TAGGING GUIDE FOR THE X' THEORY GRAMMAR

Julie A. Van Dyke

August, 1991

ABSTRACT

The X' Theory Grammar is a computational grammar that models English syntax according to the X' (pronounced "X-bar") linguistic framework. The X' theory gives a more generalized analysis of syntax than other generative linguistic theories that have previously been available to computational researchers. It is based on an abstract structural "template" that is applied to syntactic constituents at all levels. This has facilitated creating a grammar with a very broad coverage that can be compatible with many natural language generation or understanding systems. In order to fully exploit this grammar, a computational lexicon entry must include specific features which the grammar tests for as it generates sentence analyses. This tagging guide is meant to provide the dictionary builder with assistance in building such a dictionary. Use of this guide in conjunction with other documentation on this grammar will result in the creation of a dictionary with great descriptive power- not only for use with the X' theory grammar, but for other natural language applications as well.

CONTENTS

TABLES

1 INTRODUCTION

This document contains specific descriptions of the subcategorization codes and other syntactic information that must appear in the lexical entries to be used with the X' Theory grammar as described in (Van Dyke, 1991a). It is intended to help a person unfamiliar with this system create new entries for the system's dictionary. Here I indicate the lexical coding the grammar requires and discuss how these codes might be derived from an already tagged corpus. It will also be useful for the dictionary builder to refer to the test suite of this grammar given in (Van Dyke, 1991b). Here can be found more detailed explicit examples of the structures that are licensed by particular features of a word's lexical entry.

2 CONSIDERATIONS FOR DICTIONARY BUILDING

A major concern for building this dictionary is to enable a person not familiar with the implementation of the grammar to easily construct lexical entries. My first step toward this end has been to adopt traditional word categories of the sort explained in the introduction to any good desk dictionary. These will include determiners (of which articles are a subset), nouns, adjectives, verbs, adverbs, prepositions, pronouns, and conjunctions. This allows the word categories to be non-specific to this project and permits the grammar to share on-line dictionaries already available. All that would be required for this system is that the additional features described here be added into the lexical entries. This addition is considered less demanding than a complete recategorization of each word in the lexicon.

The features on each word hold the specifications for how that word can be used and what structures can follow it. In the case of nouns and verbs, they were devised based on the coding in the Oxford Advanced Learner's Dictionary (OALD) (Cowie, 1989). Attention was also paid to the word tags that might be found in a large corpus of English, such as the Brown Corpus (Francis and Kucera, 1982). This corpus contains approximately 1,000,000 words tagged with combinations of 87 word "tags." These tags typically give the word's syntactic category, but other usage information can be included. For example, the word "Great" in "Alexander the Great's" would be tagged with the adjective tag "JJ" plus the possessive tag "$". There are also tags to denote foreign words, "FW", cited words, "NC", and words appearing in titles, "TL."

The Brown Corpus of American English was the earliest tagged corpus, and for this reason most other corpora tagsets are derived from Brown's. There are drawbacks to using the Brown tagset which stem mainly from the fact that it does not sufficiently differentiate certain kinds of words. For example, the Brown corpus only has four basic tags for nouns: "NN" for singular common nouns, "NNS" for plural common nouns, "NP" for proper nouns, and "NR" for adverbial nouns. With this system there is no way to distinguish mass nouns and this is information necessary for determiner subcategorization (i.e., to rule out sentences such as "*a furniture arrived" and "boy walked"). Thus, there is not a direct correspondence between the codes used in this project and the Brown tags. Nonetheless, when appropriate I will note what Brown corpus tags correspond to the lexicon features used here. This will facilitate automatic translation of corpus tags to the codes needed here because the tags can serve as a starting point in the process. It will be necessary to manually check each word tagged with the Brown tags to check and further categorize the word; however, this has been viewed as a critical process in many automatic text tagging projects (Keulen, 1986), (Marcus et al., 1990).

One of several alternate tagsets, also based on Brown's tags, was devised to assign categories to the LOB corpus of British English. This tagset is significantly more descriptive than the Brown set; it not only differentiates common and mass nouns, but also differentiates mensural, proper, and titular nouns, more clearly specifies the pronouns, separates relative and wh-words, and identifies negative words. In addition there is a Lancaster tagset which was devised specifically for parsing and therefore makes very fine distinctions between words. Of course as these tagsets become more descriptive, they are also larger and more complicated to use. The fact that they are based on the Brown tagset should make my Brown tag notations helpful for using these corpora.

A final attempt to facilitate using this system with an already existing dictionary is that the values assigned to the properties in the dictionary, such as (SG) as the argument to the NUMBER property of a noun), have been constructed to work with macros. This would allow the numbering system used in a pre-existing dictionary to be hidden from the grammar: the grammar would access the macro rather than the direct number information. The surface value will be what the grammar calls for, but the internal specifications can come from the new dictionary. Exploiting this capability may require some extra macro programming, but will allow for maximal compatibility between systems.

3 SOURCES

The categories used to create the dictionary for this grammar were taken from the American Heritage Dictionary, Second College Edition (Revised edition, 1976). This dictionary was used as representative of the generally accepted categoric interpretations of words. Often the categories found here are less specific than those found in other dictionaries, but the objective in using them was to preserve main-stream word interpretations.

For the verb subcategorization information I have gone to a more specialized dictionary: the Oxford Advanced Learner's Dictionary (OALD) (Cowie, 1989). The OALD divides verbs into five types: linking, intransitive, transitive, complex-transitive, and di-transitive. In the section below I will describe each of these verb types and provide actual parses from the system to illustrate the differences in the structures they produce.

A linking verb functions as an equivalence which assigns characteristics described in the complement to the subject. There are two kinds of syntactic structures produced by these verbs depending on whether they take adjective or noun complements. The predicate nominative construction is illustrated below in (1), while the predicate adjective construction is illustrated in (2):

(1)

```
(DCL
      (CP (SPEC NIL)
        (CBAR (HEAD NIL)
           (COMP
            (IP
             (SPEC
              (DP (SPEC NIL)
                (DBAR (HEAD THE)
                   (COMP
                    (NP (SPEC NIL)
                      (NBAR (HEAD BOY) (NU (SG))))))))
             (IBAR (HEAD (AGR 3SGPRES))
                (COMP
                 (VP (SPEC NIL)
                   (VBAR (HEAD BE)
                      (COMP
                       (SUBJECT_COMPLEMENT
                        (DEGP (SPEC NIL)
                           (DEGBAR (HEAD NIL)
                               (COMP
                                (AP (SPEC NIL)
                                  (ABAR (HEAD SICK)))))))))))))))))
```

The boy is sick

(2)

```
(DCL
      (CP (SPEC NIL)
        (CBAR (HEAD NIL)
           (COMP
            (IP
             (SPEC
              (DP (SPEC NIL)
                (DBAR (HEAD THE)
                   (COMP
                    (NP (SPEC NIL)
                      (NBAR (HEAD MAN) (NU (SG))))))))
             (IBAR (HEAD (AGR 3SGPRES))
                (COMP
                 (VP (SPEC NIL)
                   (VBAR (HEAD BE)
                      (COMP
                       (SUBJECT_COMPLEMENT
                        (DP (SPEC NIL)
                          (DBAR (HEAD A)
                             (COMP
                              (NP (SPEC NIL)
                                (NBAR (HEAD TEACHER)
                                   (NU (SG))))))))))))))))))
```

The man is a teacher

Intransitive verbs do not subcategorize for complements, although they can have different kinds of adjuncts. The verb codes determine what kinds of adjuncts these verbs can occur with: either a PP, a DEGP, a DP, or an EC. Because they are adjuncts, none of these structures are necessary for forming grammatical sentences and a single verb will very often have codes for a number of these adjuncts. The following structures are produced by the system; the parse in (3) shows a simple intransitive verb with no adjunct, and the parse in (4) shown a prepositional phrase adjunct.

(3)

```
(DCL
     (CP (SPEC NIL)
       (CBAR (HEAD NIL)
          (COMP
           (IP
            (SPEC
             (DP (SPEC ALL)
               (DBAR (HEAD THE)
                  (COMP
                   (NP (SPEC NIL)
                     (NBAR (HEAD STUDENTS)
                        (NU (PL))))))))
            (IBAR (HEAD (AGR 3PLPAST))
               (COMP (VP (SPEC NIL)
                      (VBAR (HEAD CHATTER))))))))))
```

All the students chattered

(4)

```
(DCL
 (CP (SPEC NIL)
   (CBAR (HEAD NIL)
      (COMP
       (IP
        (SPEC
         (DP (SPEC NIL)
           (DBAR (HEAD NIL)
              (COMP
               (NP (SPEC MANY)
                 (NBAR (HEAD STUDENTS) (NU (PL))))))))
        (IBAR (HEAD (AGR 3PLPAST))
           (COMP
            (VP (SPEC NIL)
              (VBAR (HEAD CHATTER)
                 (ADJUNCT
                  (PP (SPEC NIL)
                    (PBAR (HEAD ABOUT)
                       (COMP
                        (DP (SPEC NIL)
                          (DBAR (HEAD THE)
                             (COMP
                              (NP (SPEC NIL)
                                (NBAR
                                 (HEAD EXAM) (NU (SG))))))))))))))))))))
```

Many students chattered about the exam

Transitive verbs are considered to be two-place predicates, taking a subject and a direct object. The most typical object is a determiner phrase, but other structures that can serve as objects include small clauses of various kinds, ordinary CP clauses whose COMP node is often filled with “that,” and Exceptional clauses. Below in (5) is an example of a structure with a DP object, and (6) shows an example of a CP object:

(5)

```
(DCL
 (CP (SPEC NIL)
   (CBAR (HEAD NIL)
      (COMP
       (IP
        (SPEC
         (DP (SPEC NIL)
           (DBAR (HEAD THE)
              (COMP
               (NP (SPEC NIL)
                 (NBAR (HEAD BOY) (NU (SG))))))))
        (IBAR (HEAD (AGR 3SGPAST))
           (COMP
            (VP (SPEC NIL)
              (VBAR (HEAD HIT)
                 (COMP
                  (OBJ
                   (DP (SPEC NIL)
                     (DBAR (HEAD THE)
                        (COMP
                         (NP (SPEC NIL)
                           (NBAR (HEAD BALL) (NU (SG)))))))))))))))))))
```

