

The Prime Machine: A user-friendly corpus tool for English language teaching and self-tutoring based on the Lexical Priming theory of language

Thesis submitted in accordance with the requirements of the University of Liverpool
for the degree of Doctor in Philosophy by Stephen Jeaco.

June 2015

Contents

List of Figures	7
List of Tables	15
List of Abbreviations	19
Abstract.....	21
List of conference presentations arising from this project.....	23
Acknowledgements.....	24
Chapter 1: Introduction	25
1.1 Research questions	26
1.2 Structure	26
Chapter 2: Survey of Students and Teachers.....	29
2.1. Background, teaching context and personal motivations	29
2.1.1 English study skills for International students	29
2.1.2 English teaching in China	30
2.1.3 Personal motivations	32
2.1.4 Language learner strategies and dictionary resources	33
2.1.5 Corpora as a source for additional examples	33
2.1.6 The internet and other technology to collect and share examples.....	35
2.1.7 Items in L2 with limited L1 congruence	36
2.1.8 Purpose of the current phase of this research	37
2.2. Method	37
2.2.1 Participants	37
2.2.1.1 The institutions	37
2.2.1.2 Teacher survey	38
2.2.1.3 Student survey and test	38
2.2.2 Materials	39
2.2.3 Procedure.....	40
2.3. Results and Analysis.....	40
2.3.1 Results of the teacher questionnaire.....	41
2.3.2 Results of the student test and questionnaire	46
2.4. Conclusion.....	53
2.4.1 Limitations.....	53
2.4.2 Implications for EAP teaching and support.....	53
2.4.3 Implications for design and scope of future tools	54

2.4.4 Lexical Priming	55
2.5 Summary	57
Chapter 3: Software architecture and data pathways.....	58
3.1 Fundamental design considerations	58
3.2 Corpus selection.....	63
3.3 Choice of programming languages	66
3.4 How do you store a corpus?	75
3.4.1 Store data as they come	75
3.4.2 Store data ready for output	76
3.4.3 Store as XML	76
3.4.4 Make the data as few as possible	77
3.5 The refactoring process	83
3.5.1 Corpus formats.....	83
3.5.2 <i>The Prime Machine</i> corpus refactoring application	89
3.5.3 Summary tables	93
3.6 Design of the user interface.....	95
3.6.1 Query syntax, auto-complete and spelling support.....	95
3.6.2 Helping users compare words.....	99
3.6.3 Helping users compare results from two corpora	102
3.7 Summary	103
Chapter 4: Collocation	105
4.1 Collocation and language teaching.....	105
4.2 Defining collocation	107
4.3 What do concordancers offer in terms of collocation?	111
4.4 Multi-word units of more than 2 words	115
4.5 Priorities for Collocation in <i>The Prime Machine</i>	118
4.6 Mutual Information measures used in <i>The Prime Machine</i>	120
4.7 A new approach for collocation	125
4.8 Uses of collocation results	136
4.8.1 Collocation clouds and tables	136
4.8.2 Collocations and concordance lines.....	140
4.8.3 Collocations and search query formulation.....	144
4.8.4 Collocations and indications of semantic association	147
4.8.5 Collocations and concordance line ranking and selection.....	153

Summary	172
Chapter 5: Further features of Lexical Priming	173
5.1 Measuring tendencies for text position	173
5.1.1 Headings and Position in Text	176
5.1.2 Making Position in Text more prominent in concordance lines	177
5.1.3 Position in text	190
5.1.4 Position in sentence	202
5.1.5 Icons to represent different positions	208
5.2 Colligation	209
5.2.1 Sentence Complexity	210
5.2.2 Modality	214
5.2.3 Voice	218
5.2.4 Sentence Charge	221
5.2.5 Definiteness/indefiniteness	223
5.2.6 Prepositions	226
5.2.8 Icons to represent colligations	229
5.3 Self-Repetition	231
5.4 Storage and retrieval of data	233
5.5 Ways in which the results are displayed	237
5.5.1 Hot Icons	237
5.5.2 Graphs	240
5.5.3 Table	245
5.6 Filter and Compare Modes	248
Summary	251
Chapter 6: Metadata, KeyTags and Key Associates	253
6.1 Showing category and citation information	253
6.2 Fitting labels into the database schema	261
6.3 Key words	265
6.3.1 What do concordancers offer in terms of key words?	265
6.3.2 Limits and cautions for Log-Likelihood Key Words	268
6.3.3 Priorities for keyness measures in <i>The Prime Machine</i>	277
6.4 KeyTags	277
6.4.1 Calculating KeyTags	278
6.4.2 Examples of KeyTags	280

6.4.3 Potential uses of KeyTag results in language learning.....	292
6.4.4 Potential uses of KeyTags as search queries.....	294
6.5 Key Associates.....	298
Summary.....	306
Chapter 7: Evaluation.....	307
7.1 Evaluation considerations as design features.....	308
7.1.1 Authentication and user settings.....	309
7.1.2 Stars.....	314
7.1.3 Exporting results.....	317
More about the Cards Tab.....	319
7.1.4 Logs.....	320
7.2 Method.....	322
7.2.1 Participants.....	322
7.2.2 Materials.....	322
7.2.3 Procedure.....	323
7.2.4 Summary.....	328
7.3 Results and Analysis.....	328
Summary.....	343
Chapter 8: Conclusion.....	345
8.1 Scenarios.....	345
8.1.1 Scenario for a teacher's use of <i>The Prime Machine</i>	345
8.1.2 Scenario for students' use of <i>The Prime Machine</i>	347
8.2 Implications for future software development.....	350
8.3 Implications for the software as a learning and teaching tool.....	354
8.4 Beyond language learning.....	361
Summary.....	362
Appendix 1 Two word collocation measures.....	364
Appendix 2: Collocation measures for more than two words.....	366
Appendix 3: List of source files.....	368
Note on the source code.....	368
File formats for the source code and other files.....	368
1. <i>Delphi</i> files.....	368
2. <i>SQL</i> scripts.....	369
3. Other files.....	369

List of source files for the Refactoring Application.....	369
Main <i>Delphi</i> application	369
Subfolder AdvGridDefaults	370
Subfolder Rules	370
Subfolder Lookup Tables.....	371
Subfolder Templates	371
List of source files for the Lexical Priming Script Generator	372
Main <i>Delphi</i> application	372
Tables of rules	372
List of source files for the database compression and processing script	373
Filenames and processing order for corpora with more than one category	373
Filenames and processing order for corpora with only one category	374
List of source files for the one time setup and other metadata updates	375
Scripts to set up corpus administration database	375
Scripts to set up links to various resources.....	375
Scripts to update citation data and metadata labels for specific corpora.....	376
Script to improve speed after rebooting server machine.....	376
List of source files for the Corpus Management Application	376
Main <i>Delphi</i> application	376
Text for help screens providing explanations	377
List of Source Files for the Server Application	377
Main <i>Delphi</i> application	377
Other files.....	378
List of source files for the Client Application	378
Core files for main <i>Delphi</i> application.....	378
Source code for other forms used in the <i>Delphi</i> application	378
Source code for threads used in the <i>Delphi</i> application	379
Source code for other units	380
Other files.....	381
Bibliography	382

List of Figures

Figure 1.1: The tabs across the top of the screen in <i>The Prime Machine</i> concordancer.....	26
Figure 2.1: Teachers' reported frequency of use of different resources.....	41
Figure 2.2: Reported use of any concordancer to produce examples.....	42
Figure 2.3: Reported use of different concordancers.....	42
Figure 2.4: Teacher perceptions of ease of use of concordancing software.....	43
Figure 2.5: Teacher perceptions of how easily good examples can be generated through intuition and reflection.....	43
Figure 2.6: Teacher perceptions of the importance of examples.....	44
Figure 2.7: Teacher perceptions of the main causes of student errors.....	45
Figure 2.8: Reported knowledge and correctness grouped by question type.....	46
Figure 2.9: First reference choice grouped by question type.....	47
Figure 2.10: First reference choice grouped by question type (Institution C only).....	48
Figure 2.11: Websites where this was the first choice grouped by question type.....	49
Figure 2.12: Reported frequency of use of different tools – students from Institution A.....	50
Figure 2.13: Reported frequency of use of different tools – students from Institution C.....	50
Figure 2.14: Best tools reported for different problems.....	51
Figure 3.1: Screenshot of hints and tips which appear while results are retrieved (top) and some other examples of hints and tips (below).....	68
Figure 3.2: Diagram showing the data pathways for the three tier concordancer.....	74
Figure 3.3: Simplified schema for ranks of items in the corpus database; note: “_other_columns_removed” indicates that some columns used in the full schema have been removed from this diagram to clarify the relationships being described.....	79
Figure 3.4: Examples of the use of the same symbol for apostrophes and quotes in <i>the Guardian</i> and <i>Financial Times</i> corpora.....	85
Figure 3.5: The main screen for <i>The Prime Machine</i> corpus refactoring application.....	90
Figure 3.6: Diagram showing the processes involved in transforming raw text files for a corpus into its final state.....	93

