NEWDER	NESS-*ness*
singleness	INLEX	NESS-*ness*
sinuousness	NEWDER	NESS-*ness*
skillfulness	NEWDER	NESS-*ness*
slovenliness	NEWDER	NESS-*ness*
slowness	NEWDER	NESS-*ness*
slyness	NEWSTEMDER	NESS-*ness*
smallness	NEWDER	NESS-*ness*
smoothness	NEWDER	NESS-*ness*
softness	NEWDER	NESS-*ness*
solicitousness	NEWDER	NESS-*ness*
soreness	NEWDER	NESS-*ness*
soundness	NEWDER	NESS-*ness*
spaciousness	NEWDER	NESS-*ness*
speechlessness	NEWDER	NESS-*ness*
squeamishness	NEWDER	NESS-*ness*
staginess	NEWDER	NESS-*ness*
steadiness	NEWDER	NESS-*ness*
stiffness	NEWDER	NESS-*ness*
stillness	NEWDER	NESS-*ness*
strangeness	NEWDER	NESS-*ness*
stubbornness	NEWDER	NESS-*ness*
suddenness	NEWDER	NESS-*ness*
suppleness	NEWDER	NESS-*ness*
surfaceness	NEWDER	*NESS-*ness*
sweetness	NEWDER	NESS-*ness*
swiftness	NEWDER	NESS-*ness*
tactlessness	NEWDER	NESS-*ness*
tardiness	NEWDER	NESS-*ness*
tenderness	NEWDER	NESS-*ness*
thankfulness	NEWDER	NESS-*ness*
thickness	INLEX	*NESS-*ness*
thinness	NEWDER	NESS-*ness*
thoroughness	NEWDER	NESS-*ness*
thoughtfulness	NEWDER	NESS-*ness*

tidiness	NEWDER	NESS-*ness*
timeliness	NEWDER	NESS-*ness*
tiredness	NEWDER	NESS-*ness*
toughness	NEWDER	NESS-*ness*
truthfulness	NEWDER	NESS-*ness*
tunefulness	NEWDER	NESS-*ness*
ugliness	NEWDER	NESS-*ness*
unawareness	NEWDER	NESS-*ness*
underhandedness	NEWDER	NESS-*ness*
uneasiness	NEWDER	NESS-*ness*
unhappiness	NEWDER	NESS-*ness*
uniqueness	NEWDER	NESS-*ness*
unnaturalness	NEWDER	NESS-*ness*
unpleasantness	NEWDER	NESS-*ness*
unselfconsciousness	NEWDER	NESS-*ness*
untidiness	NEWDER	NESS-*ness*
untrustworthiness	NEWSTEMDER	NESS-*ness*
unwillingness	NEWDER	NESS-*ness*
usefulness	INLEX	NESS-*ness*
uselessness	NEWDER	NESS-*ness*
vagueness	NEWDER	NESS-*ness*
viciousness	NEWDER	NESS-*ness*
vividness	NEWDER	NESS-*ness*
vociferousness	NEWDER	NESS-*ness*
wakefulness	NEWDER	NESS-*ness*
weakness	INLEX	NESS-*ness*
weariness	NEWDER	NESS-*ness*
weightlessness	NEWDER	NESS-*ness*
wetness	NEWDER	NESS-*ness*
whiteness	NEWDER	NESS-*ness*
wholeness	NEWDER	NESS-*ness*
wickedness	NEWDER	NESS-*ness*
wildness	NEWDER	NESS-*ness*
willingness	NEWDER	NESS-*ness*
wonderfulness	NEWDER	NESS-*ness*
worthlessness	NEWDER	NESS-*ness*
wretchedness	NEWDER	NESS-*ness*

A.2.25 Words serving as bases for -*ful*

art	INLEX	ABST-*ful*
bane	INLEX	ABST-*ful*
bash	INLEX	*ABST-*ful*
beauty	INLEX	ABST-*ful*
bliss	NEWDERSTEM	ABST-*ful*
brim	INLEX	*ABST-*ful*
care	INLEX	ABST-*ful*
cheer	INLEX	ABST-*ful*
color	INLEX	ABST-*ful*
deceit	INLEX	ABST-*ful*
delight	INLEX	ABST-*ful*
disgrace	INLEX	ABST-*ful*
distaste	INLEX	ABST-*ful*
dole	INLEX	*ABST-*ful*
doubt	INLEX	ABST-*ful*
dread	INLEX	ABST-*ful*
faith	INLEX	ABST-*ful*
fancy	INLEX	ABST-*ful*
fate	INLEX	ABST-*ful*
fear	INLEX	ABST-*ful*
fit	INLEX	*ABST-*ful*
force	INLEX	ABST-*ful*