The boy hit the ball

(6)

```
(DCL
 (CP (SPEC NIL)
   (CBAR (HEAD NIL)
      (COMP
       (IP
        (SPEC
         (DP (SPEC NIL)
           (DBAR (HEAD THE)
              (COMP
               (NP (SPEC NIL)
                 (NBAR (HEAD TEACHER) (NU (SG))))))))
        (IBAR (HEAD (AGR 3SGPAST))
           (COMP
            (VP (SPEC NIL)
              (VBAR (HEAD BELIEVE)
                 (COMP
                  (OBJ
                   (DCL
                    (CP (SPEC NIL)
                      (CBAR (HEAD (COMPLEMENTIZER THAT))
                         (COMP
                          (IP
                           (SPEC
                            (DP (SPEC NIL)
                              (DBAR (HEAD THE)
                                 (COMP
                                  (NP (SPEC NIL)
                                   (NBAR (HEAD EXAM) (NU (SG))))))))
                           (IBAR (HEAD (AGR 3SGPAST))
                            (COMP
                             (VP (SPEC NIL)
                               (VBAR (HEAD BE)
                                  (COMP
                                   (SUBJECT_COMPLEMENT
                                     (DEGP (SPEC NIL)
                                        (DEGBAR (HEAD NIL)
                                         (COMP
                                           (AP (SPEC NIL)
                                             (ABAR (HEAD DIFFICULT)))))))))))))
                         )))))))))))))))
```

Complex-Transitive is the name the OALD gives to verbs that are three-place predicates. These verbs have a primary DP object and a secondary object which modifies the primary object. The secondary object could take the form of a verb phrase with the verb ending in -ing, an infinitival verb phrase, determiner phrases, adjective phrases, and exceptional clauses. Government and Binding syntax analyzes the structures these verbs produce in terms of small clauses and this is how they have been implemented here. In the examples below, I show an adjective degree phrase object complement in (7), an exceptional clause object complement in (8), and an infinitival small clause object complement in (9).[1]

(7)

```
(DCL
      (CP (SPEC NIL)
        (CBAR (HEAD NIL)
           (COMP
            (IP
             (SPEC
              (DP (SPEC NIL)
                (DBAR (HEAD THE)
                   (COMP
                    (NP (SPEC NIL)
                      (NBAR (HEAD TEACHER)
                         (NU (SG))))))))
             (IBAR (HEAD (AGR 3SGPRES))
                (COMP
                 (VP (SPEC NIL)
                   (VBAR (HEAD KEEP)
                      (COMP
                       (OBJ
                        (SMALL_CLAUSE
                         (DP (SPEC NIL)
                           (DBAR (HEAD THE)
                              (COMP
                               (NP (SPEC NIL)
                                 (NBAR (HEAD STUDENTS)
                                    (NU (PL)))))))
                         (DEGP (SPEC NIL)
                            (DEGBAR (HEAD NIL)
                                (COMP
                                 (AP (SPEC NIL)
                                   (ABAR (HEAD BUSY)))))))))))))))))))
The teacher keeps the students busy
```

1. Notice in examples (8) and (9) the subject of the EC and SC is PRO and TRACE, respectively. The structure in (8) accounts for the analysis GB gives to Object-Controlled Equi and Subject-to-Object Raising constructions. GB predicts that these are derived through different mechanisms, but their surface structures are similar and I have therefore accounted for them similarly. Refer to Chapter 5 for further discussion of this issue.

(8)

```
(DCL
         (CP (SPEC NIL)
            (CBAR (HEAD NIL)
                 (COMP
                   (IP
                     (SPEC
                       (DP (SPEC NIL)
                          (DBAR (HEAD THE)
                               (COMP
                                 (NP (SPEC NIL)
                                    (NBAR (HEAD MAN)
                                         (NU (SG))))))))
                     (IBAR (HEAD (AGR 3SGPAST))
                          (COMP
                            (VP (SPEC NIL)
                               (VBAR (HEAD FORCE)
                                    (COMP
                                      (OBJ
                                        (DP (SPEC NIL)
                                           (DBAR (HEAD THE)
                                                (COMP
                                                  (NP (SPEC NIL)
                                                     (NBAR (HEAD WOMAN)
                                                          (NU (SG)))))))
                                        (OBJ_COMP
                                          (EXCEPTIONAL_CLAUSE
                                   (DP (DBAR PRO))
                                         (INFL TO)
                                           (VP (SPEC NIL)
                                             (VBAR (HEAD GIVE)
                                                  (COMP
                                                    (IOBJ
                                                      (DP (SPEC NIL)
                                                         (DBAR (HEAD (PRO HIM))
                                                           (NU (SG))
                                                         (COMP NIL)))
                                                          (OBJ
                                                              (DP (SPEC NIL)
                                                          (DBAR (HEAD (PRO HER))
                                                            (COMP (NP (SPEC NIL)
                                                               (NBAR
                                                               (HEAD MONEY)
                                                               (NU (SG)))))))))))))))))))))))))))
```

The man forced the woman to give him her money

(9)

```
(DCL
      (CP (SPEC NIL)
        (CBAR (HEAD NIL)
      (COMP
            (IP
             (SPEC
              (DP (SPEC NIL)
                (DBAR (HEAD THE)
                   (COMP
                    (NP (SPEC NIL)
                      (NBAR (HEAD TEACHER)
                         (NU (SG))))))))
             (IBAR (HEAD (AGR 3SGPRES))
                (COMP
                 (VP (SPEC NIL)
                   (VBAR (HEAD LET)
                      (COMP
                       (OBJ
                        (DP (SPEC NIL)
                          (DBAR (HEAD (PRO HER))
                             (COMP
                              (NP (SPEC NIL)
                                (NBAR (HEAD STUDENTS)
                                   (NU (PL)))))))
                        (OBJ_COMP
                         (SMALL_CLAUSE
                      (DP (DBAR TRACE))
                           (VP (SPEC NIL)
                             (VBAR (HEAD PLAY)))))))
                         (ADJUNCT
                          (PP (SPEC NIL)
                             (PBAR (HEAD IN)
                                 (COMP
                                    (DP (SPEC NIL)
                                (DBAR (HEAD (PRO HER))
                                      (COMP
                                       (NP (SPEC NIL)
                                     (NBAR (HEAD CLASSES)
                                           (NU (PL)))))))))))))))))))))
```

The teacher lets her students play in her classes

Di-transitive verbs are those that take an indirect object, which is either a DP or a PP. Their object can be another DP, a PP, a CP, or an Exceptional Clause. They are different from the Complex-transitive verbs because the two constituents following them have particular semantic roles (i.e., object and indirect object). In the two examples that follow, I have given the most typical D-transitive verb construction in (10) and a construction in (11) having a PP indirect object and a CP object.

(10)

```
(DCL
      (CP (SPEC NIL)
        (CBAR (HEAD NIL)
           (COMP
            (IP
             (SPEC
              (DP (SPEC NIL)
                (DBAR (HEAD THE)
                   (COMP
                    (NP (SPEC NIL)
                      (NBAR (HEAD MAN)
                         (NU (SG))))))))
             (IBAR (HEAD (AGR 3SGPAST))
                (COMP
                 (VP (SPEC NIL)
                   (VBAR (HEAD TEACH)
                      (COMP
                       (OBJ
                        (DP (SPEC NIL)
                          (DBAR (HEAD NIL)
                             (COMP
                              (NP (SPEC NIL)
                                (NBAR (HEAD ENGLISH)
                                   (NU (SG/PL)))))))
                        (IOBJ
                         (PP (SPEC NIL)
                           (PBAR (HEAD TO)
                              (COMP
                               (DP (SPEC NIL)
                                 (DBAR (HEAD THE)
                                    (COMP
                                     (NP (SPEC NIL)
                                      (NBAR (HEAD STUDENTS)
                                       (NU (PL))))))))))))))))))))))))
```

The man taught English to the students.

(11)

```
(DCL
     (CP (SPEC NIL)
   (CBAR (HEAD NIL)
           (COMP
            (IP (SPEC
             (DP (SPEC NIL)
               (DBAR (HEAD THE)
                  (COMP
                   (NP (SPEC NIL)
                     (NBAR (HEAD WOMAN) (NU (SG))))))))
             (IBAR (HEAD (AGR 3SGPAST))
               (COMP
                (VP (SPEC NIL)
                  (VBAR (HEAD ANNOUNCE)
                     (COMP
                      (IOBJ
                       (PP (SPEC NIL)
                         (PBAR (HEAD TO)
                            (COMP
                             (DP (SPEC NIL)
                               (DBAR (HEAD (PRO HER))
                                  (COMP
                                   (NP (SPEC NIL)
                                     (NBAR (HEAD GUESTS)
                                      (NU (PL))))))))))
                       (OBJ
                        (DCL
                         (CP (SPEC NIL)
                           (CBAR
                             (HEAD (COMPLEMENTIZER THAT))
                            (COMP
                              (IP (SPEC (DP (SPEC NIL)
                                   (DBAR (HEAD NIL)
                                    (COMP (NP (SPEC NIL)
                                      (NBAR (HEAD DINNER)
                                       (NU (SG/PL))))))))
                              (IBAR  (HEAD (AGR 1SGPAST))
                               (COMP
                                (VP (SPEC NIL)
                                  (VBAR (HEAD BE)
                                     (COMP
                                      (SUBJECT_COMPLEMENT
                                       (DEGP (SPEC NIL)
                                         (DEGBAR (HEAD NIL)
                                          (COMP (AP (SPEC NIL)
                                        (ABAR (HEAD READY))))))))))))))))))))))))))))))
```

The woman announced to her guests that dinner was ready.