Figure 3.7: Screenshot showing auto-complete support for a query	96
Figure 3.8: Screenshot showing <i>SoundEx</i> suggestions for "consequence" in the <i>BNC: Academic sub-corpus</i>	98
Figure 3.9: Screenshot showing the frequency of "polymorph" in alternative corpora.	99
Figure 3.10: Screenshot showing prompts which appear for <i>consequence</i> in the <i>BNC: Academic sub-corpus</i>	102
Figure 3.11: The "Compare with another corpus" sub-tab on the main Search Tab.	103
Figure 4.1: Information provided for language learners about collocation on the Life Ring screen for the Collocations Tab.	110
Figure 4.2: Table structure for MI Collocations	122
Figure 4.3: MI Collocation cloud and MI Collocation table for the node <i>consequences</i> in the <i>BNC: Newspapers</i> sub-corpus, sorted by the Dice score	124
Figure 4.4: Table structure for Log-Likelihood Collocations	134
Figure 4.5: Table structure for Collocation Extensions	135
Figure 4.6: Log-likelihood Collocation Clouds and Tables in the <i>BNC: Academic</i> sub-corpus for the node <i>outcome</i>	137
Figure 4.7: Log-likelihood Collocation Clouds and Tables in the <i>Financial Times</i> corpus for the node <i>outcome</i>	138
Figure 4.8: Log-likelihood Collocation Clouds and Tables in the <i>Hindawi Biological Sciences</i> corpus for the node <i>outcome</i>	138
Figure 4.9: Log-likelihood Collocation Clouds and Tables in the <i>Hindawi Mathematics</i> corpus for the node <i>outcome</i>	139
Figure 4.10: Log-likelihood Collocation Clouds and Tables in the <i>Hindawi Computer Science</i> corpus for the node <i>outcome</i>	139
Figure 4.11: Context menu for clouds and collocation table; cloud showing data from the <i>BNC: Academic</i> sub-corpus for the node <i>outcome</i>	141
Figure 4.12: Collocation-based captions on cards; screenshot showing data from the <i>BNC: Academic</i> sub-corpus for the node <i>outcome</i>	143

Figure 4.13: Auto-Complete suggestions showing collocations and raw window search queries; showing data from the <i>BNC: Academic</i> sub-corpus for the query <i>outcome longterm</i>	147
Figure 4.14: Two collocation clouds displayed side by side for the nodes <i>consequences</i> and <i>results</i> from the <i>BNC: Newspapers</i> sub-corpus.	149
Figure 4.15: Table structure for emotional charging of words derived from their collocates	151
Figure 4.16: Collocation and emotion clouds for the nodes <i>consequences</i> (top) and <i>results</i> (bottom) from the <i>BNC: Newspapers</i> sub-corpus.	152
Figure 4.17: Examples of longer duplicate sentences in corpora; exact matches are instances picked up automatically for the sample sentence and “very similar instances” are where almost the exact wording and punctuation was used.	157
Figure 4.18: Concordance lines for <i>outcome</i> in the <i>BNC: Academic</i> sub-corpus, ranked by fixed random order.....	170
Figure 4.19: Concordance lines for <i>outcome</i> in the <i>BNC: Academic</i> sub-corpus, ranked by the GDEX-like method	170
Figure 4.20: Concordance lines for <i>outcome</i> in the <i>BNC: Academic</i> sub-corpus, ranked by Log-Likelihood Collocations and then by the Collier-like method.....	171
Figure 4.21: Concordance lines for <i>outcome</i> in the <i>BNC: Academic</i> sub-corpus, ranked by MI Collocations and then by the Collier-like method	171
Figure 4.22: Concordance lines for <i>outcome</i> in the <i>BNC: Academic</i> sub-corpus, ranked by text abridgement-based method	172
Figure 5.1: Early design for the project, showing simple visual components with incidental data from BAWE.	179
Figure 5.2: Designs for the Cards Tab (right), based on the navigation bars which appear as default on the <i>TAdvSmoothTileList</i> component (left).	181
Figure 5.3 The card template for the Cards Tab (left) and the full set of data fields used (right).	182
Figure 5.4: Cards of different heights on the Cards Tab with captions at the top and gentle yellow highlighting of the line containing the node; with incidental data from the <i>BNC</i> :	

<i>Academic</i> sub-corpus for a search on the node <i>pilot</i> . The currently selected card is shown with a yellow caption and border.	184
Figure 5.5: The same card from the Cards Tab, with no highlighting (left), dotted line highlighting (centre) and gentle yellow highlighting (right) of the node word, showing one concordance box from the <i>BNC: Academic</i> sub-corpus for <i>pilot</i>	185
Figure 5.6: Text alignment in different columns of the grid on the Lines Tab for a relatively narrow width (top) and a relatively wide width (bottom) with incidental data from the <i>BNC: Academic</i> sub-corpus for a search on the node <i>pilot</i> . The currently selected line is shown with a blue background.	187
Figure 5.7: The Lines Tab showing left and right contexts for the node extending beyond sentence boundaries, with concordance lines for the node <i>pilot</i> in the <i>BNC: Academic</i> sub-corpus.	188
Figure 5.8: The same display when contexts are reduced to only include the sentence containing the node.	189
Figure 5.9: Information provided on the Life Ring help screen for the <i>Text Title</i> submenu on the Graphs Tab.	194
Figure 5.10: Information provided on the Life Ring help screen for the <i>Heading</i> submenu on the Graphs Tab.	195
Figure 5.11: Information provided on the Life Ring help screen for the <i>Text Position</i> submenus on the Graphs Tab.	198
Figure 5.12: Information provided on the Life Ring help screen for the <i>Text Position</i> and <i>Paragraph Position</i> submenus on the Graphs Tab.	200
Figure 5.13: Information provided on the Life Ring help screen for the <i>Theme/Rheme</i> submenu on the Graphs Tab.	205
Figure 5.14: Information provided on the Life Ring help screen for the <i>Sentence Position</i> submenu on the Graphs Tab.	207
Figure 5.15: Information provided on the Life Ring help screen for the <i>Complexity</i> submenu on the Graphs Tab.	214
Figure 5.16: Information provided on the Life Ring help screen for the <i>Modals</i> submenu on the Graphs Tab.	217

Figure 5.17 Information provided on the Life Ring help screen for the <i>Voice</i> submenu on the Graphs Tab.....	220
Figure 5.18: Information provided on the Life Ring help screen for the <i>Polarity</i> submenu on the Graphs Tab.	222
Figure 5.19: Information provided on the Life Ring help screen for the <i>Articles and Possessives</i> submenu on the Graphs Tab.	225
Figure 5.20: Information provided on the Life Ring help screen for the <i>Prepositions</i> submenu on the Graphs Tab.	227
Figure 5.21: Information provided on the Life Ring help screen for the <i>Repetition</i> submenu on the Graphs Tab.	231
Figure 5.22: The database schema for the uncompressed tables for sentences and words (left), and for the compressed tables (right).	235
Figure 5.23: Schema for the <i>cb_primings</i> table.....	237
Figure 5.24: Lines Tab showing the card for the currently selected concordance line and the dock of icons for the node <i>pilot</i> in the <i>BNC: Academic</i> sub-corpus.	239
Figure 5.25: Enlarged icon showing positive evidence for a tendency to occur after indefinite articles. The hand icon represents the mouse cursor position.....	240
Figure 5.26: Graphs for the <i>Paragraph Position</i> submenu on the Graphs Tab for the node <i>pilot</i> in the <i>BNC: Academic</i> sub-corpus, with fixed random order (top) and the log-likelihood collocation and concordance bonding rank method (bottom).....	241
Figure 5.27: Graph display for compare mode for the <i>Voice</i> submenu on the Graphs Tab with results for <i>consequences</i> compared against <i>outcomes</i> from the <i>BNC: Academic</i> sub-corpus.	242
Figure 5.28: Different norm values for <i>Complexity</i> on the Graphs Tab, with results shown for the node <i>pilot</i> in the <i>BNC: Academic</i> sub-corpus (top) and the <i>BNC: Newspapers</i> sub-corpus (bottom).	244
Figure 5.29: Screenshot of the table of features for the nodes <i>consequences</i> and <i>outcomes</i> in the <i>BNC: Academic</i> sub-corpus.....	247
Figure 5.30: Checkboxes and filter buttons for one of the submenus on the Graphs Tab..	248

