Adding a Realistic Lexicon to SHARDS

Matthew Purver

November 5, 2002

Abstract

This paper describes the expansion of the SHARDS system by the addition of a sizeable (40,000 root forms) lexicon, the Oxford Advanced Learner's Dictionary of Current English. Details are given of the processing of the dictionary into a Prolog-compatible form, and the modification of the SHARDS grammar and parser. The impact on performance of the current system is shown to be negligible, although this may change if the grammar is expanded in the future.

1 Introduction

The SHARDS system (see Ginzburg et al., 2001) is a Question-Under-Discussion (QUD)-based ellipsis resolution system. It uses a version of HPSG as developed by Ginzburg and Sag (2000). To date, the lexicon has been manually specified within the SHARDS code and consists of about 30 words. Relation of morphological variants of the same word stem has also been specified manually.

As part of the ROSSINI project[1], the SHARDS system is intended to be used as an input processing module within a dialogue system, CLARIE. If this system is to be able to successfully cope with a broad range of user input, a larger lexicon is required.

In order to allow as much domain-independence as possible, a realistic lexicon of English was used: the computer-usable version (Mitton, 1992) of the Oxford Advanced Learner's Dictionary of Current English (OALD) (Hornby, 1974). The dictionary had to be converted into a suitable format, and the SHARDS grammar modified to allow integration of the two.

In this paper, section 2 explains the choice of lexicon, section 3 describes the format chosen and processing required, and section 4 describes the modifications made to the grammar. Section 5 then gives a description of the performance of the final system.

[1] ROSSINI is a project investigating the **R**ole **O**f **S**urface **S**tructural **IN**formation **I**n Dialogue.

2 Choice of Lexicon

2.1 Size Requirement

The current SHARDS lexicon was required only to demonstrate the functionality of the system's ellipsis resolution capability. As such its small size (∼30 words) and manually specified nature were not a problem. Any use within a dialogue system will require a significant change.

The dialogue system on which CLARIE is based, GoDiS (see Larsson et al., 2000), uses a relatively small domain-dependent lexicon consisting of around 100 keywords and phrases. As GoDiS's input processing is performed by keyword/phrase spotting (and direct relation of keywords to semantic concepts and dialogue moves), this small size does not pose a problem for processing of utterances: any out-of-vocabulary (OOV) words can simply be ignored. As long as the specified keywords give good coverage of the domain concepts, this is acceptable.

However, the use of a simple grammar in input processing poses a new problem (quite separate from coverage of domain concepts): OOV words prevent parsing of utterances. A large lexicon was therefore required to give reasonable coverage of common English words. This could be supplemented by a domain-specific lexicon to provide specific (less common) words for particular applications.

2.2 Information Requirement

In order for a lexicon to be used within any grammar, at least part-of-speech (PoS) data and verb subcategorisation data are required. Information on inflectional morphology is also required if the semantic content of the resulting parses is to be used sensibly in a dialogue system – word stem or root form must be identified to know that e.g. *likes* and *like* are forms of the same verb. Additional useful information could include: count/mass noun nature, gender, and lexical semantics.

2.3 Available Resources

A summary of some available computer-readable lexicon resources is given in table 1 below.

The Brown Corpus and Moby project lexica were rejected on grounds of insufficient information; COMLEX and LDOCE on grounds of cost. This left CELEX2 and the OALD – as the OALD is free and forms a substantial part of CELEX2, it was chosen.

Name	Available from	Size	Information	Cost
Brown	with Brill tagger	∼52k words	PoS only	free
CELEX2	LDC[a]	∼60k[b] roots	PoS, subcat, morph	US$150
COMLEX	LDC	∼38k roots	Lots	US$1,500
LDOCE[c]	Longman	∼80k roots	PoS, subcat, morph	> UK£1,000
Moby	Uni of Sheffield	∼230k words	PoS only	free
OALD	OTA[d]	∼41k roots	PoS, subcat, morph	free

Table 1: Available Resources

[a] Linguistic Data Consortium, `http://www.ldc.upenn.edu`

[b] 41k taken from OALD, 53k from LDOCE 1978 version

[c] Longman Dictionary of Contemporary English

[d] Oxford Text Archive, `http://ota.ahds.ac.uk`

3 Lexicon Processing

3.1 File Formats

The OALD is supplied in ASCII format as shown in listing 1 below. The different columns give the word entry, pronunciation, PoS/inflection/-number/count/mass/rarity data, syllable count and verb subcategorisation data. All information is encoded according to a specific OALD scheme.

```
shard          SAd          K6$           1
shards         SAdz         Kj$           1
share          Se@R         J2%,M6%       13A,6A,14,15B
```

Listing 1: Example OALD raw ASCII format

```
noun( 'shard', 'shards', count, _ ).
noun( 'share', 'shares', both, _ ).
verb( 'share', 'shares', 'sharing', 'shared', 'shared',
                              _, ['3A','6A','14','15B'] ).
```

Listing 2: Example OALD final Prolog format

The final format chosen is a Prolog-readable format as shown in listing 2 above. A definition of the format is given in listing 3. Entries are given as compound terms, with the functor defining PoS category. Arguments are PoS-dependent, but the first argument is always the root form of the word.