All the codes given in the OALD for that verb are included in a verb's lexical entry so that all structures that could possibly complement the main verb are described. This makes the grammar dependent on the OALD codes and consequently, the information in the lexicon is very specific to this project. The best scenario would be to use Roger Mitton's computerized version of the OALD (see reference in (Garside, et al., 1987), but other electronic dictionaries could be used as well. The verb subcategorization information given in the OALD could be derived from information in the electronic version of Longman's Dictionary of Contemporary English (LDOCE). A problem with using this dictionary is that the subcategorization information is not as accessible as in OALD because there is no formal coding like OALD uses. This can be overcome with an automatic translator that could convert the LDOCE codes, and then it would be possible to use this dictionary as the large dictionary this grammar requires. This procedure could be recreated with most any electronic dictionary or text as long as it includes codes or tags which are sufficiently descriptive.

4 LEXICAL ENTRIES

Each lexical entry has been implemented as a LISP symbol whose form is equivalent to the word itself. Each of these symbols has an associated property list which has the following schematic form:

(12) <CATEGORY X> <toggled features> <value features> <ROOT X>

Here, X is a variable which is equal to either the category or the root of the word, as appropriate. Toggled features are those whose value is either "t" or "nil" and include the verb subcategorization codes. Only the relevant features are included in a word's dictionary entry, meaning that the lexical entry will contain no features with the value "nil". Value features, the most important of which is "NUMBER", have a set value which is specific to that word. These are the features I mentioned previously as being engineered to accept macros as values, but they may also take LISP lists as values. The ROOT property is optional to allow for categories whose lexicalization is static (e.g., prepositions). The ROOT property is most required for verbs, but must also be used with other inflected words, such as comparative and superlative adjectives.

4.1 CATEGORY

As explained above, this system primarily uses the traditional category assigned to a word, with the possible exception of the category MODAL which is given to words like "would" and "could". The arguments to the CATEGORY property, along with reference to the appropriate Brown Corpus tags from which they could be derived, are given in Table 4.1 below.

Table 4.1 CATEGORY codes and sources

CODE	TRADITIONAL CATEGORY	BROWN CORPUS TAGS
N	Noun	NN, NNS, NP, NPS,NR, NRS
V	Verb	VB, VBD, VBG, VBN,VBZ
DET	Determiner	ABL, ABN, ABX, AT, WDT
PRO	Pronoun	PN, PN$, PP$, PP$$, PPL, PPLS, PPO, PPS, PPSS, WP$, WPO, WPS,
ADJ	Adjective	JJ, JJR, JJS, JJT, QLP, AP
PREP	Preposition	IN
CONJ	Conjunction	CC, CS
ADV	Adverb	QL, RB, RBR, RBT, RN, RP, WQL, WRB
MODAL	Modal	MD

To create a dictionary entry for a word, it must be assigned at least one category from this list. This is done with an association like “CATEGORY V” in the property list. A word could be assigned multiple categories by making the argument to the “CATEGORY” property a list containing the appropriate codes. For example, the word “garden”, which could be both a verb or noun, would have the category entry “CATEGORY (N V)”.

Each lexical entry must contain all the category codes and lexical features that apply to that word because the grammar only accesses the last word entry read into memory. This means that the lexical entry for “garden” will include the specifications for its noun sense (cf. Noun section below) and also the specifications for its verb sense (cf. Verb section below). It will have the following form:

(13) (setplist ‘garden ‘(CATEGORY (N V) COUNT t UNTENSED t I t IPR t
NUMBER (SG) PNCODE (X3SG) ROOT garden))[2]

Notice this entry follows the schematic given in (12): category names, valued features, toggled features, and lastly, root.

4.2 Toggled features

These are features that describe when a word can be used and with what other words: these represent the word’s subcategorization. A list of possible toggled features and the word categories they can be used with is given in (2). For further details about each feature refer to the section below on that feature.

Table 4.2 Category Features

CATEGORY	FEATURE
DET	PRE-DET, CENTRAL-DET, QUANT, ART, REL, WH, NP
ADJ	QUANT, CENTRAL-DET, PRE-DET, PRED
N	CARDINAL, MASS, COUNT, PROPER PRE-DET, MENSURAL
PRO	CENTRAL-DET, PROPER, POSS, DEM, PRE_DET, WH, REL
ADV	DEG, NEG, WH, REL
V	CNT, CNG, CNA, CNN, DNPR, DNF, CNI, DNN, DPRW, DNT, DPRF, DNW, LN, I, IPR, IP, INPR, IT, PRESPART, LA, UNTENSED, PASTPART, DPRT, TNP, TF, TNT, TG, TNG, TNI, TN, TNPR, TW, TT

2. The value of PNCODE in this entry is a macro which has the value of the following list: (1SG-PRES 2SGPRES 1PLPRES 2PLPRES 3PLPRES).

All of these features will be assigned the value “t” when they occur in a lexical entry, because they are only included when appropriate. The property assignment takes the form of “<FEATURE> t” (i.e.,“CENTRAL-DET t”). Just as with the categories, all features that apply to any sense or usage of the word must be included in a single lexical entry. This means that it is possible for noun and verb features to occur in the same lexical entry, as was the case in the “garden” entry of (13).

4.2.1 PRE-DET

This feature is used to specify that a word can occur in “pre-determiner” position, which means that it can occur in the environment “X the noun.” Examples of pre-determiners are “all, half, both, etc.” Also categorized as PRE-DET are multipliers like “twice, thrice, and once” and fractions like “one-third, one-half.” Pre-determiners correspond to the Brown corpus tags ABL, ABN, ABX. They can be adjectives, pronouns, or nouns.

4.2.2 CENTRAL-DET

This feature is used to specify words that occur in the central position in a determiner phrase. It includes articles and other words falling into the canonical designation “determiner”. In addition, this feature is appropriate for the possessive pronouns like “my” and “his,” demonstrative pronouns such as “that” and “this,” and quantifying adjectives like “every” and “some.” A word should receive this feature if it can occur in the environment “all X noun” as an instantiation of X. These words correspond to the Brown corpus tags AT, PP$, and DT, DTI, DTS, DTX.

4.2.3 ART

This feature further classifies a determiner as an article. It is typically assigned to the words “a”, “an”, and “the.” These words correspond directly to the Brown corpus tag AT.

4.2.4 REL

This feature is used to identify words that could be used to introduce restrictive relative clauses. Typically it is given to determiners like “which”, pronouns like “who” and “whose”, adverbs like “when”, and also the word “that.” It can be given to any word with the Brown corpus tag WDT. If a word can introduce a relative clause, replacing “X” in the following sentences, it should be given this feature:

(14) a. John X stole the mulberry cake...

b. The dog X ate my homework...

c. The time X I told you...

d. The woman X turtle I borrowed...

4.2.5 WH

This feature identifies determiners, adverbs, and pronouns that could introduce WH questions. Its distribution largely overlaps that of the REL feature, but is separated in order to account for words like “that” which are REL but not WH. Words that receive this tag, behave like the following:

(15) a. X stole the mulberry cake?

b. X dog ate the plants?

c. X ate the roots of the plant?

d. X did you discover the plant was missing?

It can be given to most words with the Brown corpus tags WDT, WP$, WPO, WPS, WQL, WRB.

4.2.6 NP

This code is to denote which determiners require noun phrases as complements. For example the determiner "the" can not stand alone as the subject of a sentence, but the determiner "that" can. Consider the sentences below:

(16) a. *The is going to the store.

b. *The is funny.

c. This is going to be difficult.

d. That is funny.

If a determiner can occur in sentences like (16c-d) it should be given the feature NP.

4.2.7 POSS

This feature further specifies a pronoun as a possessive form and is typically assigned to possessive pronouns like "his" or "her." It can also be assigned to words with the Brown corpus tags PP$, PP$$, WP$, or any other combination including a "$". If the word infers ownership, the word should receive this tag.

4.2.8 DEM

Specifies a demonstrative pronouns as such. It is given to words like “that” or “this” and some words with the Brown corpus tags DT, DTS. These words can occur in environments such as:

(17) a. X nasty old man...

b. So it was X boy that plucked Annie’s flowers!

c. X is ludicrous.

4.2.9 MENSURAL

This feature classifies nouns as quantifying units, for example “half”, “pair”, “dozen”, “bushel”, “mile”, etc. Classification must be done intuitively by considering what kinds of words can be used as a unit of measurement.

4.2.10 QUANT

This feature identifies adjectives as quantifiers. It includes cardinal numbers, ordinal numbers, and some specific adjectives like “many”, “few”, “much”, “little”, “several”, “every”, “some”, etc. If the word tells “how much” or “how many” then it should receive this feature. These words typically would have the Brown corpus tags AP, CD, OD, and some of QLP.

4.2.11 CARDINAL

This feature is used to separate the cardinal numbers from the quantifiers. It is given to numbers like “one”, “two”, and “thirty”. It corresponds directly to the Brown tag CD.

4.2.12 MASS

This feature is assigned to nouns which refer to substances, qualities, collections: objects which cannot be individuated or whose individual parts can not be counted. Examples of these nouns are “anger”, “money”, “oats”, “gasoline”, and “physics”. A useful test for a word like this is to ask “If part of this substance is removed, can I still call what is left by the same word?” For example, if there is a quantity of sugar, and some is removed, what is left can still be called “sugar”; it has not lost its character. In contrast, if the bread is taken away from a sandwich, what is left can not still be called a “sandwich” and so “sandwich” would not be considered a mass noun.

A syntactic test for mass nouns is whether it can occur without a determiner. If it is ungrammatical to say either “X sit” or “X sits” then it is likely that the word X is not a mass noun. These words are usually considered to be singular, although this is not always the case, such as the word “oats” (cf “oats sit” vs. *”oats sits”). In this case the word is given a “neutral” number signifying that it is both singular and plural (e.g., SG/PL). This feature corresponds to the OALD code U or to some words with the Brown Corpus tags NN, NR, NRS.

