ProPOSEL: a human-oriented prosody and PoS English lexicon for machine learning and NLP

Claire Brierley
School of Games Computing & Creative Technologies
University of Bolton
Deane Road
BOLTON
BL3 5AB
cb5@bolton.ac.uk

Eric Atwell
School of Computing
University of Leeds
LEEDS
LS2 9JT
eric@comp.leeds.ac.uk

Abstract

ProPOSEL is a prosody and PoS English lexicon, purpose-built to integrate and leverage domain knowledge from several well-established lexical resources for machine learning and NLP applications. The lexicon of 104049 separate entries is in accessible text file format, is human and machine-readable, and is intended for open source distribution with the Natural Language ToolKit. It is therefore supported by Python software tools which transform ProPOSEL into a Python dictionary or associative array of linguistic concepts mapped to compound lookup keys. Users can also conduct searches on a subset of the lexicon and access entries by word class, phonetic transcription, syllable count and lexical stress pattern. ProPOSEL caters for a range of different cognitive aspects of the lexicon[©].

1 Introduction

ProPOSEL (Brierley and Atwell, 2008) is a prosody and part-of-speech (PoS) English lexicon which merges information from respected electronic dictionaries and databases, and which is purpose-built for linkage with corpora; for populating tokenized corpus text with a priori linguistic knowledge; for machine learning tasks which involve the prosodic-syntactic chunking of text; and for open source distribution with NLTK - the Python-based Natural Language Toolkit (Bird *et al*, 2007a).

A pronunciation lexicon like ProPOSEL is an integral part of the front-end natural language processing (NLP) module in a generic text-to-speech (TTS) synthesis system and constitutes a natural way of giving such a system phonetic, prosodic and morpho-syntactic insights into input text. For English, three such resources, originally developed for automatic speech recognition (ASR) and listing words and their phonetic transcriptions, are widely used: CELEX-2 (Baayen *et al*, 1996); PRONLEX (Kingsbury *et al*, 1997); and CMU, the Carnegie-Mellon Pronouncing Dictionary (Carnegie-Mellon University, 1998). The latter is used in Edinburgh's state of the art Festival speech synthesis system (Black *et al*, 1999) and is included as one of the datasets in NLTK.

The starting point for ProPOSEL is CUVPlus[1] (Pedler, 2002), a computer-usable and human-readable dictionary of inflected forms which uniquely identifies word class for each entry via C5 PoS tags, the syntactic annotation scheme used in the BNC or British National Corpus (Burnard, 2000). CUVPlus is an updated version of CUV2 (Mitton, 1992), an electronic dictionary in accessible text file format which in turn derives from the traditional paper-based Oxford Advanced Learner's Dictionary of Current English (Hornby, 1974).

Recently, lexica for thirteen world languages, including US-English, have been created via the European-funded LC-STAR project (Hartinkainen *et al*, 2003) to address the shortage of language resources in the form of wide coverage lexica with detailed morpho-syntactic information that meet the needs of ASR, TTS and speech-to-speech translation (SST) applications. The incorporation of C5 PoS-tags in CUVPlus provides this kind of detail and

[1] http://ota.ahds.ac.uk/textinfo/2469.html

distinguishes this lexicon from other paper-based and electronic English dictionaries, including CELEX-2, PRONLEX and CMU; it also facilitates linkage with machine-readable corpora like the BNC.

However, CUVPlus entries compact PoS variants for a given word form into a single field as in the following example where *burning* is classified as an adjective, a present participle and a noun in Table 1:

```
burning|AJ0:14,VVG:14,NN1:2|
```

Table 1: Sample from CUVPlus record structure showing PoS variants for the word form *burning*

An early operation during ProPOSEL build was therefore to introduce one-to-one mappings of word form to word class, as defined by C5, to facilitate their use as compound lookup keys when the lexicon is transformed into a Python dictionary or associative array (§4).