Figure 5.31: Compare mode for the node <i>consequences</i> in the <i>BNC: Academic</i> sub-corpus, filtered by <i>definite articles or possessives</i>	250
Figure 6.1: Citation information displayed at the top of a card from the <i>Hindawi Biological Sciences</i> corpus (left) and the complete <i>BNC</i> corpus (right), as shown on the Cards Tab.	255
Figure 6.2: Clipped screenshot from the Lines Tab, showing citation information for a concordance line from the <i>BNC</i> in a pop-up balloon.	255
Figure 6.3: Examples of SQL script update statements created from a spreadsheet.....	257
Figure 6.4: Clipped screenshot from the Refactoring Application, showing examples of tags and how these are to be handled as metadata in the database.....	262
Figure 6.5: Database schema for metadata, with arrows showing links through primary keys	263
Figure 6.6: Further pop-up information for a card from the <i>BNC</i>	264
Figure 6.7: Further pop-up information for a concordance line from the <i>Hindawi Biological Sciences</i> corpus.....	265
Figure 6.8: Measure for key words in <i>LexTutor</i>	267
Figure 6.9: Claims made by Gabrielatos and Marchi (2012).....	269
Figure 6.10: Details about the SiBol 93 and SiBol 05 corpora	273
Figure 6.11: Table structure for key tag data.....	279
Figure 6.12: Tag clouds and tables for <i>therefore</i> (top) and <i>thus</i> (bottom) in the <i>BNC: Complete</i>	281
Figure 6.13: Tag clouds for <i>goal</i> in the <i>BNC: Academic</i> sub-corpus (top) and the <i>BNC: Newspapers</i> sub-corpus (bottom).	282
Figure 6.14: Tag clouds and tables for <i>aim</i> (top) and <i>goal</i> (bottom) in the SpringerOpen corpus.	284
Figure 6.15: Tag clouds for <i>important</i> (top) and <i>significant</i> (bottom) in the <i>Hindawi Biological Sciences</i> corpus.....	286
Figure 6.16: Text and Producer clouds for <i>mm</i> in the <i>BNC: Spoken Corpus</i>	290
Figure 6.17: Text and Producer clouds for <i>mhm</i> in the <i>BNC: Spoken Corpus</i>	291

Figure 6.18: Text and Producer clouds for <i>hmm</i> in the <i>BNC: Spoken Corpus</i>	291
Figure 6.19: Producer cloud for <i>okay</i> in the <i>BNC: Spoken Corpus</i>	292
Figure 6.20: Information about KeyTags cache provided in <i>The Prime Machine User Manual</i> , Version 2.0, January 2015.....	293
Figure 6.21: The auto-complete functionality of the Tags sub-menu on the Search Tab, with incidental data from the <i>Hindawi Biological Sciences corpus</i>	296
Figure 6.22: Table structures for key words and key associates.	301
Figure 6.23: Key Associates for <i>pilot</i> in the <i>BNC: Newspapers sub-corpus</i> (top) and the <i>BNC: Academic sub-corpus</i> (bottom).	303
Figure 6.24: Information about the corpus and the major categories which is provided on the Corpus Info. Tab.	304
Figure 6.25: Key Associates for <i>drugs</i> in the <i>BNC: Newspapers sub-corpus</i> (top) and the <i>Hindawi Biological Sciences corpus</i> (bottom).	305
Figure 7.1: Information about the cache provided in <i>The Prime Machine User Manual</i> , Version 2.0, January 2015.....	310
Figure 7.2: Information about the “Sign In” process, provided in <i>The Prime Machine User Manual</i> , Version 2.0, January 2015.	313
Figure 7.3: Concordance cards with the star rating elements visible; with incidental data for the node <i>outcomes</i> in the <i>BNC: Academic</i> sub-corpus.....	315
Figure 7.4: Star rating elements on the right-hand side of each concordance line on the Lines Tab; with incidental data for the node <i>outcomes</i> in the <i>BNC: Academic</i> sub- corpus.	316
Figure 7.5: The search history screen with pins highlighted in blue for sets of results which the user wishes to bookmark and retain in the cache beyond the usual timeframe of 7 days.....	317
Figure 7.6: Export options available for cards.	319
Figure 7.7: Export options available for tables.	319
Figure 7.8: The design of the closing screen before the pilot (top) and after the pilot (bottom).....	326
Figure 7.9: Reported use of different resources.....	330

Figure 7.10: Judgements given by participants on the best resource for a variety of language issues..... 331

Figure 7.11: Attitudes to the importance of different aspects of language for good writing in English..... 333

Figure 7.12: Reported frequency of use during different stages of the writing task. 334

Figure 7.13: Evaluation of the usefulness of some of the main features of the software .. 335

List of Tables

Table 3.1: The format and refactoring required for various corpora	88
Table 3.2 The main steps in the refactoring process leading up to the importing of SQL dump files into the database server.....	91
Table 4.1: Collocation measures available in concordancing software	113
Table 4.2: Approximate Bayes Factors and Equation for BIC approximation.....	127
Table 4.3: Contingency table and formula for key words.....	128
Table 4.4: Contingency table for Log-likelihood Collocations for a specific set of slots.....	130
Table 4.5: Cut-off points for storage of collocations for high frequency items as node in the complete <i>BNC</i> based on frequency	131
Table 4.6: Contingency table for extending collocations.....	135
Table 4.7: Emotion-Linked Collocation Contingency Table	150
Table 4.8: Features of GDEX and their implementation in <i>The Prime Machine</i>	160
Table 4.9: Magnitude of the growth of operations required for concordance line comparisons.....	165
Table 5.1: Features of Lexical Priming on the Graphs Tab.....	175
Table 5.2: Contingency table for sentence level features	192
Table 5.3: Tendencies to occur in text titles in the <i>BNC: Newspapers</i> sub-corpus.	193
Table 5.4: Tendencies to be used (or not used) in paragraph headings in the <i>BNC: Academic</i> sub-corpus.	195
Table 5.5: Tendencies to be used in the first or last sentence of texts in the <i>Hindawi</i> <i>Computer Science</i> corpus.....	196
Table 5.6: Tendencies to be used in the first or last paragraph of texts in the <i>Hindawi</i> <i>Computer Science</i> corpus.....	197
Table 5.7: Tendencies to be used in the first or last sentence of paragraphs in the <i>Hindawi</i> <i>Biological Science</i> corpus.....	199
Table 5.8: Token counts for different frequency ranges for corpora used in the summary tables provided in this chapter.....	201

Table 5.9: Proportions of types at different frequency thresholds showing at least one tendency for use in sentences in particular positions in text.....	201
Table 5.10: <i>CLAWS</i> tags used to identify the beginning of “Rheme”	202
Table 5.11: Contingency table for word level features.....	203
Table 5.12: Tendencies to be used in Theme or Rheme in the <i>BNC: Academic</i> sub-corpus.	204
Table 5.13: Tendencies to be used in the first or last 20% of a sentence in the <i>BNC: Academic</i> sub-corpus.....	206
Table 5.14: Proportions of types at different frequency thresholds showing at least one tendency for use in particular positions in a sentence.....	207
Table 5.15: Icons representing features related to position	209
Table 5.16: <i>CLAWS</i> tags used to identify <i>complex</i> sentences.....	211
Table 5.17: Tendencies to be used in Complex or Simple sentences in the <i>BNC: Newspapers</i> sub-corpus.	211
Table 5.18: Tendencies to be used in Complex or Simple sentences in the <i>BNC: Academic</i> sub-corpus.	212
Table 5.19: <i>CLAWS</i> tags used to identify modal verbs.....	214
Table 5.20: Limitations on the modal groupings included in the software	215
Table 5.21: Tendencies to be used with three groups of modal verbs in the <i>BNC: Academic</i> sub-corpus.	216
Table 5.22: <i>CLAWS</i> tags used to identify passive voice	218
Table 5.23: Tendencies to be used (or not used) in passive voice sentences in the <i>BNC: Newspapers</i> sub-corpus.....	219
Table 5.24: <i>CLAWS</i> tags used to identify polarity	221
Table 5.25: Tendencies to be used in negative sentences in the <i>BNC: Academic</i> sub-corpus.	221
Table 5.26: Proportions of types at different frequency thresholds showing at least one tendency for use with the first set of features of colligation.	222
Table 5.27: <i>CLAWS</i> tags used to identify articles and possessives	223