4.2.13 COUNT

This feature is for countable nouns, which contrast with the mass nouns such that if a noun is not a mass noun, it is probably a count noun.[3] Count nouns include words that

---

3. It is possible for a noun to be both a count and a mass noun. These are called “Dual” nouns and should be given the classification necessary for both interpretations (i.e., number specifications must reflect the dual character of the noun). This means these nouns will typically receive the NUMBER code “SG/PL”.

change form in the plural, such as “man/men”, and “boy/boys”, but they could also apply to words like “sheep” which can occur in the noun phrases “one X”, “two X”. Any word that can exist in this environment must be given the feature COUNT, but their number must be separately determined for each surface form. These words correspond to the OALD code C and some words with the Brown tags, NN, and NNS.

4.2.14 PROPER

This feature is for proper nouns. It corresponds to the Brown tags PPSS, PPS, PN, NP and NPS. This feature applies to the personal pronouns (as if they were a person’s name) as well as third person proper nouns (which are indeed a person’s name).

4.2.15 DEG

This feature identifies words, typically adverbs, that can serve as heads of a degree phrase. As discussed in Chapter 3, these include “as”, “so”, “far”, “too”, but can also include some words with the Brown corpus tag QL. Tests for these words are whether or not they can occur in any of the phrases “X many”, “X rich”, “X quickly”, or “X down the road”.

4.2.16 NEG

This feature is for negative words, such as “not”. It would also be necessary tag the “n’t” contraction with this feature.

4.2.17 UNTENSED

This feature is for infinitival verb forms, and indicates that the verb does not have a PNCODE. It directly corresponds to the Brown tag VB. If the word is of the same form that occurs in the verb phrase “to X to the store” then it is untensed.

4.2.18 PASTPART

PASTPART indicates that the lexical entry is the past participle of some verb form, given as the ROOT of the same lexical entry. It directly corresponds to the Brown tag VBN and some of VBD. This is the word form that occurs in X in the following verb phrase:

(18) a. John has/have X to the store.

b. The rock has/have X.

c. Tom has/have X before.

4.2.19 PRESPART

This feature indicates that the lexical entry is the present participle of the verb form in ROOT of that lexical entry. It is usually a gerund with an -ing ending and directly corresponds to the Brown tag VBG. The verb form must be able to serve as the X in the following examples:

(19) a. John is X.

b. The rock is X.

c. The dog could be X.

d. The man should have been X.

4.2.20 PRED

This feature is given to adjectives which can only occur in predicative position. For example words like “ablaze” which correspond to the OALD code “pred.” If the adjective can occur in the sentence “John is X.” but can not occur in the sentence “The X man went home.” or “The X rock didn’t move.” then it should receive this feature.

4.2.21 LA

Linking + Adjective

This is a verb subcategorization feature that denotes a linking verb that takes an adjective phrase predicate. The best example is the verb “is” in sentences like “The man is sleepy.” The sentence pattern underlying this code is [DP V ADJ], where the verb is intransitive and assigns the attribute of ADJ to the DP subject.

4.2.22 LN

Linking + Noun (i.e., determiner) phrase

This verb subcategorization feature is just like LA, except the predicate is a noun phrase such as “The man is a doctor.” The sentence pattern underlying this code is DP V DP, where the verb is not a transitive verb and the DP is not an object. Rather, the verb serves as a kind of “=” sign to assign a description to the subject.

4.2.23 I

Intransitive

This subcategorization is given to intransitive verbs. An example of this is the verb “cry” in the sentence “John cries.” No object is required with this verb, although there may be adjuncts, such as in the sentence “John cries profusely” or “John cries in the living room.”

4.2.24 IPR

Intransitive + PRepositional phrase

This is a code for intransitive verbs specifying that this verb can occur with a prepositional phrase adjunct. In general, most intransitive verbs can occur with prepositional phrase adjuncts, but this code is included for completeness. An example of a verb with both the I and IPR code is “cry” which is grammatical in either of the two sentences:

(20) a. John cried.

b. John cried in the living room.

4.2.25 IP

Intransitive + Particle

This is the same kind of code as described with the previous code; it indicates that a verb can occur with a particle (i.e., “up”, “off”, “away”). Not all intransitive verbs have this code though, exemplified in the ungrammatical sentence “*I like to garden up/away/off”. These particles are not closely associated with the meaning of the verb, for example, both of the following sentences have basically the same meaning:

(21) a.The birds chattered.

b.The birds chattered away.

4.2.26 INPR

Intransitive + Noun (i.e., determiner) phrase or PRepositional phrase

Like the previous I codes, this code specifies the kind of adjunct the Intransitive verb can occur with. This code indicates that the verb could take either a determiner phrase or a prepositional phrase and retain the same meaning. An example of this type of verb is “lasted” which is grammatical in either “The meeting lasted for a week” or “The meeting lasted a week.”

4.2.27 IT

Intransitive +To- infinitive

This Intransitive verb code is given to verbs that allow a to-infinitive (i.e., an exceptional clause with an unlexicalized subject) to follow them as an adjunct. The verb is still intransitive, so there is no object in sentences with these kinds of structures. An example of this kind of verb is “hesitated” in “John hesitated to phone home.” The exceptional clause can not be an object because “John hesitated” is also a grammatical sentence. Recall from chapter 3 that a verb subcategorizes for its objects, so that if a verb does have an object, it is necessary for completing the sense of the verb. The ungrammatical sentence “*Nathan killed.” is ungrammatical because it does not have the object required for the “killing” action.

4.2.28 TN

Transitive verb + Noun (i.e., determiner) phrase

This is the code given to the common [DP V DP] structure where the second DP is the verb object. The verb "hit" has this subcategorization, as revealed in the grammaticality of "John hit the baseball." The object position is required for this verb because it is transitive, and this code says the structure of that object will be a DP.

4.2.29 TNPR

Transitive + Noun (i.e., determiner) phrase + PRepositional phrase

This code is given to Transitive verbs that take a single object made up of a determiner phrase and prepositional phrase (i.e., a small clause of the form [DP PP]). An example of this kind of verb is shown in the sentence "Mary convinced the court of her innocence." This is in contrast to a sentence like "John saw the movie in the living room" because with "saw", the prepositional phrase "in the living room" is modifying the action of the verb (i.e., where the "seeing" took place). Conversely, with the "convinced" sentence, the prepositional phrase is more fundamental to the meaning of the sentence. The sentence "Mary convinced the court" is not a complete thought, whereas ""John saw the movie" is.

4.2.30 TNP

Transitive + Noun (i.e., determiner) phrase + Particle

This code is for Transitive verbs that take a single object made up of a determiner phrase and a particle (i.e., a small clause of the form [DP ADV]). An example of this is the verb "shook" in "The nurse shook the medicine up." Like the prepositional phrase in code TNPR, the particle in this code has a close association with the object of the sentence. It is therefore a different construction than would be for a sentence like "I was shaken up." where the particle is more like part of the meaning of the verb (cf. Frazier, 1991).

4.2.31 TF

Transitive + Finite "that" clause (i.e., CP)

This code is for Transitive verbs that take a CP as object. An example of this is the verb "believe" as in the sentence "John believes that the class will be cancelled." The ungrammaticality of "*John believes[4]" shows that the object is required to complete the meaning of this verb, and this code determines that the structure of that object will be a finite clause.

4.2.32 TW

Transitive + Wh-clause

This code is for Transitive verbs that take a wh-clause, or indirect question, as object. There are two forms of complements possible with these verbs, as in the sentences "John decided what we should do next." and "John decided what to do next." This is implemented as either a CP object or an EC object introduced by the WH word.

---

4. If this is a grammatical sentence it is because there is an object that is understood as part of the current discourse and so need not be lexicalized. Accounting for this goes beyond the scope of syntax.

4.2.33 TT

Transitive + To-infinitive

This code is for transitive verbs that can take an exceptional clause with an unlexicalized subject, so that the underlying pattern is [DP Transitive-verb [PRO to VP]]. An example of this code is in the sentence “Mary hates to drive in the city.” The subject of the exceptional clause is represented by “PRO”” but actually refers back to the subject of the sentence. Transformational grammar refers to this structure as either “Subject-to-Subject Raising” or “Subject-Controlled Equi” because both verbs in the sentence share the same subject.[5]

4.2.34 TNT

Transitive + Noun (i.e., determiner) phrase + To-infinitive

This code is like the code TT except here the subject of the exceptional clause is lexicalized. The underlying pattern is [DP Transitive-verb DP [PRO to VP]] where the PRO in this case refers to the object DP. For example, the sentence “John expected Mary to wait for him” could be paraphrased “John expected that Mary would wait for him” where Mary is the subject of “waiting”. The structure this code describes is what Transformational Grammar calls “Subject-to-Object Raising” because the subject of the embedded clause raises to be the object of the main clause.

5. Subject-to-Subject Raising and Equi sentences are believed to be derived differently. Here I am not attempting to explain that derivation, only to describe the resulting surface structure. Cf. chapter 5 for further discussion on this issue.

4.2.35 TG

Transitive + verb+ inG headed verb phrase

This code describes a structure similar to that for TT; however, the object in this case is a small clause with an unlexicalized subject and a gerundive verb form. The underlying pattern is [DP Transitive verb [TRACE V-ing]]. This is also an example of Subject-dominated sentences, because the subject of the V-ing is also subject of the main clause. The difference is that the verbal inflection is different because there is no INFL in the embedded clause. An example of verbs that take this structure is “enjoys,” which produces sentences like “John enjoys playing baseball.”

4.2.36 TNG

Transitive + Noun (i.e., determiner) phrase + verb + inG verb phrase

This code is similar to TNT in that it is an Subject-Raising structure where the subject of the embedded clauses “raises” to be the object of the main clause. Like TG, this code has a different verb inflection because there is no INFL. This makes the pattern underlying this structure [DP Transitive verb DP [TRACE V + ing phrase]] where the embedded clause is a gerundive small clause. Consider the sentence “The man spotted the children waving from the playground.” as an example.