Where inflectional morphology is defined (for nouns, verbs and adjectives) the alternative forms of the same root are given as the next arguments. For nouns, the only inflected form given is the plural; for verbs, 3rd person singular, present participle, past and past participle; and for adjectives, comparative and superlative.

The remaining arguments depend on PoS category and reflect the information provided in the OALD. For example, nouns are given a syntactic type (count, mass, both count & mass, or proper) and a semantic class (person or location) where available. In a couple of cases, extra "dummy" arguments are provided for later use (semantics for determiners, conversational move type for interjections). In the case of pronouns, case and semantic class (person/non-person) are determined from the word form without the information being explicitly given by the OALD.

```
noun( RootForm, PluralForm, count/mass/both/proper, per/loc ).
verb( RootForm, 3SingForm, PresPartForm, PastForm, PastPartForm,
                             tran/intran/both, SubcatList ).
adj( RootForm, CompForm, SupForm, normal/attr/pred/affix ).
adv( Form, normal/whrel/whq ).
pron( Form, nom/acc/case, normal/whrel/whq, per/nper ).
det( Form, def/indef, Semantics ).
prep( Form , prep ).
conj( Form, conj ).
misc( Form, prefix/interj/partcl/unknown, Semantics ).
```

Listing 3: OALD final Prolog format definition

3.2 Processing

Processing from one format to the other is performed using a Perl script, asc2lex.perl. Entries are read in line by line, converted to a Prolog-friendly format (lower case, escape sequences for quote characters, quoted strings etc.) and the encoded information is used to generate PoS category, inflected forms and other information. This is then stored in a buffer (to allow irregular forms to be generated – see below) and written out in the new format at the end.

3.3 Inflectional Morphology and Irregular Forms

For regular forms, the OALD gives derivational rules which allow the inflected forms of words to be generated. In these cases, when the root form is encountered, the rules are used to generate all inflected forms (which are then written out in the new format).

However, while root forms with irregular morphology are flagged, and the inflected forms are given in the dictionary, no connection between root and inflected forms is made, so the correct form must be guessed in some way.

3.3.1 Verbs

When non-root (inflected) forms are encountered, these are stored in buffers for later use. When irregular root forms are encountered, they are given a flag to mark them as such. Once all words have been processed or stored, the list of verb root forms is re-processed: for each irregular root form, the closest-matching inflected forms are chosen and assigned.

The closest match is chosen on the basis of the longest matching prefix (with shorter words being preferred in the case of a tie). As this is by no means a foolproof method (`beds` is preferred to `is` as the closest-matching 3rd-person-singular form for `be`), a `*` flag is also written out with these forms, to allow manual checking and correction later.

3.3.2 Nouns

The same process could be followed for nouns, but greater accuracy was achieved by writing a new set of inflectional rules for forming plurals (e.g. `-eau` → `-eaux`, `-ouse` → `-ice`, `-in-law` → `-s-in-law`).

Again, as this method could not cover every form encountered, a `*` flag was written to allow for manual post-correction.

3.3.3 Adjectives

As few irregular adjectives were encountered, it was sufficient to guess the comparative and superlative forms by adding `-er`, `-est` and manually post-editing.

4 Grammar Modification

The HPSG grammar used in SHARDS defined lexical entries via ProFIT[2] templates corresponding to PoS categories. The templates themselves defined context-free grammar (CFG) rules, thus in effect defining CFG rules for each lexical entry (the ProFIT templates are expanded during compilation to give a full list of CFG rules in standard Prolog form). Each inflectional variant had to be given its own lexical entry, with its root form specified therein. An example of this scheme is shown in listing 4 below.

This approach is problematic for a large lexicon: during ProFIT compilation, the proliferation of CFG rules will create an extremely large grammar which is slow to compile and search during parsing. It also requires the lexicon to be specified in ProFIT format, which restricts us to use of this language, and has the side-effect of forcing recompilation whenever the grammar or HPSG type system is changed.

[2] ProFIT – **Pro**log with **F**eatures, **I**nheritance and **T**emplates – is a Prolog extension allowing easy coding of feature structures (see Erbach, 1995).

```
% templates for lexical entries
lex_noun(Word,Case,Rest,RN) :=
              ( @noun(Word,Case,Rest,RN) ---> [Word] ).
lex_intran(Word,VForm,Rel,RN) :=
              ( @intran(Word,VForm,Rel,RN) ---> [Word] ).
lex_tran(Word,VForm,Rel,RN) :=
              ( @tran(Word,VForm,Rel,RN) ---> [Word] ).
lex_tran(Word,VForm,Rel,RN) :=
              ( @slashtran(Word,VForm,Rel,RN) ---> [Word] ).

% lexical entries (using templates)
@lex_noun(book,_,<book_rel,<book).
@lex_intran(snores,<fin,<snore_rel,<snore).
@lex_intran(snore,<inf,<snore_rel,<snore).
@lex_tran(likes,<fin,<like_rel,<like).
@lex_tran(like,<inf,<like_rel,<like).
```