Table 5.28: Tendencies to be used with two groups of articles and possessives in the <i>BNC</i> : <i>Academic</i> sub-corpus.....	224
Table 5.29: Tendencies to be used with two groups of articles and possessives in the <i>BNC</i> : <i>Newspapers</i> sub-corpus.....	225
Table 5.30: <i>CLAWS</i> tags used to identify prepositions	226
Table 5.31: Examples for tendencies to be used with (or without) prepositions in the <i>BNC</i> : <i>Academic</i> sub-corpus.....	226
Table 5.32: Examples for tendencies to be used with (or without) prepositions in the <i>BNC</i> : <i>Newspapers</i> sub-corpus.....	227
Table 5.33: Examples for tendencies of collocates of <i>of</i> to be used with prepositions in the <i>BNC</i> : <i>Academic</i> sub-corpus, along with the proportions accounted for by collocations containing <i>of</i>	228
Table 5.34: Proportions of types at different frequency thresholds showing at least one tendency for use with the second set of features of colligation.....	229
Table 5.35: Icons representing features related to colligation.....	230
Table 5.36: Proportions of types at different frequency thresholds showing a tendency for repetition of form or stem.....	232
Table 5.37: Icons representing a tendency for repetition	232
Table 5.38: Overview of automatic processing.....	234
Table 5.39: Table of some of the priming features for the node consequences in the <i>BNC</i> : <i>Academic</i> sub-corpus, with 100 results in the downloaded set.....	246
Table 5.40: Icons used to show filtering is active	251
Table 6.1: List of transformation rules related to citations for the <i>Financial Times</i> corpus	257
Table 6.2: Different research aims and different uses of keyness values	272
Table 6.3 Data from by Gabrielatos and Marchi (2012)	275
Table 6.4: KeyTags contingency table.....	279
Table 6.5: Raw data for <i>gosh</i> before and after re-labeling the metadata tags for readability	288

Table 6.6: Raw data for <i>sorry</i> before and after re-labeling the metadata tags for readability.	289
Table 6.7: Key words and collocations for <i>conclusions</i> in the <i>Hindawi Biological Sciences</i> corpus	298
Table 7.1: User actions which can be automatically logged by the software.....	321
Table 7.2: Logs showing the number of views and time spent on different tabs in the software.....	336
Table 7.3: Logs showing the number of view, time spent and the number of different users for the results tabs after the main input session.	337
Table 8.1: Summary of principles for evaluating CALL, quoted from Chapelle (2001, p. 52) but presented in a different order.....	355

List of Abbreviations

%DIFF	Formula to calculate percentage difference. Described in Chapter 6.
ASCII	A form of character encoding (American Standard Code for Information Interchange)
AWL	Academic Word List
BASE	The British Academic Spoken English Corpus
BAWE	The British Academic Written English corpus
BIC	Bayesian Information Criterion
BLLDs	Bilingualized learner dictionaries
BNC	The British National Corpus
C5	CLAWS Tag Set 5
C7	CLAWS Tag Set 7
CALL	Computer Aided Language Learning
CC-EDICT	A Chinese English Dictionary project.
CD-rom	Compact Disc Read Only Memory
CLAWS	CLAWS software (Constituent Likelihood Automatic Word-tagging System)
CSV	.CSV is Comma-Separated Values file format; used for storing database or spreadsheet data in a very simple structure
DDL	Data Driven Learning
DICE	Dice's similarity coefficient
DNS	Domain Name System
EAP	English for Academic Purposes
EFL	English as a Foreign Language
ESL	English as a Second Language
EXE	.EXE is an executable file (program) for Windows.
FT	The Financial Times
FTW	Formula teaching worth
GBP	British Pounds
GDEX	Good Dictionary Examples; concordance ranking system in <i>The Sketch Engine</i>
HTML	HyperText Markup Language (file formatting structure for web pages)
ICE	Interactive Communication Environment; the virtual learning environment used for learning and teaching at XJTLU
IELTS	The International English Language Testing System
INT	Columns to hold integer values in a database; these can be various lengths (e.g. TINYINT, MEDIUMINT, LONGINT))

IP	Internet Protocol (address)
ISP	Internet Service Provider
IT	Information Technology
JPEG	.JPG or .JPEG files used to store images
KWIC	Key Word in Context
LL	Log Likelihood coefficient
MCQ	Multiple choice question
MI	Mutual Information (score)
MI3	Cubic association ratio
MLDs	Monolingual learner dictionaries
MWU	Multi Word Unit
MySQL	<i>MySQL</i> relational database system
OS	Operating System
POS	Part-of-Speech
QR code	Quick Response Code; a kind of matrix barcode
regex	Regular Expressions (a text pattern matching system used in some programming languages)
RMB	Renminbi is the currency of the People's Republic of China
SFL	Systemic Functional Linguistics
SGML	The Standard Generalized Markup Language
SLA	Second Language Acquisition
SQL	Structured Query Language
TEI	Text Encoding Initiative
TMS	TMS Software; a software development company
UCREL	University Centre for Computer Corpus Research on Language, Lancaster University
URL	Uniform Resource Locator (a web address)
USB	Universal Serial Bus
WECCCL	Written English Corpus of Chinese Learners
XJTLU	Xi'an Jiaotong-Liverpool University
XLS	.XLS files are spreadsheet files saved in the <i>Microsoft Excel 97-2003</i> file format
XML	Extensible Markup Language

***The Prime Machine: A user-friendly corpus tool for English language teaching
and self-tutoring based on the Lexical Priming theory of language***

Stephen Jeaco

University of Liverpool

Abstract

This thesis presents the design and evaluation of a new concordancer called *The Prime Machine* which has been developed as an English language learning and teaching tool. The software has been designed to provide learners with a multitude of examples from corpus texts and additional information about the contextual environment in which words and combinations of words tend to occur.

The prevailing view of how language operates has been that grammar and lexis are separate systems and sentences can be constructed merely by choosing any syntactic structure and slotting in vocabulary. Over the last few decades, however, corpus linguistics has presented challenges to this view of language, drawing on evidence which can be found in the patterning of language choices in texts. Nevertheless, despite some reports of success from researchers in this area, only a limited number of teachers and learners of second language seem to make direct use of corpus software tools.

The desire to develop a new corpus tool grew out of professional experience as an English language teacher and manager in China. This thesis begins by introducing some background information about the role of English in international higher education and the language learning context in China, and then goes on to describe the software architecture and the process by which corpus texts are transformed from their raw state into rows of data in a sophisticated database to be accessed by the concordancer. It then introduces innovations including several aspects of the search screen interface, the concordance line display and the use of collocation data. The software provides a rich learning platform for language learners to independently look up and compare similar words, different word forms, different collocations and the same words across two corpora. Underpinning the design is a view of language which draws on Michael Hoey's theory of Lexical Priming. The

software is designed to make it possible to see tendencies of words and phrases which are not usually apparent in either dictionary examples or the output from other concordancing software.

The design features are considered from a pedagogical perspective, focusing on English for Academic Purposes and including important software design principles from Computer Aided Language Learning. Through a small evaluation involving undergraduate students, the software has been shown to have great potential as a tool for the writing process. It is believed that *The Prime Machine* will be a very useful corpus tool which, while simple to operate, provides a wealth of information for English language teaching and self-tutoring.

List of conference presentations arising from this project

Jeaco, S. (2011). Developing an intuitive screen design for a learner-centred concordancer. Presented at the Sixth International Corpus Linguistics Conference (CL2011), Birmingham, UK, 20-22 July 2011.

Jeaco, S. (2013). "Hold on a minute; where does it say that?" - Calculating key section headings and other metadata for words and phrases. Presented at the Seventh International Corpus Linguistics Conference (CL2013), UCREL, Lancaster, UK, 23-26 July 2013.

Jeaco, S. (2014). Making collocation clouds more transparent: towards an asymmetrical, position sensitive collocation measure for improved learning. Presented at the Second Asia Pacific Corpus Linguistics Conference, the Hong Kong Polytechnic University, Hong Kong, 7-8 March 2014.

Jeaco, S. (2014). Tell me what I'm missing: Helping language learners make useful comparisons through enhancing the features of concordancing software. Presented at the 11th Teaching and Language Corpora Conference, UCREL, Lancaster, UK, 20-23 July 2014.