4.2.37 TNI

Transitive + Noun (i.e., determiner) phrase + Infinitival verb phrase

This code produces a structure similar to that in TNG except the verb form is infinitival rather than gerundive. The pattern underlying this structure is virtually identical to that in TNG: [DP Transitive verb DP[TRACE V+0]. An example is seen in the sentence “We watched the men unpack the china.”

4.2.38 CNT

Complex-transitive + Noun (i.e., determiner) phrase + To-infinitive

As I mentioned earlier in this chapter, OALD calls verbs that can take two objects “complex-transitive.” What this really means is that the object of the sentence takes its own objects and therefore the sentences is, as transformational grammar describes it, “Object Controlled.” To understand the difference between an object controlled sentence like those produced by the code CNT and sentences with apparently the same surface structure but with the code TNT, consider the following:

(22) a. Jane forced the man to give up the money.

b. Jane promised the man to give up the money.

In sentence (22a), the subject of “giving up the money” is “the man”, whereas in sentence (22b) it is Jane who must give up the money. Thus, (22a) is the object-controlled, CNT sentence and (22b) is a subject-controlled TNT sentence.

The underlying structure of CNT sentences is identical to that of TNT sentences:

[DP Transitive-verb DP [PRO to VP]].

4.2.39 CNN

Complex-transitive +Noun (i.e., determiner) phrase + Noun (i.e., determiner) phrase

This code denotes another object-control structure, but there are no PRO or TRACE in the place of unlexicalized items. The complement structure is a small clause of the form [DP DP] where the first DP is the primary object of the sentence and the second DP modifies the first. An example of this sentence is "The court considered Smith a trustworthy witness."

4.2.40 CNA

Complex-transitive + Noun (i.e., determiner phrase) + Adjective phrase (i.e., degree phrase)

The structure described by this code is identical to that in CNN, except the small clause has the form [DP AP] (where AP is implemented as a DegP). An example of this kind of sentence is "The freezer kept the ice cold."

4.2.41 CNG

Complex-transitive +Noun (i.e., determiner) phrase + Gerundive verb phrase

Another control structure, the form underlying CNG complements is the small clause [DP VP + ing]. This pattern is characterized by the sentence "The policeman got the traffic moving."

4.2.42 CNI

Complex-transitive + Noun (i.e., determiner) phrase + infinitival verb phrase

This object-control structure has complements that are infinitival small clauses of the form [DP V+0]. Consider as an example the sentence "Mother won't let the children play in the road."

4.2.43 DNN

Di-transitive + Noun (i.e., determiner) phrase + Noun (i.e., determiner) phrase

Di-transitive verbs take both an object and an indirect object. The object is a variable structure but the form of the object is simply described: it is either a DP or a PP. Verbs with the code DNN have both their direct and indirect objects being determiner phrases. An example is the sentence "John taught the children French" where the first DP is the indirect object and the second is the object.

4.2.44 DNPR

Di-transitive + Noun (i.e., determiner) phrase + Prepositional phrase.

This is another form of di-transitive sentences in which the indirect object is a prepositional phrase occurring last and the object is the determiner phrase which immediately follows the verb. This is exemplified with the sentence "John taught French to the children."

4.2.45 DNF

Di-transitive + Noun (i.e., determiner) phrase) + Finite clause

For this complement code, the indirect object is a determiner phrase and the object is a finite clause. An example of this is the sentence "The leader told Paul that the job would be difficult."

4.2.46 DPRF

Di-transitive + PRepositional phrase + Finite clause

This code is just like DNF with the exception that the indirect object is a prepositional phrase instead of a determiner phrase. This kind of pattern shows up in sentences like "The President announced to the journalists that he was resuming his duties."

4.2.47 DNW

Di-transitive + Noun (i.e., determiner) phrase + Wh-clause

This code specifies that the indirect object, which occurs directly after the verb, is a determiner phrase and that the object is a Wh-clause. Recall in the description of the code TW that there are two possible variations of WH-clauses. Here they are implemented the same as there, so that the possible sentences are as follows: "The host reminded the guests where to put their luggage" or "The host reminded the guests to put their luggage away."

4.2.48 DPRW

Di-transitive + PRepositional phrase + Wh-clause

The pattern underlying this code is very much like that just described in DNW with the exception that the indirect object is a prepositional phrase rather than a determiner phrase. The indirect object still follows directly after the verb, as in the sentences "You should indicate to the team where they should assemble" or "You should indicate to the team where to assemble".

4.2.49 DNT

Di-Transitive + Noun (i.e., determiner) phrase + To-infinitive

The pattern specified here is for a determiner phrase indirect object and a direct object that is a to-infinitive, meaning an exceptional clause with an unlexicalized subject. The underlying structure is just like that in CNT and TNT: [DP Transitive-verb DP [PRO to VP], and indeed some words share more than one of these codes. The DNT sentences are differentiated from the CNT and TNT sentences with regard to the meaning of the verb and the semantic roles it discharges. Consider the sentences in (23):

(23) a. Jane wanted the man to give up the money. [CNT]

b. Jane expected the man to give up the money. [TNT]

c. Jane warned the man to give up the money. [DNT]

Di-transitive sentences can be understood as describing a transferal relationship between the object and direct object. In (23a-b) it is difficult to identify the goal and the transferring action, whereas in (23c) it is clearly a "warning." given to "the man". These verbs can also often be rephrased like "subject gave a indirect object to the object" where the indirect object is specifically described in the object position, but is generally described by the meaning of the verb. For example, in (23c), the indirect object is "to give up the money" and this was the specific warning given to the man.

4.2.50 DPRT

Di-transitive + PRepositional phrase + To-infinitive

The DPRT pattern specifies a prepositional phrase indirect object and a to-infinitive direct object. It is a structure similar to DNT in that the indirect object directly follows the verb; however the form of the direct object is a prepositional phrase. Consider the example "Fred signalled to the waiter to bring an extra napkin."

4.3 Value features

A value feature is one whose argument is a pre-determined value rather than simply a "t" or "nil" as was the case for toggled features. A typical feature assignment takes the form "<FEATURE> <VALUE>" (i.e. "NUMBER (SG)"). A chart of features, categories they apply to, and possible values is given below in Table 4.3.

Table 4.3 Value Features and Appropriate Categories

FEATURE	CATEGORY	VALUES
NUMBER	N, PRO, DET, ADJ	SG, PL, SG/PL
PNCODE	V	PRES, PAST, X3SG, 3SG, NONE, ANY
TAKES	DET	SGCT, PLCT, NONCT
ZONE	ADJ	1, 2, 3

Only the values for the TAKES and ZONE are specific to this project; the NUMBER and PNCODE values are macros. The NUMBER values are combinations of the word's person, number, and tense (cf. Chapter 3 on INFL). It would be possible to implement person, number, and tense separately or in a different way, but this would require revising some of the primitives used in the agreement tests associated with this grammar. With this said, I will proceed to discuss the details of the implementation of the numbering scheme used here.

4.3.1 NUMBER & PNCODE

The values assigned to both these features are a subset of the variables 1SGPRES, 1PLPRES, 2SGPRES, 2PLPRES, 3SGPRES, 3PLPRES, 1SGPAST, 1PLPAST, 2SGPAST, 2PLPAST, 3SGPAST, and 3PLPAST. The noun's number and person determines the first part of the code, and the verb's tense determines the last part, as described below:

4.3.1.1 Nouns

Number itself is strictly a binary feature: singular and plural. Nouns use this feature in conjunction with person to specify verb agreement. Person is a ternary feature having the values first, second, and third. First person is typically only for self-referring personal pronouns like "I", "me", or "we"; second person occurs with the personal pronouns "your", "y'all" or "you"; third person is for referring expressions like "John" or "her", but also includes pronouns like "he", "she", and "it" and nouns like "man", "dog", etc. Possible combinations for person and number are:

(23) 1SG, 2SG, 3SG, 1PL, 2PL, 3PL.

The 3SG and 3PL are used most often.

In order to simplify number classification, my implementation uses the macros "SG", "PL", and "SG/PL" to assign number values. Here, "SG" refers to all of 1SG, 2SG, 3SG; "PL" refers to all of 1PL, 2PL, 3PL; and "SG/PL" refers to all six combinations. The tense part of the codes is not relevant to nouns, but when a word has the person and number 3SG, for example, it is given both 3SGPRES and 3SGPAST in order to account for all the verbs that can agree with it.

4.3.1.2 Verbs

Noun-verb agreement depends on tense in addition to person and number. This is clearly shown when trying to classify the usage of a verb like "hit", which produces the following grammaticality judgements according to the tense of the verb:

(24)

a. *The man hit John [3SG, Present tense]

b. The man hit John [3SG, Past tense]

c.The men hit John [3PL, Present tense]

d.The men hit John [3PL, Past tense]

The motivation for including the tense in the values for NUMBER is to capture data like this. Using these values, it is easy to specify the number on a surface structure which behaves like "hit"; the NUMBER assignment in the lexical entry for "hit" is the following:[6]

(25) NUMBER ‘(3PLPRES, 1SGPAST, 1PLPAST, 2SGPAST, 2PLPAST, 3SGPAST, 3PLPAST)

To make it easier to assign these values to words, my system uses the following macros: “PRES” for all present tense codes; “PAST” for all past tense codes, “X3SG” for all present tense codes except 3SGPRES; “3SG” for 3SGPRES; “ANY” for all codes.

Note that the verb feature PNCODE is only applicable to inflected verb forms; infinitives are given the toggled feature UNTENSED to account for their lacking person and number inflection.

4.3.2 TAKES

This feature is assigned to central determiners to specify the kind of noun that a particular determiner can occur with. This is analogous to the verbal subcategorization codes, except it is not a toggled feature. Possible values to this feature are SGCT, PLCT, and NONCT. These are derived from the possible noun types (cf. COUNT, MASS, and PROPER below). These codes should be assigned to determiners according to the paradigm below:

(26) SGCT --> “X man” or “X dog”

PLCT --> “X men” or “X dogs”

NONCT --> “X fish”, “X sugar”, or “X music” or proper nouns

---

6. Another side-effect of implementing the noun and verb numbers with the same variables is that noun-verb agreement is now a trivial task that can be done with a simple LISP member function.