forgit	NEWSTEM	*ABST-*ful*
fright	INLEX	ABST-*ful*
fruit	INLEX	ABST-*ful*
gain	INLEX	ABST-*ful*
grace	INLEX	ABST-*ful*
grate	INLEX	ABST-*ful*
harm	NEWDERSTEM	ABST-*ful*
hate	INLEX	ABST-*ful*
health	INLEX	ABST-*ful*
help	INLEX	ABST-*ful*
hope	INLEX	ABST-*ful*
joy	INLEX	ABST-*ful*
law	INLEX	ABST-*ful*
light	INLEX	ABST-*ful*
lust	INLEX	ABST-*ful*
master	INLEX	*ABST-*ful*
meaning	INLEX	ABST-*ful*
mercy	INLEX	ABST-*ful*
mind	INLEX	ABST-*ful*
pain	INLEX	ABST-*ful*
peace	INLEX	ABST-*ful*
pity	INLEX	ABST-*ful*
play	INLEX	ABST-*ful*
plenty	INLEX	ABST-*ful*
power	INLEX	ABST-*ful*
purpose	INLEX	ABST-*ful*
remorse	INLEX	ABST-*ful*
resent	NEWSTEM	*ABST-*ful*
resource	INLEX	ABST-*ful*
respect	INLEX	ABST-*ful*
rest	INLEX	ABST-*ful*
right	INLEX	ABST-*ful*
scorn	NEWDERSTEM	ABST-*ful*
shame	INLEX	ABST-*ful*
sin	INLEX	ABST-*ful*
skill	INLEX	ABST-*ful*
sloth	INLEX	ABST-*ful*
song	NEWDERSTEM	ABST-*ful*
soul	INLEX	ABST-*ful*
stress	NEWDERSTEM	ABST-*ful*
success	INLEX	ABST-*ful*
success	NEWDERSTEM	ABST-*ful*
taste	INLEX	ABST-*ful*
thought	INLEX	ABST-*ful*
truth	INLEX	ABST-*ful*
tune	INLEX	ABST-*ful*
use	INLEX	ABST-*ful*
wake	INLEX	ABST-*ful*
waste	INLEX	ABST-*ful*
watch	INLEX	ABST-*ful*
wish	NEWDERSTEM	ABST-*ful*
woe	INLEX	ABST-*ful*
wonder	INLEX	ABST-*ful*
worship	INLEX	ABST-*ful*
wrath	NEWDERSTEM	ABST-*ful*
wrong	INLEX	ABST-*ful*
youth	INLEX	ABST-*ful*

A.2.26 Words derived using -*ful*

artful	INLEX	LESSANT-*ful*
careful	INLEX	LESSANT-*ful*

colorful	INLEX	LESSANT-*ful*
fearful	INLEX	LESSANT-*ful*
fruitful	INLEX	LESSANT-*ful*
harmful	NEWDER	LESSANT-*ful*
helpful	INLEX	*LESSANT-*ful*
hopeful	INLEX	*LESSANT-*ful*
lawful	INLEX	LESSANT-*ful*
meaningful	INLEX	LESSANT-*ful*
merciful	INLEX	LESSANT-*ful*
mindful	INLEX	*LESSANT-*ful*
painful	INLEX	LESSANT-*ful*
pitiful	INLEX	*LESSANT-*ful*
powerful	INLEX	LESSANT-*ful*
purposeful	INLEX	LESSANT-*ful*
remorseful	INLEX	LESSANT-*ful*
restful	INLEX	*LESSANT-*ful*
sinful	INLEX	LESSANT-*ful*
tasteful	INLEX	LESSANT-*ful*
thoughtful	INLEX	LESSANT-*ful*
useful	INLEX	LESSANT-*ful*

A.2.27 Words derived using *-less*

artless	INLEX	FULANT-*less*
careless	INLEX	FULANT-*less*
colorless	INLEX	FULANT-*less*
fearless	INLEX	FULANT-*less*
fruitless	INLEX	FULANT-*less*
harmless	INLEX	FULANT-*less*
helpless	INLEX	*FULANT-*less*
hopeless	INLEX	*FULANT-*less*
lawless	INLEX	FULANT-*less*
meaningless	INLEX	FULANT-*less*
merciless	INLEX	FULANT-*less*
mindless	INLEX	*FULANT-*less*
painless	INLEX	FULANT-*less*
pitiless	INLEX	*FULANT-*less*
powerless	INLEX	FULANT-*less*
purposeless	INLEX	FULANT-*less*
remorseless	NEWDER	FULANT-*less*
restless	INLEX	*FULANT-*less*
sinless	INLEX	FULANT-*less*
tasteless	INLEX	FULANT-*less*
thoughtless	INLEX	FULANT-*less*
useless	INLEX	FULANT-*less*