Listing 4: Original SHARDS lexical entries

Instead, the standard Prolog format for lexical entries described in the previous section can be used directly once certain modification have been made to the grammar. First, a morphological interface is needed, to take any inflected word form and associate with it the root form of the word, and any further useful information given by morphology or the lexicon (e.g. singular/plural and count/mass nature for nouns, tense for verbs). Second, the CFG rules for each PoS category must be converted to allow any word form of the correct category to be a (lexical) daughter. This is achieved by associating a separate Prolog goal with the CFG rule: this goal both checks PoS category and retrieves useful information, by calling the morphological interface. An example of the scheme is shown in listing 5 below.

The addition of this goal to the CFG rule necessitated a small change to the parser to ensure that the goal is checked when the chart is initialized.

In summary, these modifications reduce the number of CFG rules available to the parser (as rules are no longer created for each lexical entry), but instead require the parser to check a Prolog goal (the morphological interface) for each lexical rule. As the size of the lexicon increases, the number of CFG rules *does not increase*, although the search time associated with the morphological interface will.

5 System Performance

In its current state, parse time is effectively the same as that of the original system. There is a measurable difference: time for lexical lookup is

```
% grammar rules for each PoS
@noun( Word, _Case, <rest_rel, [Stem] )
              ---> [Word], {noun( Stem, Word, _Number )}.
@intran( Word, VForm, [Stem] )
              ---> [Word], {verb( Stem, Word, VForm, intran, _CL )}.
@tran( Word, VForm, [Stem] )
              ---> [Word], {verb( Stem, Word, VForm, tran, _CL )}.
@slashtran( Word, VForm, [Stem] )
              ---> [Word], {verb( Stem, Word, VForm, tran, _CL )}.

% morphological interface to lexicon
noun( Word, Word, <sing ) :-
              noun( Word, _Plural, MassCount, SemClass ).
noun( Stem, Word, <plur ) :-
              noun( Stem, Word, MassCount, SemClass ).

verb( Word, Word, <inf, Cat, CatList ) :-
              verb( Word, _, _, _, _, Cat, CatList ).
verb( Stem, Word, <pres, Cat, CatList ) :-
              verb( Stem, Word, _, _, _, Cat, CatList ).
verb( Stem, Word, <past, Cat, CatList ) :-
              verb( Stem, _, _, Word, _, Cat, CatList ).

% lexical entries - actually in separate lexicon file
noun( 'book', 'books', count, _ ).
verb( 'snore', 'snores', 'snoring', 'snored', 'snored',
              intran, ['2A','2C'] ).
verb( 'like', 'likes', 'liking', 'liked', 'liked',
              tran, ['6A','6D','7A','17','19B','19C','22'] ).
```

Listing 5: Equivalent new SHARDS lexical entries

increased by a factor of approximately 3 times (this varies with part of speech and position in the lexicon), but as the actual lookup time is only of the order of 10^{-5} seconds on a 1GHz PC, it is insignificant compared to the rest of the parse time.

However, this holds only for unambiguous sentences. One unavoidable effect of a large lexicon is to introduce increased lexical ambiguity, and this in turn will lead to parse ambiguity for some sentences. As the grammar is currently very restricted, it is difficult to tell what effect this will have: as the grammar is expanded in the future, the effect may become significant.

References

Gregor Erbach. Prolog with features, inheritance and templates. In *Proceedings of the Seventh Conference of the European Association for Computational Linguistics*, pages 180–187, 1995.

Jonathan Ginzburg, Howard Gregory, and Shalom Lappin. SHARDS: Fragment resolution in dialogue. In H. Bunt, I. van der Sluis, and E. Thijsse, editors, *Proceedings of the Fourth International Workshop on Computational Semantics (IWCS-4)*, pages 156–172. ITK, Tilburg University, Tilburg, 2001.

Jonathan Ginzburg and Ivan Sag. *Interrogative Investigations: the Form, Meaning and Use of English Interrogatives.* Number 123 in CSLI Lecture Notes. CSLI Publications, 2000.

Albert S. Hornby. *Oxford Advanced Learner's Dictionary of Current English.* Oxford University Press, third edition, 1974. With the assistance of Anthony P. Cowie and J. Windsor Lewis.

Staffan Larsson, Peter Ljunglöf, Robin Cooper, Elisabet Engdahl, and Stina Ericsson. GoDiS - an accommodating dialogue system. In *Proceedings of ANLP/NAACL-2000 Workshop on Conversational Systems*, 2000.

Roger Mitton. *Oxford Advanced Learner's Dictionary of Current English: expanded "computer usable" version.* Oxford Text Archive, 1992. URL `http://ota.ahds.ac.uk/`.