As is the case with the verb codes, all values which apply to a specific determiner must be included in the argument to the TAKES property. Thus, a typical value assignment for the determiner such as “the”, might be “TAKES (SGCT PLCT NONCT)”.

4.3.3 ZONE

This property allows adjectives to be categorized according to the preferred order of their occurrence (cf. Chapter 3). This categorization is based on the modification zones posited by Bache in (Bache, 1978). The first zone is reserved for adjectives which semantically define or specify rather than describe. Examples of this are words like “same”, “usual”, “whole”, and others that denote size, time or age. Syntactically these words never occur in predicate position and can not be compared or intensified with words like “very” or “extremely”:

(27) a. * The very usual steps...

b. * The extremely same smell...

They also can not be coordinated with other adjectives of any zone:

(28) a. * The same and smooth action...

b. *The whole and inexorable web...

A word should be assigned zone 1 if it is ungrammatical in the following structures:

(29) a. *The X and smooth noun.

b. * The extremely/very X noun.

c. *Noun phrase is X.

Zone 2 is given to central adjectives: those Quirk calls the “most adjectival” in (Quirk, 1985). These adjectives can be compared and coordinated like in (27) and (28) and can also occur as predicates. This means if the word is grammatical in the tests given in (29) it is a ZONE 2 adjective. Semantically these words describe or characterize and also include some present and past participles like “exciting”, “terrifying”, and “exhausted.”

My zone 4 is the same as Bache’s zone 3, which is for peripheral adjectives that occur closest to the noun in a noun phrase. They are similar to zone 1 adjectives in that they can be expected to fail the tests in (29): no coordination, comparison, intensification, or predication applies. There is one exception: when there is a lexicalized degree head in the degree phrase. In this case it is possible for these adjectives to serve as predicates, as in the following examples:

(30) a. John was too political.

b. The party was more theatrical today.

Semantically, zone 4 words classify the noun and are usually noun or verbal cognates, such as “political”, “social”, and -able/ -ible adjectives like “washable”, and “understandable”. Zone 4 also includes nationalities and the specific words “little, “old”, and “young”.

With the use of these zones, it is possible to explain the data that follows:

(32) a.very rich white American man

b. ?? very white rich American man

c. ??American very rich white man

d. ??rich American very white man

e. ??white very American rich man

f. ??very American white rich man

Bache would explain these data such that the adjective “very” is a zone 1 adjective, “rich” is a zone 2 adjective, and both “white” and “American” are zone 3 (i.e., what I am calling zone 4) adjectives. It is often the case; however, that color words like “white” must occur before the words that Bache describes as being in what I call zone 4. Consider:

(33) a. white American man

b. *American white man

Because of this I have created a separate zone code for color words: they are assigned zone 3. In this way my grammar captures the ordering preference that occurs among adjectives: “very” is zone 1, “rich” is zone 2, “white” is zone 3, and “American” is zone 4.

4.4 ROOT

The value of the ROOT property is the uninflected form of a word. For verbs, this means the infinitival form: the ROOT of “hitting” and “hits” is “hit”. The lexical entry for this word would therefore contain an assignment like in the following:

(34) ROOT hit

Comparative and superlative adjective forms are also considered to be inflected forms and will also have a ROOT which is the positive form of the adjective. The lexical entry for the word “greener” would include the property assignment:

(35) ROOT green

Words like nouns and determiners typically do not have the ROOT property in their lexical entry.