Jeaco, S. (2015). "Can you give me a few pointers?" Helping learners notice and understand tendencies of words and phrases to occur in specific kinds of environment. To be presented at the Eighth International Corpus Linguistics Conference (CL2015), UCREL, Lancaster, UK, 21-24 July 2015.

Acknowledgements

I would like to express my gratitude to my wife for her untiring support throughout this work. The project began with a young daughter and a new born baby boy in the home, and the patience and kindness of the family has been truly wonderful. We've fitted family outings, games and stories around pressures of work and the thesis, and the children are to be appreciated for brimming with enthusiasm about anything and everything, for not growing up too quickly, and for putting up with their father's disappearances into his study.

I am also grateful to my Mum and Dad for their support – stretching back to the beginnings of my interest in programming, running through undergraduate and Masters studies and running through the work of this thesis.

As an inspiration for the whole project in writing his books and giving a talk at our University, and as an on-going inspiration through supervision, my sincere thanks are also owed to Michael Hoey. The ringing tone on *SKYPE* video calls will be forever associated in my mind with the perfect mixture of intensity and kindness which he brought to me through our online meetings. I also have fond memories of the precious face-to-face meetings snatched between other XJTLU business.

My thanks also go to colleagues and students at XJTLU.

Chapter 1: Introduction

While learners and teachers may be using more materials based on patterns from corpora, the impact of corpus technology on self-study and in the classroom has not been as great as the shift in the academic research or publishing fields. Indeed, it would seem that of the vast numbers of language teachers working around the world, only a relatively small number attempt to motivate learners to use concordancers, often finding that learning to navigate the user interfaces requires a deep understanding of linguistic jargon and that learners only experience a limited amount of success in being able to process snippets from authentic sentences which have been decontextualised.

This thesis presents details of the design and evaluation of a new concordancer called *The Prime Machine* which has been developed as a language learning and teaching tool. The software aims to make insights about language based on Hoey's theory of Lexical Priming (2005) accessible and rewarding. The software has been designed to provide a multitude of examples from corpus texts and additional information about the contextual environment in which words and combinations of words tend to occur. Hoey argues that priming is "the result of a speaker encountering evidence and generalising from it" (2005, p. 185), and also considers some of the challenges that learners of a foreign language face due to limited opportunities to encounter language data naturally, and also due to the severe limitations of wordlists and isolated grammar rules. As well as supporting aspects of language which have been focussed on by corpus linguists for many years, such as the role of collocation and the distinction between word forms, the claims within the theory of Lexical Priming also demand a wider appreciation of the kinds of information which are primed with a word (or a combination of words) when they are learned. Within the theory, there are also several areas such as semantic association and textual colligation which are often neglected in descriptions of language and in language learning resources and are also often neglected in language teaching. Therefore the theory of Lexical Priming presents further challenges for a corpus software developer. If position in the sentence and paragraph can give vital cues to a learner investigator, and if a word's inclination towards repetition is also important, learners need to be able to see a much wider context in a more natural way if they are to appreciate the power of a corpus as a testing ground and as a tool for writing. Although this project builds on the results of other studies where researchers have taken corpus software tools and considered how they could be used in language learning, it will be argued that the design of each aspect of the new concordancer

has focused first and foremost on how the most basic building blocks of the data structures and the user interface can support pedagogical priorities. Inspiration and methodological approaches have been drawn from other concordancing software, and some of the strengths and weaknesses of leading English language corpus tools will be considered, but the emphasis will be on language learning and teaching. The aim of this project was to build a learner-centred front-end to a corpus engine, which draws on data which has been processed to be rich in mark-ups and cross-referencing consistent with Lexical Priming theory, and which presents this evidence as an enriched learning resource for language learning purposes.

1.1 Research questions

The thesis is chiefly concerned with the following research questions:

1. How can existing sub-processes of corpus tools be adapted to mark and index the contextual environments according to some of the features of Lexical Priming for retrieval in a computer program?
2. How can pedagogical considerations drive the design of concordancing software, ensuring that it supports language learning activities and that it is intuitive to use?
3. To what extent can these methods provide language learners with examples that they find useful and provide them with insights about language usage which they find helpful?

1.2 Structure

This introduction concludes with an overview of the structure of the thesis.

Like many other software applications, one of the main visual components of *The Prime Machine* is a set of tabs which can be used to switch between different functions and different pages of results. Figure 1.1 shows the tabs which appear at the top of the screen.

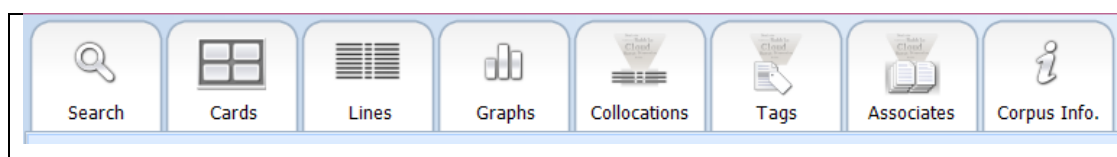


Figure 1.1: The tabs across the top of the screen in *The Prime Machine* concordancer.

Chapter 1: Introduction

In some ways, the structure of the thesis mirrors the structure of these tabs, with some of the tabs corresponding quite closely to specific pedagogical aims for specific linguistic features. However, just as it will be argued that the concordance lines are to be seen as central to the user's experience of the software, it is also the case that each of the chapters which deals with the design of the software also focusses on different aspects of the design of the two tabs which display concordance lines: the Cards Tab and the Lines Tab.

Chapter 2 provides background information behind the motivation for the development of the software, introducing the results of a survey of students and teachers in China. It begins with a brief consideration of the potential for corpus driven technologies in China within the context of the current educational system and the language learning expectations of English language learners. After presenting the results of a language test and questionnaire data, a summary of implications for language teaching and software development is provided.

Chapter 3 begins a series of chapters which provide details of decisions made in the design and development of the software. The focus of Chapter 3 is on the overall software architecture and the channels through which data can travel. The architectural decisions include aspects such as the choice of programming languages and the computer systems which need to be in place. The data pathways include information about the development of the refactoring process whereby raw texts are transformed into rows in the application's database. This chapter also includes an introduction to the motivation and design considerations regarding the search screen (the Search Tab), and the additional search support features which are provided to help language learners with aspects such as spelling, word form and collocation and several features designed to assist learners in making comparisons.

Chapter 4 continues by presenting the considerations behind the ways in which collocations are calculated, displayed and used in other processes within the software. It provides both a pedagogical background to collocation and an exploration of the statistical methods available for both two word collocations and multi-word units beyond this. As the phenomena of collocation is one of the main features of language described within the theory of Lexical Priming (Hoey, 2005), this chapter also discusses how the methods adopted for the calculation and presentation of collocations support the aims of making this kind of information available to language learners. As well as an introduction to the Collocations Tab, some of the further uses made of collocation data in the software are

Chapter 1: Introduction

explained in this chapter: their use in captions for concordance line display on the Cards Tab; their use in providing information about semantic associations; and their use in concordance line ranking.

Chapter 5 introduces some of the other kinds of contextual information and tendencies which are based on features of Lexical Priming and which are prominently displayed within the software. It provides a rationale for the Cards Tab and Lines Tab as two different ways of displaying concordance lines within the concordancer, as well as an overview of the mark-up and measurements which facilitate the visualization of summary information about various environments in the form of graph data on the Graphs Tab, and statistically significant priming tendencies in the form of icons. This includes an explanation of the concordance line filtering features which can filter results based on these priming environments, on nearby words or on a user's own star rating.

Chapter 6 provides details of how other information about texts in each corpus is used. It explains how metadata about the source of each text is provided within the visual design of the concordancer on the Cards Tab and the Lines Tab. It also presents a new application of the key word technique to provide insights into the ways in which words and collocations are used within each corpus according to metadata tags, providing the method and visual display of a technique named KeyTags. This chapter also explores some of the issues surrounding key words as a technique in corpus linguistics, and explains how text category summaries and key associates are calculated and presented in the software. It provides details about the Tags Tab, the Associates Tab and the Corpus Info. Tab.

Chapter 7 brings an end to the series of chapters looking at the design considerations by introducing features of the software which can be used to facilitate evaluation. It then provides the methodology and results of an evaluation of the software which drew on log data as well as questionnaires before and after groups of university students tried using the software to support them in a writing task.