2 ProPOSEL: a repository of phonetic, syntactic and prosodic concepts

The current revised version of ProPOSEL[2] is a text file of 104049 separate entries, each comprising 15 pipe-separated fields arranged as follows:

(1) word form; (2) BNC C5 tag; (3) CUV2 capitalisation flag alert for word forms which start with a capital letter; (4) SAM-PA phonetic transcription; (5) CUV2 tag and frequency rating; (6) C5 tag and BNC frequency rating; (7) syllable count; (8) lexical stress pattern; (9) Penn Treebank tag(s); (10) default content or function word tag; (11) LOB tag(s); (12) C7 tag(s); (13) DISC stressed and syllabified phonetic transcription; (14) stressed and unstressed values mapped to DISC syllable transcriptions; (15) consonant-vowel [CV] pattern.

```
sunniest|AJS|0|'sVnIIst|Os%|AJS:0|3|100|JJS
|C|JJT|JJT|'sV-nI-Ist|'sV:1 nI:0 Ist:0|
[CV][CV][VCC]
```

Table 2: Example entry from ProPOSEL textfile

Table 2 shows an example entry showing all fields; subsequent illustrative examples include only a subset of fields. For an explanation of fields 3 to 7, the reader is referred to Pedler (2002) and Mitton (1992). A full account of ProPOSEL build is planned for a subsequent paper, where phonology fields in source lexica (CUVPlus, CELEX-2 and CMU) and new phonology fields in the prosody and PoS English lexicon will be discussed in detail. The rationale for fields displaying syllable count, lexical stress pattern and CFP status is summarised here in section 3.

Four major PoS tagging schemes have been included in ProPOSEL to facilitate linkage with several widely used speech corpora: C5 (field 2) with the BNC as mentioned; Penn Treebank (field 9) with Treebank-3 (Marcus *et al*, 1999); LOB (Johansson *et al,* 1986) (field 11) with MARSEC (Roach *et al*, 1993); and C7 (field 12) with the 2 million-word BNC Sampler Corpus. The lookup mechanism described in section 4 where a match is sought between (token, tag) tuples in incoming corpus text and ProPOSEL's compound dictionary keys, also in the form of (token, tag) tuples, is possible for all four syntactic annotation schemes represented in the lexicon.

3 Accessing the lexicon through sound, syllables and rhythmic structure

One field of particular significance for ProPOSEL's target application of prosodic phrase break prediction (§3) is field (8) for lexical stress patterns, symbolic representations of the rhythmic structure of word forms via a string of numbers. Thus the pattern for the word form *‚objec’tivity* - with secondary stress on the first syllable and primary stress on the third syllable - is `20100`. For some homographs, this lexical stress pattern can fluctuate depending on part-of-speech category and meaning. The wordform *present* is a case in point, as demonstrated by fields 1, 2, 4, 7, 8 and 10 for all its entries in ProPOSEL shown in Table 3:

```
present | AJ0 | ’preznt | 2 | 10 | C |
present | NN1 | ’preznt | 2 | 10 | C |
present | VVI | prI’zent | 2 | 01 | C |
present | VVB | prI’zent | 2 | 01 | C |
```

Table 3: Rhythmic structure for the homograph *present* is inverted when it functions as a verb

Two well established phonetic transcription schemes are also represented in ProPOSEL: the original SAM-PA transcriptions in field 4 and DISC stressed and syllabified transcriptions in fields 13 and 14 which, unlike SAM-PA and the International Phonetic Alphabet (IPA), use a single character to represent dipthongs: `/p8R/` for *pair*, for example.