APPENDIX

Example Dictionary

```
; NUMBER

; The following are some shortcuts for use with coding noun and
; verb number:

; For NOUNS:

(defmacro SG ()
  (list ‘quote ‘(3SGPRES 3SGPAST)))

(defmacro PL ()
  (list ‘quote ‘(3PLPRES 3PLPAST)))

(defmacro SG/PL ()
  (list ‘quote (ANY)))

; For VERBS:

(defmacro PRES ()
  (list ‘quote ‘(1SGPRES 2SGPRES 3SGPRES 1PLPRES 2PLPRES 3PLPRES)))

(defmacro PAST ()
  ; defines the PNCODEs for PAST tense verbs. Also included is the
  ; uninflected PNCODE because the participal forms are usually
  ; made by adding -ed to the root of the word.  This may not be appropriate
  ; for strong verbs.
  (list ‘quote ‘(1SGPAST 2SGPAST 3SGPAST 1PLPAST 2PLPAST 3PLPAST NONE)))

(defmacro X3SG ()
  ; defines the PNCODEs for present tense verbs that are only
  ; inflected for 3rd person singular.  ALso included is the
  ; code for uninflected verbs since that is usually the same
  ; as the X3SG form.
  (list ‘quote ‘(1SGPRES 2SGPRES 1PLPRES 2PLPRES 3PLPRES NONE)))

(defmacro 3SG ()
  (list ‘quote ‘(3SGPRES)))

(defmacro NONE ()
  (list ‘quote ‘(NONE)))

(defmacro ANY ()
  (list ‘quote ‘(1SGPRES 2SGPRES 3SGPRES 1PLPRES 2PLPRES 3PLPRES 1SGPAST 2SGPAST 3SGPAST
        1PLPAST 2PLPAST 3PLPAST NONE)))


;DETERMINERS

;PRE DETERMINERS

(setplist ‘all ‘(CATEGORY (PRO DET) PRE-DET t NUMBER (PL)))
(setplist ‘both ‘(CATEGORY DET PRE-DET t NUMBER (PL)))
(setplist ‘half ‘(CATEGORY N MENSURAL t NUMBER (PL)))
(setplist ‘such ‘(CATEGORY DET PRE-DET t))
(setplist ‘double ‘(CATEGORY ADJ PRE-DET t ))
```

```
(setplist ‘twice ‘(CATEGORY ADJ PRE-DET t))
(setplist ‘one-third ‘(CATEGORY ADJ PRE-DET t))

;CENTRAL DETERMINERS

(setplist ‘the ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT PLCT NONCT) ART t NP t))
(setplist ‘a ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT) ART t NP t))
(setplist ‘an ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT) ART t NP t))

(setplist ‘my ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT PLCT NONCT) POSS t NP t))
(setplist ‘our ‘(CATEGORY (DET PRO) CENTRAL-DET t TAKES (SGCT PLCT NONCT) POSS t NP t))
(setplist ‘your ‘(CATEGORY (DET PRO) CENTRAL-DET t TAKES (SGCT PLCT NONCT) POSS t NP t))
(setplist ‘his ‘(CATEGORY (DET PRO) CENTRAL-DET t TAKES (SGCT PLCT NONCT) POSS t NP t))

(setplist ‘that ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT NONCT) DEM t))
(setplist ‘those ‘(CATEGORY DET CENTRAL-DET t TAKES (PLCT) DEM t))

(setplist ‘every ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT) QUANT t NP t))
(setplist ‘neither ‘(CATEGORY DET CENTRAL-DET t TAKES (SGCT) QUANT t))
(setplist ‘enough ‘(CATEGORY DET CENTRAL-DET t TAKES (PLCT NONCT) QUANT t))
(setplist ‘some ‘(CATEGORY DET CENTRAL-DET t TAKES (PLCT NONCT) QUANT t))

;QUANTIFIERS/CARDINALS (POST-DETERMINERS)

(setplist ‘many ‘(CATEGORY ADJ QUANT t NUMBER (PL)))
(setplist ‘much ‘(CATEGORY ADJ QUANT t NUMBER (SG)))
(setplist ‘several ‘(CATEGORY ADJ QUANT t NUMBER (PL)))
(setplist ‘few ‘(CATEGORY ADJ QUANT t NUMBER (PL)))
(setplist ‘little ‘(CATEGORY ADJ QUANT t ZONE (2) t))
(setplist ‘one ‘(CATEGORY N QUANT t  CARDINAL t TAKES (SGCT) NUMBER (SG)))
(setplist ‘two ‘(CATEGORY N CARDINAL t NUMBER (PL)))
(setplist ‘three ‘(CATEGORY N CARDINAL t NUMBER (PL)))

;NOUNS

; MASS

(setq MASS-standard ‘(CATEGORY N MASS t NUMBER (SG)))
(setplist ‘anger MASS-standard)
(setplist ‘chaos MASS-standard)
(setplist ‘courage MASS-standard)
(setplist ‘equipment MASS-standard)
(setplist ‘shopping MASS-standard)
(setplist ‘traffic MASS-standard)
(setplist ‘cash MASS-standard)
(setplist ‘conduct MASS-standard)
(setplist ‘education MASS-standard)
(setplist ‘harm MASS-standard)
(setplist ‘leisure MASS-standard)

; DUAL nouns

(setq DUAL-standard ‘(CATEGORY N MASS t COUNT t NUMBER (SG)))

(setplist ‘cake DUAL-standard)
(setplist ‘paper DUAL-standard)
(setplist ‘beauty DUAL-standard)
(setplist ‘difficulty DUAL-standard)
(setplist ‘experience DUAL-standard)
(setplist ‘light DUAL-standard)
(setplist ‘sound DUAL-standard)
```

```
(setplist ‘talk DUAL-standard)
(setplist ‘lamb DUAL-standard)
(setplist ‘time DUAL-standard)
(setplist ‘hair ‘(CATEGORY N MASS t COUNT t NUMBER (SG/PL)))

; COUNT

(setq SGCT-standard ‘(CATEGORY N COUNT t NUMBER (SG)))
(setq PLCT-standard ‘(CATEGORY N COUNT t NUMBER (PL)))
(setplist ‘man SGCT-standard)
(setplist ‘men PLCT-standard)
(setplist ‘forest SGCT-standard)
(setplist ‘king SGCT-standard)
(setplist ‘girl SGCT-standard)
(setplist ‘bus SGCT-standard)
(setplist ‘country SGCT-standard)
(setplist ‘countries PLCT-standard)
(setplist ‘woman SGCT-standard)
(setplist ‘women PLCT-standard)
(setplist ‘street SGCT-standard)
(setplist ‘mayor SGCT-standard)
(setplist ‘position SGCT-standard)
(setplist ‘grandma SGCT-standard)
(setplist ‘week ‘(CATEGORY N COUNT t NUMBER (SG) MENSURAL t))
(setplist ‘weeks ‘(CATEGORY N COUNT t NUMBER (PL) MENSURAL t))

;PROPER

(setq SGP-standard ‘(CATEGORY N PROPER t NUMBER (SG)))
(setq PLP-standard ‘(CATEGORY N PROPER t NUMBER (PL)))
(setplist ‘Colorado SGP-standard)
(setplist ‘Mars SGP-standard)
(setplist ‘Whitehall ‘(CATEGORY N PROPER t NUMBER (SG/PL)))
(setplist ‘Kremlin ‘(CATEGORY N PROPER t NUMBER (SG/PL)))

;PRONOUNS

(setplist ‘I ‘(CATEGORY PRO PROPER t NUMBER ‘(1SGPRES 1SGPAST)))
(setplist ‘you ‘(CATEGORY PRO PROPER t NUMBER ‘(2SGPRES 2PLPRES 2SGPAST 2PLPAST)))
(setplist ‘he ‘(CATEGORY PRO PROPER t NUMBER (SG)))
(setplist ‘she ‘(CATEGORY PRO PROPER t NUMBER (SG)))
(setplist ‘it ‘(CATEGORY PRO PROPER t NUMBER (SG)))
(setplist ‘we ‘(CATEGORY PRO PROPER t NUMBER ‘(1PLPRES 1PLPAST)))
(setplist ‘they ‘(CATEGORY PRO PROPER t NUMBER (PL)))
(setplist ‘her ‘(CATEGORY PRO PROPER t NUMBER (SG)))
(setplist ‘them ‘(CATEGORY PRO PROPER t NUMBER (PL)))

;PREPOSITIONS

(setq PREP-standard ‘(CATEGORY PREP))
(setplist ‘from PREP-standard)
(setplist ‘of PREP-standard)
(setplist ‘on PREP-standard)
(setplist ‘in PREP-standard)
(setplist ‘by PREP-standard)
(setplist ‘to PREP-standard)

;ADVERBS

(setq ADV-standard ‘(CATEGORY ADV))
(setplist ‘immediately ADV-standard)
```

```
(setplist ‘easily ADV-standard)
(setplist ‘away ADV-standard)

(setq DEG-standard ‘(CATEGORY ADV DEG t))
(setplist ‘how DEG-standard)
(setplist ‘as DEG-standard)
(setplist ‘that DEG-standard)
(setplist ‘so DEG-standard)
(setplist ‘too DEG-standard)
(setplist ‘more DEG-standard)
(setplist ‘less DEG-standard)
(setplist ‘most DEG-standard)
(setplist ‘least DEG-standard)

;CONJUNCTIONS

(setq CONJ-standard ‘(CATEGORY CONJ))
(setplist ‘that CONJ-standard)

;ADJECTIVES

; Zone 1

(setplist ‘sheer ‘(CATEGORY ADJ ZONE (1)))
(setplist ‘complete ‘(CATEGORY ADJ ZONE (1)))
(setplist ‘slight ‘(CATEGORY ADJ ZONE (1)))
(setplist ‘certain ‘(CATEGORY ADJ ZONE (1)))
(setplist ‘very ‘(CATEGORY ADJ ZONE (1)))

; Zone 2

(setplist ‘hungry ‘(CATEGORY ADJ ZONE (2)))
(setplist ‘stupid ‘(CATEGORY ADJ ZONE (2)))
(setplist ‘rich ‘(CATEGORY ADJ ZONE (2)))
(setplist ‘funny ‘(CATEGORY ADJ ZONE (2)))
(setplist ‘long ‘(CATEGORY ADJ ZONE (2)))
(setplist ‘guilty ‘(CATEGORY ADJ ZONE (2)))
(setplist ‘ablaze ‘(CATEGORY ADJ ZONE (2) PRED t))


; Zone 3

(setplist ‘green ‘(CATEGORY ADJ ZONE (3)))
(setplist ‘greener ‘(CATEGORY ADJ ZONE (3) DEGREE (COMPARATIVE) ROOT green))
(setplist ‘white ‘(CATEGORY ADJ ZONE (3)))
(setplist ‘sleeping ‘(CATEGORY (ADJ V) TNS PRESENT PRESPART t ZONE (2) ROOT sleep))

; Zone 4

(setplist ‘American ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘Gothic ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘political ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘annual ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘economic ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘medical ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘social ‘(CATEGORY ADJ ZONE (4)))
(setplist ‘rural ‘(CATEGORY ADJ ZONE (4)))

;NEGATIVES

(setplist ‘not ‘(CATEGORY ADV NEG t))
```

```
;VERBS

(setplist ‘is ‘(CATEGORY V PNCODE (3SG) ROOT be LA t LN t IPR t IP t))
(setplist ‘are ‘(CATEGORY V PNCODE ‘(2SGPRES 1PLPRES 2PLPRES 3PLPRES) ROOT be LA t LN t IPR t IP t))
(setplist ‘became ‘(CATEGORY V PNCODE (PAST) ROOT become LN t LA t TN t))
(setplist ‘become ‘(CATEGORY V PNCODE (append ‘(NONE) (PRES)) ROOT become UNTENSED t PASTPART t LA t
        LN t TN t))
(setplist ‘complain ‘(CATEGORY V PNCODE (X3SG) ROOT complain UNTENSED t IPR t I t TF t DPRF t))
(setplist ‘complaining ‘(ROOT complain PNCODE (NONE) PRESPART t IPR t I t TF t DPRF t))
(setplist ‘chatter ‘(CATEGORY V PNCODE (X3SG) ROOT chatter UNTENSED t IP t I t IPR t))
(setplist ‘chattered ‘(CATEGORY V PNCODE (PAST) ROOT chatter PASTPART t IP t I t IPR t))
(setplist ‘chattering ‘(ROOT complain PNCODE (NONE) PRESPART t IPR t I t IP t))
(setplist ‘last ‘(CATEGORY V PNCODE (X3SG) ROOT last UNTENSED t INPR t I t))
(setplist ‘lasted ‘(CATEGORY V ROOT last PNCODE (PAST) PASTPART t INPR t I t))
(setplist ‘hesitated ‘(CATEGORY V PNCODE (PAST) ROOT hesitate PASTPART t IT t I t IPR t))
(setplist ‘hesitate ‘(CATEGORY V PNCODE (X3SG) ROOT hesitate UNTENSED t IT t I t IPR t))
(setplist ‘phone ‘(CATEGORY V ROOT phone PNCODE (X3SG) UNTENSED t I t IP t TN t TNP t))
(setplist ‘open ‘(CATEGORY V PNCODE (X3SG) ROOT open UNTENSED t TN t I t IP t TNPR t IPR t TNP t))
(setplist ‘opened ‘(CATEGORY V PNCODE (PAST) ROOT open PASTPART t TN t I t IP t TNPR t IPR t TNP t))
(setplist ‘convince ‘(CATEGORY V PNCODE (X3SG) ROOT convince UNTENSED t TNPR t TN t DNF t CNT t))
(setplist ‘convinced ‘(ROOT convince PNCODE (PAST) PASTPART t TNPR t TN t DNF t CNT t))
(setplist ‘shook ‘(CATEGORY V PNCODE ‘(1SGPAST 2SGPAST 3SGPAST 1PLPAST 2PLPAST 3PLPAST) ROOT
        shake PASTPART t TNP t CNA t TN t I t LA t IPR t))
(setplist ‘shaken ‘(CATEGORY V PNCODE (NONE) PASTPART t ROOT shake))
(setplist ‘shake ‘(CATEGORY V PNCODE (X3SG) ROOT shake UNTENSED t TNP t CNA t TN t I t LA t IPR t))
(setplist ‘believe ‘(CATEGORY V PNCODE (X3SG) ROOT believe UNTENSED t TF t TN t TW t TNT t I t))
(setplist ‘decide ‘(CATEGORY V PNCODE (X3SG) ROOT decide UNTENSED t TF t I t IPR t TN t TNPR t TW t TT t))
(setplist ‘decided ‘(CATEGORY V PNCODE (PAST) ROOT decide PASTPART t TF t I t IPR t IN t TNPR t TW t TT t))
(setplist ‘hates ‘(CATEGORY V PNCODE (3SG) ROOT hate TT t TN t TNT t TG t TSG t))
(setplist ‘hate ‘(CATEGORY V PNCODE (X3SG) ROOT hate UNTENSED t TT t TN t TNT t TG t TSG t))
(setplist ‘expect ‘(CATEGORY V PNCODE (X3SG) ROOT expect UNTENSED t TT t TN t TNPR t TF t TNT t))
(setplist ‘enjoys ‘(CATEGORY V PNCODE (3SG) ROOT enjoy TG t TN t))
(setplist ‘enjoy ‘(CATEGORY V PNCODE (X3SG) ROOT enjoy UNTENSED t TG t TN t))
(setplist ‘playing ‘(CATEGORY V ROOT play PNCODE (NONE) PRESPART t TG t I t IPR t IP t TN t TNPR t DNN t CNN/
        A t TNP t DNPR t))
(setplist ‘play ‘(CATEGORY V ROOT play PNCODE (X3SG) UNTENSED t TG t I t IPR t IP t TN t TNPR t DNN t CNN/A t
        TNP t DNPR t))
(setplist ‘spotted ‘(CATEGORY V ROOT spot PNCODE (PAST) PASTPART t TNG t I t TN t TNPR t TW t CNN/A t IPR t))
(setplist ‘spot ‘(CATEGORY V ROOT spot PNCODE (X3SG) UNTENSED t TNG t I t TN t TNPR t TW t CNN/A t IPR t))
(setplist ‘watched ‘(CATEGORY V PNCODE (PAST) ROOT watch PASTPART t TNI t I t TN t TW t TNG t IPR t))
(setplist ‘watch ‘(CATEGORY V PNCODE (X3SG) ROOT watch UNTENSED t TNI t I t TN t TW t TNG t IPR t))
(setplist ‘keeps ‘(CATEGORY V PNCODE (3SG) ROOT keep CNA t LA t IPR t IP t TNPR t TNP t CNG t DNN t))
(setplist ‘keep ‘(CATEGORY V PNCODE (X3SG) ROOT keep UNTENSED t CNA t LA t IPR t IP t TNPR t TNP t CNG t
        DNN t))
(setplist ‘considered ‘(CATEGORY V PNCODE (PAST) ROOT consider PASTPART t CNN t TN t TNPR t TW t TG t CNA
        t CNT t CNN/A t TF t))
(setplist ‘consider ‘(CATEGORY V PNCODE (X3SG) ROOT consider UNTENSED t CNN t TN t TNPR t TW t TG t CNA t
        CNT t CNN/A t TF t))
(setplist ‘accept ‘(CATEGORY V PNCODE (X3SG) ROOT accept UNTENSED t CNN/A t TN t I t TF t TW t))
(setplist ‘accepted ‘(ROOT accept PNCODE (PAST) CATEGORY V PASTPART t CNN/A t TN t I t TF t TW t))
(setplist ‘forced ‘(CATEGORY V PNCODE (PAST) ROOT force PASTPART t CNT t TNPR t TNP t TN t CNA t))
(setplist ‘force ‘(CATEGORY V PNCODE (X3SG) ROOT force UNTENSED t CNT t TNPR t TNP t TN t CNA t))
(setplist ‘got ‘(CATEGORY V PNCODE (PAST) ROOT get PASTPART t CNG t TN t DNN t DNPR t TNPR t LA t CNA t
        CNT t TG t IT t IPR t IP t TNP t))
(setplist ‘get ‘(CATEGORY V PNCODE (X3SG) ROOT get UNTENSED t CNG t TN t DNN t DNPR t TNPR t LA t CNA t
        CNT t TG t IT t IPR t IP t TNP t))
(setplist ‘let ‘(CATEGORY V PNCODE (ANY) ROOT let UNTENSED t CNI t TNPR t TNP t TN t))
(setplist ‘lets ‘(CATEGORY V PNCODE (3SG) ROOT let CNI t TNPR t TNP t TN t))
(setplist ‘taught ‘(CATEGORY V PNCODE (PAST) ROOT teach PASTPART t CNN t I t DNT t TN t DNW t DNN t DNPR t
        TF t DNF t))
```

```
(setplist 'teach '(CATEGORY V PNCODE (X3SG) ROOT teach UNTENSED t CNN t I t DNT t TN t DNW t DNN t DNPR t
        TF t DNF t))
(setplist 'told '(CATEGORY V PNCODE (PAST) ROOT tell PASTPART t DNF t TN t DNN t DNPR t DNW t DNT t I t TF t
        TW t TNPR t IPR t))
(setplist 'tell '(CATEGORY V PNCODE (X3SG) ROOT tell UNTESED t DNF t TN t DNN t DNPR t DNW t DNT t I t TF t
        TW t TNPR t IPR t))
(setplist 'announced '(CATEGORY V PNCODE (PAST) ROOT announce PASTPART t DPRF t TN t TF t TW t DNPR t
        DPRW t))
(setplist 'announce '(CATEGORY V PNCODE (X3SG) ROOT announce UNTENSED t DPRF t TN t TF t TW t DNPR t
        DPRW t))
(setplist 'reminded '(CATEGORY V PNCODE (PAST) ROOT remind PASTPART t DNW t TN t DNF t DNT t TNPR t))
(setplist 'remind '(CATEGORY V PNCODE (X3SG) ROOT remind UNTENSED t DNW t TN t DNF t DNT t TNPR t ))
(setplist 'indicate '(CATEGORY V PNCODE (X3SG) ROOT indicate UNTENSED t DPRW t TN t TF t TW t DNPR t DPRF
        t I t))
(setplist 'warned '(CATEGORY V PNCODE (PAST) ROOT warn PASTPART t DNT t TN t TNPR t DNF t DNW t TNPR t))
(setplist 'warn '(CATEGORY V PNCODE (X3SG) ROOT warn UNTENSED t DNT t TN t TNPR t DNF t DNW t TNPR t))
(setplist 'signalled '(CATEGORY V PNCODE (PAST) ROOT signal PASTPART t DPRT t I t IPR t TN t TNPR t TW t TF t
        DNF t DPRF t DNW t DPRW t DNT t))
(setplist 'signal '(CATEGORY V PNCODE (X3SG) ROOT signal UNTENSED t DPRT t I t IPR t TN t TNPR t TW t TF t
        DNF t DPRF t DNW t DPRW t DNT t))
(setplist 'did '(CATEGORY V PNCODE '(1SGPAST 2SGPAST 3SGPAST 1PLPAST 2PLPAST 3PLPAST) ROOT do
        PASTPART t I t TN t DNN t DNPR t TG t IPR t))
(setplist 'do '(CATEGORY V PNCODE (X3SG) ROOT do UNTENSED t I t TN t DNN t DNPR t TG t IPR t))
(setplist 'hit '(CATEGORY V PNCODE (append (X3SG) (PAST)) ROOT hit UNTENSED t PASTPART t I t TN t DNN t
        TNPR t))
(setplist 'hits '(CATEGORY V PNCODE (3SG) ROOT hit I t TN t DNN t TNPR t))
(setplist 'hitting '(CATEGORY V ROOT hit PNCODE (NONE) PRESPART t I t TN t DNN t TNPR t))
(setplist 'took '(CATEGORY V PNCODE '(1SGPAST 2SGPAST 3SGPAST 1PLPAST 2PLPAST 3PLPAST) ROOT take
        PASTPART t TN t DNN t TNPR t TNP t CNG t DNPR t CNN t CNT t CNN/A t TG t))
(setplist 'take '(CATEGORY V PNCODE (X3SG) ROOT take UNTENSED t TN t DNN t TNPR t TNP t CNG t DNPR t
        CNN t CNT t CNN/A t TG t))
(setplist 'could '(CATEGORY (V MODAL) PNCODE '(1SGPAST 2SGPAST 3SGPAST 1PLPAST 2PLPAST 3PLPAST)
        ROOT can PASTPART t TN t))
(setplist 'can '(CATEGORY (V MODAL) PNCODE (PRES) ROOT can TN t))
(setplist 'would '(CATEGORY (V MODAL) PNCODE '(1SGPAST 2SGPAST 3SGPAST 1PLPAST 2PLPAST 3PLPAST)
        ROOT will UNTENSED t I t TN t TNT t TF t DNN t DNPR t))
(setplist 'will '(CATEGORY (V MODAL) PNCODE (PRES) ROOT will I t TN t TNT t TF t DNN t DNPR t))
(setplist 'have '(CATEGORY V PNCODE (X3SG) ROOT have UNTENSED t TN t TNPR t CNA t TNG t TNT t CNN/A t
        TNP t CNI t CNG t))
(setplist 'has '(CATEGORY V PNCODE (3SG) ROOT have TN t TNPR t CNA t TNG t TNT t CNN/A t TNP t CNI t CNG t))
(setplist 'been '(CATEGORY V PNCODE (PAST) PASTPART t ROOT be LA t LN t))
(setplist 'be '(CATEGORY V PNCODE (NONE) ROOT be UNTENSED t LA t LN t I t IPR t))

;+WH words

(setplist 'what '(CATEGORY (PRO DET) CENTRAL-DET t TAKES (SGCT PLCT NONCT) DEM t WH t NUMBER (SG/
        PL)))
(setplist 'which '(CATEGORY (PRO DET) CENTRAL-DET t TAKES (SGCT PLCT NONCT) DEM t WH t REL t NUMBER
        (SG/PL)))
(setplist 'who '(CATEGORY PRO WH t REL t NUMBER (SG/PL)))
(setplist 'whose '(CATEGORY PRO WH t REL t NUMBER (SG/PL)))
(setplist 'whom '(CATEGORY PRO WH t REL t NUMBER (SG/PL)))
(setplist 'that '(CATEGORY PRO REL t NUMBER (SG/PL)))
(setplist 'when '(CATEGORY ADV WH t NUMBER (SG/PL)))
(setplist 'how '(CATEGORY ADV WH t NUMBER (SG/PL)))
```