Chapter 8 is devoted to a consideration of the implications of the evaluation in terms of further development of the software and in terms of longer term evaluation. The chapter brings the thesis to a close by exploring to what extent the design of the software tool and the evaluation of it which has already been completed meet the pedagogical aims of the overall project.

Chapter 2: Survey of Students and Teachers¹

This chapter presents results of a preliminary survey and test of students in Eastern China and their perceptions of existing reference tools. It begins with background information about the role of English in international higher education and the language learning context in China. Through analysis and discussion of the results, some specific areas of need in terms of linguistic knowledge are identified, and the results also show that search engines are surprisingly popular among students as a source of examples and linguistic information. These findings guided the development of the concordancer for language learners which forms the subject of this thesis. The chapter ends by arguing how the theory of Lexical Priming (Hoey, 2005) and the concordancing tool could be of special importance for developing and enhancing the language learning of students in this kind of context.

2.1. Background, teaching context and personal motivations

2.1.1 English study skills for International students

As the number of students studying degree subjects through the medium of English grows, it is becoming increasingly important to understand the language needs and learning strategies of these students as they use English as a tool for their academic careers. Andrade (2006) provides a review of studies into the international student experience and calls for further research and more targeted services. There has also been a call for a more detailed focus on measurable outcomes for international students in terms of their language skills and a better understanding of difficulties they face in achieving them (Y. Zhang & Mi, 2010). With the large numbers of Chinese international students enrolled in English-Speaking countries, some studies have focussed specifically on this group. For example, Zhang and Mi (2010) showed international students in Australia felt that they had coped well with many areas, but that Chinese learners in particular felt they needed more on-going support for academic writing. In another study, Chen and Duanmu (2010) found that at post-graduate level, the Chinese students who were interviewed struggled more with academic writing than their counterparts and were more likely to employ less active

¹ An early version of this chapter was privately produced for Xi'an Jiaotong-Liverpool University and is held by their research office.

study strategies. In addition, some researchers have looked at the socialization challenges that face mainland Chinese students in English-medium universities and highlighted the importance of research into understanding how they cope because of the importance of English language learning for the success of these students (Gao, 2010).

2.1.2 English teaching in China

It is clear that there are particular challenges for learners making the transition from high school to university in countries like China. The national curriculum for English in Chinese high schools is strongly influenced by the grammar-translation method, and emphasises a distinct separation of grammar and vocabulary. Although it is argued that Hallidayan linguistics is more widely accepted in China than alternative approaches which put grammar at the centre of language (G. Huang, 2002), recent textbooks for English teacher training programmes explain that the grammar-translation method² is still commonly used across China and that it has a strong place alongside trends towards the communicative approach or task-based learning (see Z. Li & Hao, 2009). Time constraints and a heavy emphasis on examination results mean that teachers have to balance innovation and adoption of other teaching methodologies against a view that the majority of time should be spent explaining English language using Chinese and working on grammar and translation activities (X. Zheng & Adamson, 2003). In brief, students entering university may have a strong foundation in some aspects of English grammar, but usually hold to a belief in the possibility of translating each item directly from their first language (L1) into their second language (L2). The widespread practice of memorizing words without any contextual information leads to problems with measuring students' long-term abilities in English (L. He & Qi, 2010). There have also been studies which have shown that weaker performing students tend to make the most use of memorization strategies, favouring these over other vocabulary learning strategies (Gu & Johnson, 1996).

The aims and guiding principles for English curricula in China are a subject of great discussion (Y. Zheng & Cheng, 2008). For example there are calls for changes to the national examination systems to make them fairer to all ethnic groups and to consider who really wants to learn English (Hu & Alsagoff, 2010). Others have argued that they need to be more responsive to the local context (Hu, 2005), and to embrace China English as an internationally recognised variety (D. He & Li, 2009). Two important areas of change are a

² The grammar-translation method of language teaching is characterised by the teaching of grammar rules with translation activities between the first and foreign language.

greater focus on learner autonomy and a greater emphasis on process over product (Ruan & Jacob, 2009). These areas can be explained as a greater focus on helping the students take more control over the pace and direction of their studies with a view to helping them to continue studying independently beyond the course and a greater emphasis being placed on the processes which they go through as they perform language activities rather than merely checking whether the final product matches expectations. A study into Chinese high school students' levels of motivation for learning English by Kyriacou and Zhu (2008) showed that motivation is relatively high and the students have a clear awareness of both the importance of getting good marks for the university entrance examination in English³ and the importance of English for international communication in the future. However, the study also showed that in actual fact the motivation levels were not as high as for other academic subjects.

Within this complex and changing English learning context, the opening up of China to international universities has brought with it a rather different focus for undergraduate English study, where English for Academic Purposes (EAP) provides the support students need as they make their transition from English as a core subject in high school to English as a tool for academic study at institutions where English is the medium of instruction. Zhang (2008) argues that international institutions and joint ventures provide many benefits to students including access to learning and teaching resources of an international standard, a reduced cost compared to studying overseas and greater opportunities to develop English and communicate with others on the international stage. Despite the fact that many institutions starting up in China face difficulties or fail, it is predicted that more of these types of institution will be established (Gide, Wu, & Wang, 2010). Furthermore, as well as the increase in international cooperative ventures as independent entities or colleges within established institutions, it is becoming increasingly common for higher level universities in China to provide courses which are wholly or partly delivered in English, drawing on international textbooks (F. Huang, 2006). While there may be a minimal move in this direction across the whole education system from early years through high school, Hu and Alsagoff (2010) have argued English medium primary and secondary schools are likely to experience too many practical problems to become very commonplace at this stage. However, at the university level, there are a growing number of students in China

3 The full set of University Entrance Examinations in China is called "Gaokao", but the papers and scoring systems vary from province to province.

who study in situations where their level of English acts as a filter to their overall educational experience and for whom effective EAP provision is essential.

2.1.3 Personal motivations

This thesis describes the development of a piece of language learning software, the idea for which grew out of my own experience in China as a teacher and manager of language teachers. At the time when I began work on this project, I had been interested in corpus linguistics for several years, but I had had limited success in passing on this enthusiasm to my students or colleagues. As the head of an English language centre from the time it was first established through to a staffing level of well over 100, I had had some success in encouraging staff from a wide variety of backgrounds to use computer and internet technologies to enhance their teaching, but I found little opportunity to integrate corpus linguistics into the syllabus or classroom sessions beyond using “corpus informed” textbook materials and dictionaries. Part of the problem was being able to find ways to systematically present convincing examples from corpora which learners could understand and appreciate. Another aspect of the problem was finding ways to introduce the functions of corpus software tools without needing to explain complicated procedures or difficult to grasp background information about the corpus linguistic theories underpinning the results.

Given the limited time available in class and a deep sense of the need to help my Chinese learners of English develop skills to explore language themselves, one of the main reasons for developing the concordancing tool was so that it could be an additional language resource to which my students could turn in order for them to check the meaning and use of words as they were composing, to consider alternative wordings as they were proof-reading and editing their own work or the work of a peer, and to explore in their own time some of the words and phrases which they had encountered briefly in a class session. The potential role of the concordancer I wanted to develop centred on the idea of it being a language resource which might be consulted by students to answer language questions related to their own linguistic and communicative needs. The role of any new language resource would need to be considered in terms of its ability to replace or complement other resources. It is because of these considerations that the survey of teachers and students which is presented in this chapter can be seen as having been one of the important foundations of the development of the software.

2.1.4 Language learner strategies and dictionary resources

An obvious strategy which learners can adopt when needing to write in a second language is to consult a reference resource to learn or check how a word or phrase can be used. Tickoo (1989) and Cowie (Cowie, 1999) provide extended reviews of studies looking at dictionary design and support for foreign language users at the end of the last two decades of the twentieth century. Since this time there have been many studies looking at the dictionary preferences of language learners, and several looking specifically at those of Chinese learners of English. Chan (2011) gave questionnaires to students in Hong Kong to look into perceptions and reported use of monolingual learner dictionaries (MLDs) compared to bilingualized learner dictionaries (BLLDs). Two thirds of respondents in the study said they used both MLDs and BLLDs. However, she found that learners had preconceptions about features they thought were not present in bilingualized dictionaries, and argues that ESL teachers should try to incorporate both types of dictionaries into classroom activities to help learners see the value of each. Her results also confirmed previous reflections by Rundell (1999) on learners' preferences for bilingualized dictionaries being based on the greater speed of access and ease of use these afford over monolingual dictionaries. Over 30 years ago, Nesi (1987) argued that dictionaries need to provide more examples showing how words are used if learners are to use them as a tool for productive use. Rundell (1999) looked specifically at a range of features which can help learners with productive use and found favourable results for monolingual dictionaries at the time of the study. However, it has been shown that English learner dictionaries have a very limited number of examples. Although a recent survey of learners in a general Chinese university setting found that students were reasonably satisfied with the number of examples in their dictionaries (Xu, 2009), this may not hold true for productive use in an EAP setting. In a study of the number of examples for words from three different levels of word frequency, Hai (2008) found that the most frequent words had on average 1.9 examples per sense, mid-frequency 1.65 and the third range 1.05. This study suggests that more examples are needed, particularly for words in the mid-frequency range.