[2] April 2008

Phonology fields in ProPOSEL constitute a range of access routes for users. As an illustration, a search for like candidates to the verb *obliterate* might focus on structure and sound: verbs of 4 syllables (fields 2 and 7), with vowel reduction on the *first* syllable (fields 8 or 14), and primary stress on the *second* syllable (again, a choice of fields as users may wish to use the SAM-PA phonetic transcriptions). This filter retrieves sixty-seven candidates - most but not all of them end in `/eIt/` - and includes one oddity among the examples in Table 4. Further examples of live filtered searches are presented in section 5.

```
('affiliate', "@'fIlIeIt")
('caparison', "k@'p&rIs@n")
('corroborate', "k@'r0b@reIt")
('manipulate', "m@'nIpjUleIt")
('originate', "@'rIdZIneIt")
('perpetuate', "p@'petSUeIt")
('subordinate', "s@'bOdIneIt")
('vociferate', "v@'sIf@reIt")
```

Table 4: Sample of 8 candidate verbs retrieved which share requested phonological features with the template verb: *obliterate*

4 ProPOSEL: domain knowledge for machine learning

As previously stated, the rationale for ProPOSEL was to integrate information from different dictionaries and databases into one lexicon, customised for language engineering tasks which involve the prosodic-syntactic chunking of text. One such task is automated phrase break prediction: the classification of junctures (whitespaces) between words in the input text as either breaks (the minority class) or non-breaks. Typically, the machine learner is trained on PoS-tagged and boundary-annotated text - the speech corpus or *gold standard* - and then tested on an unseen reference dataset, *minus* the boundary tags, from the same corpus. Finally, it is evaluated by counting how many of the original boundary locations have been recaptured or *predicted* by the model.

Phrase break classifiers have been trained on additional text-based features besides PoS tags. The CFP status of a token - is it a *content* word (e.g. nouns or adjectives) or *function* word (e.g. prepositions or articles) or *punctuation* mark? - has proved to be a very effective attribute in both deterministic and probabilistic models (Liberman and Church, 1992; Busser *et al*, 2001) and therefore, a default content-word/function-word tag is assigned to each entry in ProPOSEL in field (10). It is anticipated that further research will suggest modifications to this default status when the CFP attribute interacts with other text-based features.

Syllable counts - field (7) in ProPOSEL - have already been used successfully in phrase break models for English (Atterer and Klein, 2002). However, they assume uniformity in terms of duration of syllables whereas we know that in connected speech, an indefinite number of unstressed syllables are packed into the gap between one *stress pulse* (Mortimer, 1985) and another, English being a *stress-timed* language. A lexical stress pattern, where syllables are weighted 0, 1 or 2, has therefore been included in fields (8) and (14) for entries in ProPOSEL because of its potential as a classificatory feature in the machine learning task of phrase break prediction.

The thematic programme for PASCAL[3] in 2008 focuses on approaches to supplementing raw training data (e.g. the speech corpus) with a priori knowledge (e.g. the lexicon) to improve performance in machine learning. The prosody-syntax interface is notoriously complex. Planned research into the phrase break prediction task will attempt to incorporate a dictionary-derived feature such as lexical stress (field 8 in ProPOSEL) into a data-driven model to explore this interface more fully.

5 Implementing ProPOSEL as a Python dictionary

The Python programming language has a dictionary mapping object with entries in the form of (key, value) pairs. Each key must be unique and immutable (e.g. a string or tuple), while the values can be any type (e.g. a list). This data structure can be exploited by transforming ProPOSEL into a *live* Python dictionary, where the recommended access strategy is via compound keys (word form and C5 PoS tag) which uniquely identify each lexical entry. Thus, using a sample of 4 entries to represent ProPOSEL and version 0.8 of NLTK, we can use the code in Listing 1 (§next page) to convert this mini lexicon into the new formalism. The Python dictionary method returns an as yet unsorted dictionary, where the data structure itself is represented by

[3] Pattern Analysis, Statistical Modelling and Computational Learning research network http://www.cs.man.ac.uk/~neill/thematic08.html

squigs **{ }** and where ***key : value*** pairs are separated by a colon. Table 5 displays the output from Listing 1 (below), demonstrating how multiple values representing a series of linguistic observations on syllable count, lexical stress pattern and content/function word status have now been mapped to compound keys (cf. Bird *et al*, 2007b, chapter 6; Martelli *et al*, 2005 pp. 173-5).

```
{
('cascaded', 'VVD') : ['3', '010', 'C'],
('cascaded', 'VVN') : ['3', '010', 'C'],
('cascading', 'VVG') : ['3', '010', 'C'],
('cascading', 'AJ0') : ['3', '010', 'C']
}
```