REFERENCES

Bache, C. (1978). *The Order of Premodifying Adjectives in Present-Day English.* Odense University Studies in English. vol. 3.

Cowie, A.P. (1989). *Oxford Advanced Learner's Dictionary of Current English,* Fourth Edition. Oxford: Oxford University Press.

Francis.W. & Kucera, H. (1982). *Frequency Analysis of English Usage: Lexicon and Grammar.* Boston: Houghton Mifflin Company.

Garside, R., Leech, G., & Sampson, G., eds. (1987). *The Computational Analysis of English.* London: Longman.

Keulen, F. (1986): *The Dutch Computer Corpus Pilot Project.* M.A. Thesis, University of Nijmegen.

Marcus, M.P., Santorini, B., & Magerman, D. (1990). *First Steps Toward an Annotated Database of American English.* Department of Computer and Information Science, Technical Report MS-CIS-90-46. Philadelphia, PA, University of Pennsylvania.

Quirk, R., et al. (1985). *A Comprehensive Grammar of the English Language.* London: Longman.

Van Dyke, J. (1991a). *Word Prediction for Disabled Users: Applying Natural Language Processing to Enhance Communication.* Honors BA Thesis, University of Delaware.

Van Dyke, J. (1991b). *An Annotated Test Suite for the X' Theory Grammar.* Technical Report. Center for Applied Science and Engineering, A.I. Dupont Institute.