2.1.5 Corpora as a source for additional examples

Corpus linguistics and concordancing software provide one possible answer to this need. Corpora have been used with learners in a number of different teaching situations; for a review of the use of corpora with learners see Yoon (2008) and Kennedy and Miceli (2010). There are several reasons highlighted in the literature which explain why concordancing software can be especially useful for learners. First, as Sinclair (1991) pointed out, if

learners want to learn about common patterns of syntax associated with a particular word dictionaries do not usually provide this. Secondly, as well as providing more information in an accessible way, it has been argued that concordancers give the learner an “ideal” space to test hypotheses (Kettemann, 1995; cited in Meyer, 2002). Studies have shown that teaching learners to use concordancers and then explore aspects of syntax by themselves can reduce their anxiety, and it has been suggested that this is because they can be freed from a sense of being subject to human judgement (Hunston, 2002). As well as providing the opportunity for learning about language use at the time concordancers are consulted, another advantage of teaching learners to use corpora is that it is a skill which can form part of their life-long learning (Mills, 1994). The procedures learners follow when they systematically perform searches and analyse corpus output help develop disciplines for self-access (G. D. Kennedy, 1998). In addition, for advanced learners and near expert speakers in the language teaching field, corpora empower further development of near native speaker abilities (Mair, 2002).

Although corpora have had an indirect influence on language teaching through the creation of dictionaries and materials which draw on corpus data, the main pedagogic implementation of corpus linguistics is Data Driven Learning (DDL). Johns (2002) listed several advantages of DDL over other types of learning materials, including new ways of approaching problem areas such as prepositions with a main focus on meaning and also helping teachers and learners prioritize what should be learned. As well as an approach to curriculum and materials development, Johns (2002) also found corpora helpful as a reference during tutorial feedback sessions with his learners. If an approach is taken where the learner is seen as a “traveller” rather than a “researcher”, Bernardini (2004) argues that concordancing tasks can be used as a means of meeting a variety of language teaching goals.

There have also been many studies into the use of corpora as a means for vocabulary building. Thurstun (1996) created materials for learners using lists of concordance lines, with a view to enabling them to recognise the common syntax of selected academic vocabulary and then use the terms for specific writing functions. Cobb (1999) used a concordancer as a means for students to develop their own personalised dictionaries, suggesting that new examples from a corpus could help students strengthen their knowledge of these words.

In addition to DDL and vocabulary building, corpora have also been used with learners as part of feedback on writing. Gaskell and Cobb (2004) explored the creation of hyperlinks to concordance lines for specific grammatical trouble spots based on their occurrence in the student essays and they report that learners found this kind of concordance use enjoyable. It has also been demonstrated that corpus-based activities can have an effect on the appropriateness of moves for a specific genre. Henry (2007) used corpus tools to build a set of online learning materials and to analyse the writing of learners prior and after their use.

However, despite some success, only a limited number of teachers and learners of second language seem to make regular use of these tools. Factors which may be holding teachers back from learning to use and teach corpus tools include issues with the context, the level of detail, interpretation, the time required to get results and software design. Traditional Key Word in Context (KWIC) concordance output is almost completely cut away from its context (Hunston, 2002). Also, the amount of detail which concordances can provide to a learner can be confusing (G. D. Kennedy, 1998). However, Varley (2009) reports some success for students if they can cope with the “overwhelming” amount of corpus data. Another point is that beyond dealing with the amount of raw data, the skills required to actually interpret them in order to understand grammatical patterns are far from simple (Gaskell & Cobb, 2004). Providing specific training about the limitations of corpus sources and a focus on patterns in more linguistically accessible examples within a larger sample has been shown to be one way forward (C. Kennedy & Miceli, 2010). However, from the students' perspective, exploration using carefully selected concordance lines may seem to take too long (Thurstun, 1996). In addition to these issues, as Anthony (2004) argues as he presents his classroom concordancer (*AntConc*), software for concordance exploration is not usually designed specifically with learners in mind. It is true that *AntConc* goes some way towards simplifying the interface of a concordancer, but there are still many obstacles to getting started and knowing enough about the tools and functions in order to use them. Effort is still needed to strive to make concordancers more user-friendly and more suitable for language learners (Horst, Cobb, & Nicolae, 2005; Krishnamurthy & Kosem, 2007).

2.1.6 The internet and other technology to collect and share examples

In addition to concordancing, the application of other computer solutions may also be able to provide a source for more examples. Activities have been designed for learners which attempt to make the most of the internet and collaborative technologies while putting the focus on the learner to generate their own reference bank. Online collaboration for

learners in building specialised dictionaries or word-banks has been shown to be useful. Horst and colleagues (2005) started with carefully selected sections of online publications and led a group of learners through the process of building word-banks to store vocabulary items and then test each other as part of increasing their academic vocabulary. Friedman (2009) found that guiding learners through the process of finding useful examples on web-pages and then setting a task where they created their own learner dictionary through *Google* searches⁴ was effective, particularly when collocations were used as search terms. However, web text presents many challenges for learners due to the fact that texts are not usually designed with second language users in mind and because there may be difficulties finding texts at the right level (Loucky, 2005). One way of harnessing the vastness of web data but “refining” it for language learners has been presented by Shaoqun, Franken and Witten (2009). Through making use of *Google* n-grams and combining results from established corpora, they showed that it is possible to build a tool for learners and reported positive observations of how learners made progress in forming more natural writing as a result.

2.1.7 Items in L2 with limited L1 congruence

Given that the literature has demonstrated most learners have a preference for translations, and that, as has been argued above, learners of English in China often lack the awareness of differences between L1 items and L2 “equivalences”, it would seem important to consider which language resources students use when direct translations will not provide a simple answer. In a recent study of how Portuguese learners of English respond to this linguistic problem, Frankenberg-Garcia (2011) demonstrated it was possible to construct a measurement to explore the strategies learners adopt when their knowledge of L1 is unlikely to be helpful. The results of her test and questionnaire showed a strong preference for grammar books and bilingual dictionaries, even though it was found that these would frequently lack the kind of information necessary to retrieve the correct answers. One interesting result was that a commonly reported source of information under the open category “other” was the internet. As part of her discussion and conclusions she noted that in the test learners had not been asked to indicate a specific website and that this might be a useful avenue for future exploration. By making minor changes to make it fit in the Chinese context and by extending the instructions in order to elicit more details about preferred websites, her test was felt to be a very useful way to

4 www.google.com

investigate the perceptions and habits of Chinese learners in high-school, a typical Chinese university and one of the new internationally formed universities based in China.

2.1.8 Purpose of the current phase of this research

The aims of the study reported in this chapter were to answer the following research questions:

1. What kinds of reference tool do Chinese learners prefer for different kinds of language problems where use of L1 is unlikely to help?
2. Are there different patterns between high school and national curriculum student groups compared to students studying EAP in an institution where English is the medium of instruction?
3. How widespread are notions about the use of authentic examples for both teachers and learners in these situations and do they match their preferences and understanding of language reference tools which are available.

2.2. Method

2.2.1 Participants

2.2.1.1 The institutions

For both the teacher and student components of the study, participants came from three institutions in the same city in Eastern China. The economic growth of cities in the provinces on the central Eastern coast of China is well known and middle tier cities such as the one in this study have increasing affluence as well as increasing levels of internationalisation. The high school was a high level state school which catered for around 1,000 students from age 12 to 19⁵. It will be referred to as institution "A". The university following the national curriculum for English in China had over 24,000 students

5 Ages in China are sometimes given as "虚岁" (age 1 at birth) and sometimes as "实岁" or "周岁" (age 0 at birth); therefore, it is not always possible to gather information about age in a consistent way. On the questionnaire, students were asked to indicate "实岁" (the same as age is counted in Western countries), but even so, respondents may not have taken sufficient notice of this. Therefore, ages given in this paper should be considered in the range plus or minus 1.

at undergraduate level. It will be referred to as institution “B”. Finally, the third institution was a Sino-British university established in recent years as part of the opening up of higher education to international universities in China. It is currently undergoing dramatic growth in numbers, but at the time of the study, it had around 4,000 undergraduate students on campus. It will be referred to as institution “C”.