Table 5: Output from Listing 1

```
from nltk.book import * # In NLTK 0.9, the import statement would be: import nltk, re, pprint
lexicon = """
cascaded|VVD|0|k&'skeIdId|Ic%,Id%|VVD:1|3|010|VBD|C|VVD|VBD
cascaded|VVN|0|k&'skeIdId|Ic%,Id%|VVN:0|3|010|VBN|C|VVN,VVNK|VBN
cascading|VVG|0|k&'skeIdIN|Ib%|VVG:1|3|010|VBG|C|VVG,VVGK|VBG
cascading|AJ0|0|k&'skeIdIN|Ib%|AJ0:0|3|010|JJ|C|JJ,JK|JJ,JJB,JNP
"""
lexicon = [line.split('|') for line in list(tokenize.line(lexicon))]
lexKeys = [(index[0], index[1]) for index in lexicon]
lexValues = [[index[6], index[7], index[9]] for index in lexicon]
proPOSEL = dict(zip(lexKeys, lexValues))
```

Listing 1: Code snippet using Python list comprehensions and built-ins to transform the prosody-PoS English Lexicon into an associative array

For linkage with corpora and for annotating a corpus with the prior knowledge of phonology contained in ProPOSEL, a match is sought between incoming corpus text in the familiar (token, tag) format and the dictionary keys (§Table 5). Thus intersection enables corpus text to accumulate additional values which have the potential to become features for machine learning tasks. This lookup mechanism is relatively straightforward for corpora tagged with C5, the basic tagset used in the BNC. For corpora tagged with alternative schemes (i.e. Penn, LOB, and C7), incoming tokens and tags can either be matched against word forms and PoS tokens in the corresponding tagset field in the lexicon, or C5 tags can be appended to each item in the input text such that lookup can proceed in the normal way.

6 Filtered searches and having fun with ProPOSEL

ProPOSEL will be supported by a tutorial, offering a range of Python software compatible with NLTK, to enable users to prepare the text file for NLP; to implement ProPOSEL as a Python dictionary; to cross-reference linguistic data in the lexicon and corpus text; and to customise searches via multiple criteria.

The previous section demonstrated how fine-grained grammatical distinctions in the PoS tag field(s) in ProPOSEL are integral to linkage with corpora. It also demonstrated how an electronic dictionary in the form of a simple text file can be reconceived and reconstituted as a computational data structure known as an associative memory or array. When ProPOSEL is thus transformed, filtered searches can be performed on the text itself.

Brierley and Atwell (ibid.) present automatic corpus annotations achieved via intersection of two parallel iterables: ProPOSEL's keys and a LOB-tagged corpus extract (this is a short extract of 153 tokens just for demonstration) which also carries equivalent C5 tags generated from the lexicon. A successful match between C5 tags in both lists results in a corpus sequence object where word tokens and syntactic annotations have now been complemented with prosodic information from selected fields in ProPOSEL, as in Table 6:

```
[["aren't",   'BER+XNOT',   'VBB+XX0',
  ['1', '1', 'CF', "'#nt:1"]]
```

Table 6: Entry index of length 3, with word token mapped to LOB and C5 tags plus syllable count, lexical stress pattern, CFP status and syllable-stress mapping

The corpus sequence object can now be queried. Suppose, for instance, we wanted to find all bi-syllabic prepositions and particles in this extract. By specifying part-of-speech and syllable count, we unearth just one candidate matching our search criteria, as shown in Table 7:

```
['between', 'IN', 'PRP', ['2', '01',
'F', "bI:0 'twin:1"]]
```

Table 7: There is one candidate in the 153 word extract which meets the condition: PoS equals preposition or particle and syllable count is 2

It is not always necessary to transform ProPOSEL into a Python dictionary, however. Users can also read in the lexicon textfile, apply Python's splitlines() method to process the text as a list of lines, and then apply the split() method, with the *pipe* field separator as argument, to tokenize each field. Listing 2 presents this much more succinctly:

```
lexicon = open(‘filepath’, ‘rU’).read()
lexicon = lexicon.splitlines()
lexicon = [line.split('|') for line in
lexicon]
```

Listing2: Reading in ProPOSEL as a nested structure

Users can then perform a search on a defined subset of the lexicon. For example, users may wish to retrieve all entries with seven syllables from the lexicon. As well as returning items like: *industrialisation*, *operating-theatre*, and *radioactivity*, Listing 3 discovers the rather intriguing *sir roger de coverley*!

```
for index in lexicon:
   if index[6] == '7': # look in the subset
      print index[0] # return word form(s)
```

Listing 3: Searching a subset of the lexicon

Another illustration would be finding words which rhyme. If we wanted to find all the words which rhyme with *corpus* in the lexicon, we could search field (4), for example, the SAM-PA phonetic transcriptions, for similar strings to /'kOp@s/. One way of doing this would be to compile a regular expression, substituting the metacharacter **.** for the '*c*' in *corpus* and then seek a match in the SAM-PA field[4]. We might also look for minimal pairs, replacing the phoneme /s/ with the phoneme /z/ as in /'.Op@z/. Retaining the apostrophe as diacritic for primary stress before the wildcard here imitates the lexical stress pattern for *corpus* and is part of the rhyme. It transpires there is only one candidate which rhymes with *corpus* in the lexicon and two half rhymes. Listing4 gives us *porpoise* /'pOp@s/ and then *paupers* /'pOp@z/ and *torpors* /'tOp@z/.

```
p1 = re.compile("'.Op@s")
p2 = re.compile("'.Op@z")
sampa = [index[3] for index in lexicon]
rhymes1 = p1.findall(' '.join(sampa))
rhymes2 = p2.findall(' '.join(sampa))
```

Listing 4: Using regular expressions to retrieve bi-syllabic words with primary stress on the first syllable that rhyme with *corpus*

7 Cognitive Aspects of the Lexicon

ProPOSEL and associated access tools are presented to the CogALex workshop audience to illustrate our approach to enhancing the structure, indexing and entry points of electronic dictionaries. As the Call for Papers notes, "Access strategies vary with the task (text understanding vs. text production) and the knowledge available at the moment of consultation (word, concept, sound). Unlike readers who look for meanings, writers start from them, searching for the corresponding words. While paper dictionaries are static, permitting only limited strategies for accessing information, their electronic counterparts promise dynamic, proactive search via multiple criteria (meaning, sound, related word) and via diverse access routes. … The goal of this workshop is to perform the groundwork for the next generation of electronic dictionaries, that is, to study the possibility of integrating the different resources …" ProPOSEL integrates a range of different resources, and enables a variety of access strategies, with consultation based on various combinations of partial syntactic and prosodic knowledge of the target words. It addresses the main themes of the workshop:

[4] Note that Python lists start at index 0, hence in Listing 4, the SAM-PA field is at position [3] in the inner list of tokenized list fields for each entry.

7.1 Conceptual input of a dictionary user

Human users of electronic dictionaries can start from partial concepts or patterns when they are generating a message or looking for a (target) word. Other papers in the workshop focus on semantic cues, such as conceptual primitives, semantically related words, some type of partial definition, something like *synsets* etc; but speakers/writers may also be searching for a word which matches syntactic, phonetic or prosodic partial patterns, for example seeking a matching rhythm or rhyme.

7.2 Access, navigation and search strategies

The Call for Papers notes that "we would like to be able to access entries by word form but also by meaning and sounds (syllables) …Even if input is given in an incomplete, imprecise or degraded form." Meaning is clearly the main focus of many lexicography researchers, but access by sound, rhythm, prosody, and also syntactic similarity may also prove useful complementary strategies for some users.

7.3 Indexing words and organizing the lexicon

Another key issue for discussion in the Call for Papers is robust yet flexible organization of lexical resources: "Indexing must robustly allow for multiple ways of navigation and access… ". By building on and integrating with Python and the NLTK Natural Language Tool Kit, ProPOSEL can be accessed by other NLP tools or via the standard Python interface for direct browsing and search. ProPOSEL is also a potential exemplar for lexical entry standardization. Many lexicographers focus on standardization of semantics or definitions, but standardization of syntactic, phonetic and prosodic information is also an issue. Our pragmatic approach is to integrate lexical entries from a range of resources into a standardized Python dictionary format.

7.4 NLP Applications

We initially developed ProPOSEL in the context of research in linking lexical, syntactic and prosodic markup in English corpus text, and specifically as a resource for prosodic phrase break prediction (Brierley and Atwell, 2007a,b,c). The software developed within the NLTK architecture has been able to utilize existing NLTK tools for PoS-tagging, phrase-chunking and partial parsing; in turn, other researchers in these fields may want to use the syntactic information in ProPOSEL in their future NLP applications, particularly in research which attempts to compare or map between alternative tagsets or labeling systems, eg (Nancarrow and Atwell 2007), Atwell and Roberts 2006), (Atwell et al 2000), (Teufel 1995).

8 Conclusions

The English lexicon presented in this paper, - a revised version to that reported in (Brierley and Atwell, 2008), - is an assembly of domain knowledge of phonology and syntax from several widely used lexical resources. Linkage with corpora is facilitated by the inclusion of four variant PoS tagging schemes in the lexicon and by re-thinking and reconstituting the lexicon as a Python dictionary or associative array. A successful match between (token, tag) pairings in input text and new linguistic annotations mapped to ProPOSEL's compound keys will in turn embed a priori knowledge from the lexicon in data-driven models derived from a corpus and enhance performance in machine learning. The lexicon is also *human-oriented* (de Schryver, 2003). ProPOSEL's software tools are compatible with NLTK and enable users to define and search a subset of the lexicon and access entries by word class, phonetic transcription, syllable count and rhythmic structure. ProPOSEL was initially developed as a language engineering resource for use in our own research, but in the process of development we have also addressed several more general issues relating to cognitive aspects of the lexicon: the partial patterns in the mind of a dictionary user; the need for access and search by sound, rhythm, prosody, and also syntactic similarity; robust and standardised organization of lexical entries from different sources; and ease of integration into NLP applications.

References

Atterer M., and E. Klein. 2002. Integrating Linguistic and Performance-Based Constraints for Assigning Phrase Breaks. In *Proceedings of Coling 2002*:29-35.

Atwell, E., G. Demetriou, J. Hughes, A. Schriffin, C. Souter, S. Wilcock. 2000. A comparative evaluation of modern English corpus grammatical annotation schemes. *ICAME Journal*, vol. 24, pp. 7-23.

Atwell, E. and A. Roberts. 2006. Combinatory hybrid elementary analysis of text. In Kurimo, M, Creutz, M & Lagus, K (editors) *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*. Venice.

Baayen, R. H., R. Piepenbrock, and L. Gulikers 1996. *CELEX2* Linguistic Data Consortium, Philadelphia

Bird, S., E. Loper, and E. Klein 2007a. *NLTK-lite 0.8 beta* [June 2007] Available online from: http://nltk.sourceforge.net/index.php/Main_Page (accessed: 21/06/07).

Bird, S., E. Klein, and E. Loper 2007b. *Natural Language Processing* Available online from: http://nltk.sourceforge.net/index.php/Book (accessed: 21/09/07).

Black A.W., P. Taylor, and R. Caley. 1999. *The Festival Speech Synthesis System: System Documentation Festival version 1.4* Available online from: http://www.cstr.ed.ac.uk/projects/festival/manual/festival_toc.html (Accessed: 07/03/08)

Brierley, C. and E. Atwell. 2007a. Corpus-based evaluation of prosodic phrase break prediction in: *Proceedings of Corpus Linguistics 2007*, Birmingham University.

Brierley, C. and E. Atwell. 2007b. An approach for detecting prosodic phrase boundaries in spoken English. *ACM Crossroads journal*, vol. 14.1.

Brierley, C. and E. Atwell. 2007c. Prosodic phrase break prediction: problems in the evaluation of models against a gold standard. *Traitement Automatique des Langues,* vol. 48.1.

Brierley, C. and E. Atwell. 2008 ProPOSEL: a Prosody and POS English Lexicon for Language Engineering. In *Proceedings of LREC'08 Language Resources and Evaluation Conference,* Marrakech, Morocco. May 2008.

Burnard, L. (ed.) 2000. *Reference Guide for the British National Corpus (World Edition)* Available online from: http://www.natcorp.ox.ac.uk/docs/userManual/ (accessed: 20/05/07).

Busser, B. W. Daelemans, and A. van den Bosch 2001. Predicting phrase breaks with memory-based learning. *4th ISCA Tutorial and Research Workshop on Speech Synthesis*. Edinburgh, 2001.

Carnegie-Mellon University 1998. *The CMU Pronouncing Dictionary (v. 0.6)* Available online from: http://www.speech.cs.cmu.edu/cgi-bin/cmudict (accessed: 21/06/07).

Hartinkainen, E., G. Maltese, A. Moreno, S. Shammass, U. Ziegenhain 2003. Large Lexica for Speech-to-Speech Translation: frm specification to creation. *EUROSPEECH-2003*:1529-1532.

Hornby, A.S. 1974. *Oxford Advanced Learner's Dictionary of Current English* (third edition) Oxford: Oxford University Press

Johansson, S; Atwell, E S; Garside, R; Leech, G. 1986. *The Tagged LOB Corpus - User Manual,* 160pp, Bergen, Norwegian Computing Centre for the Humanities.

Kingsbury, P., S. Strassel, C. McLemore, and R. MacIntyre 1997. *CALLHOME American English Lexicon (PRONLEX)* Linguistic Data Consortium, Philadelphia

Liberman, M.Y., and K.W. Church 1992. Text Analysis and Word Pronunciation in Text-to-Speech Synthesis. In Furui, S., and Sondhi, M.M., (eds.) *Advances in Speech Signal Processing* New York, Marcel Dekker, Inc.

Marcus, M.P., B. Santorini, M.A. Marcinkiewicz, and A. Taylor 1999. *TREEBANK-3* Linguistic Data Consortium, Philadelphia

Martelli, A., A. Martelli Ravenscroft, and D. Ascher 2005. *Python Cookbook* (second edition) Sebastopol: O'Reilly Media, Inc.

Mitton, R. 1992. *A description of a computer-usable dictionary file based on the Oxford Advanced Learner's Dictionary of Current English* Available online and accessed (22/03/08) from: http://comp.lin.msu.edu/stabler-notes/1850/ascii_0710-2.txt

Mortimer, C. 1985. *Elements of Pronunciation.* Cambridge: Cambridge University Press

Nancarrow, O. and E. Atwell. 2007.A comparative study of the tagging of adverbs in modern English corpora *Proceedings of Corpus Linguistics 2007.* Birmingham University.

Pedler, J. 2002. *CUVPlus* [Electronic Resource] Oxford Text Archive Available online from: http://ota.ahds.ac.uk/textinfo/2469.html (accessed: 21/06/07)

Roach P., G. Knowles, T. Varadi and S.C. Arnfield. 1993. Marsec: A machine-readable spoken English corpus *Journal of the International Phonetic Association*, vol. 23, no. 1, pp. 47—53.

Schryver, G. M. de. 2003. Lexicographers' Dreams in the Electronic-Dictionary Age. *International Journal of Lexicography* 2003 16(2):143-199

Teufel, S.. 1995. A support tool for tagset mapping. Proceedings *of SIGDAT 1995. Workshop in cooperation with EACL 95*, Dublin