2.2.1.2 Teacher survey

English language teachers from the three institutions were invited to take part in the study and to complete paper-based questionnaires. A total of 49 teachers took part, and approximately one third were male and two thirds were female. From institutions A and B, all the teachers were Chinese nationals and non-native speakers of English, and this can be considered very typical for state institutions in China. At institution C, around 57% reported themselves to be native speakers of English, with just under 23% from mainland China. The level of teaching experience varied in all institutions, with some teachers reporting less than 3 years' teaching experience and some more than 30.

2.2.1.3 Student survey and test

Students were recruited from the same three institutions to take part in the test and questionnaire. The vast majority of students at all three institutions were Chinese, and in order to focus specifically on the needs of Chinese native speakers, data from two international students who volunteered to participate at institution C were excluded from the results and analyses below. The reason for this was that the questionnaire focussed specifically on the use of Chinese-English reference resources and the language of the questionnaire was Chinese.

The high school participants (institution A) were 162 final year students (reported ages ranged from 17 to 20). The 47 students from institution B were all in the same first year College English class group, with majors mainly focussed on medicine and forensics. The students from institution C were recruited from finance related majors (a group which recruits the highest number of students at the university), from both Year 1 (87 students) and Year 2 (34 students) of their four year degree. Most of the students in the high school came from the city or surrounding area, but just over 19% came from outside the city, and just under 2.5% from other provinces. For the two universities, 12.8% and 16.5% of students at institutions B and C came from the city where the study took place respectively. More female students volunteered to take part than male students in all three institutions,

with just over 53% of students from institution A, 59% from institution B and 83% from institution C. It should be noted that Finance programmes at institution C tend to attract a higher proportion of female students.

2.2.2 Materials

The materials consisted of a test and two questionnaires.

The student groups were given a modified version of the test which was originally developed for Portuguese students by Frankenberg-Garcia to focus on language problems which cannot be solved through L1 direct translation (2011). This test elicits responses from students as to the kind of reference tool they would turn to if they were unable or unsure of the correct answer. The same five categories of language problems were used: selection of appropriate prepositions, formation of irregular past-tense verbs, spelling, collocation and hidden meaning. The 20 items in the test were kept the same as in the original study, but the instructions were translated into Chinese and there were a few significant differences made to the options which were made available. First, the options for Portuguese-English and English-Portuguese dictionaries were switched to Chinese-English and English-Chinese. Secondly, since in her study Frankenberg-Garcia reported that the encyclopaedia was not a popular choice, but that the internet was the most popular resource students entered under “other”, the choice of the internet was added, and students were invited to specify the website.

The questionnaires for students and teachers were fairly similar in content, although the teacher version was presented bilingually while the student version was Chinese language only. For open-ended questions, participants were invited to respond in either English or Chinese. Items included questions about learning and other background information, use of reference tools and ideas about language learning. There was also a hand-out which showed examples of eight different reference tool types and participants were asked to indicate which of these they thought would be most useful for different kinds of language problem. The hand-out was also designed to provide some useful guidance to different kinds of resources (including web links) as a way of helping students learn more about the resources which had been listed in the test after the end of the session. This hand-out was provided to participants as it was not practical to run the kind of follow-up sessions which Frankenberg-Garcia (2011) conducted.

2.2.3 Procedure⁶

Following a pilot, the student tests and questionnaires were conducted within a two week period. Students within institutions A and B all took the test and completed the questionnaires at the same time. However, at institution C two sessions were scheduled to fit in around different timetables for the first and second year groups, but both these took place on the same afternoon. The students took the tests for the most part in silence with minimal prompting from invigilators, and after completing the test activity in their own time, they raised a hand and their paper was collected and they were given the questionnaire and hand-out.

The distribution of questionnaires for teachers for the three institutions varied slightly, with one volunteer acting as a distributor of the questionnaires at each of the institutions A and B, while for institution C, it was possible to schedule a common time for all participants to go to a specific room and to complete the questionnaire at the same time.

Translation of the wording for the tests and questionnaires and for the analysis of the open-ended questions was conducted by a team of 4 academic support officers, who were trained language teachers with a very high level of English. Data were entered anonymously, and results calculated for different groups and sub-groups using a spreadsheet.

2.3. Results and Analysis

The results showed some interesting differences between students studying at the Sino-British university where they had EAP courses, compared with students at the other institutions. There were also some interesting differences between the preferences reported from all three institutions compared with previous studies.

⁶ The author is grateful to the students and teachers at the institutions involved for volunteering their time to participate in the study which was presented in this chapter. Special thanks are also owed to the group of helpers from the teaching staff of the English Language Centre who assisted with translation work as well as distributing and conducting the questionnaires and tests.

2.3.1 Results of the teacher questionnaire

As can be seen in Figure 2.1, the reported frequency of use of different resources by teachers from all institutions was fairly evenly spread. However, while the English-English dictionaries were skewed towards more frequent use, concordancers were skewed heavily towards infrequent use.

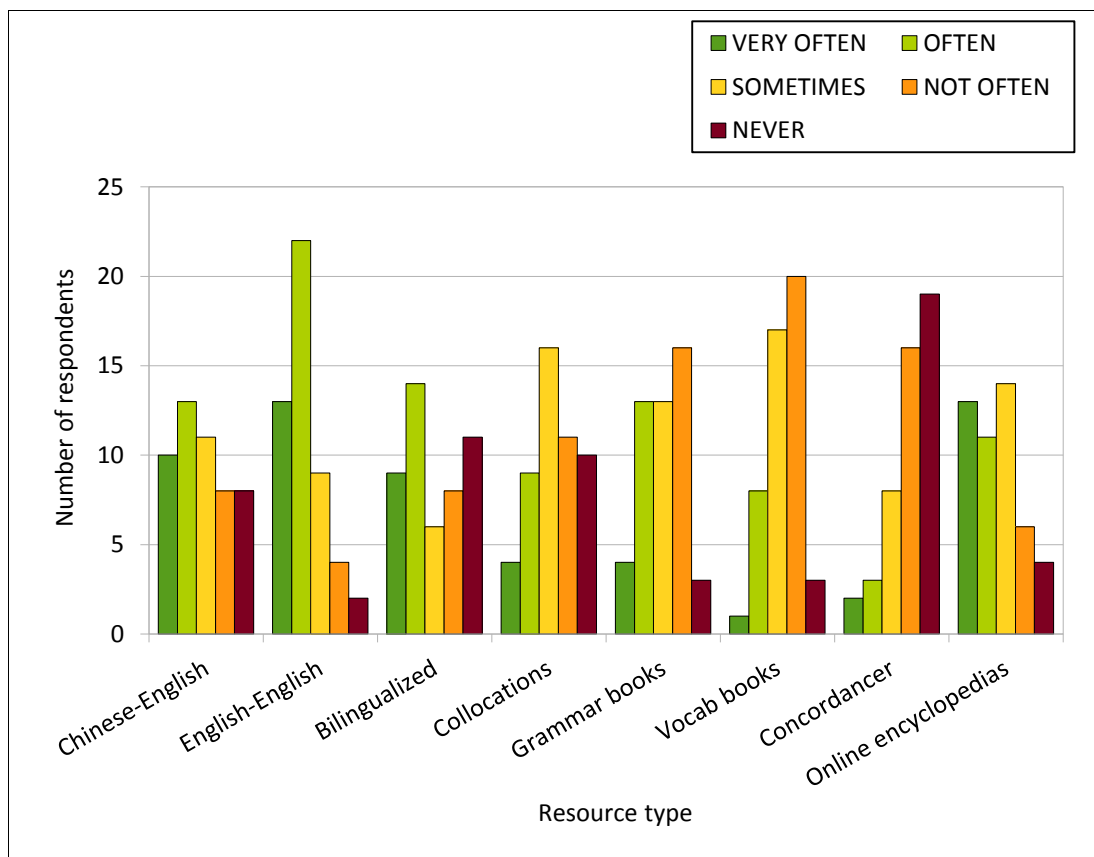


Figure 2.1: Teachers' reported frequency of use of different resources

Figure 2.2 shows the responses teachers gave when asked if they had ever used a concordancer. As can be seen, only around one third from institution C indicated “Yes”, and the vast majority of staff said “No”. This suggests that the actual usage of concordancers was even lower than the responses to the first question would suggest.

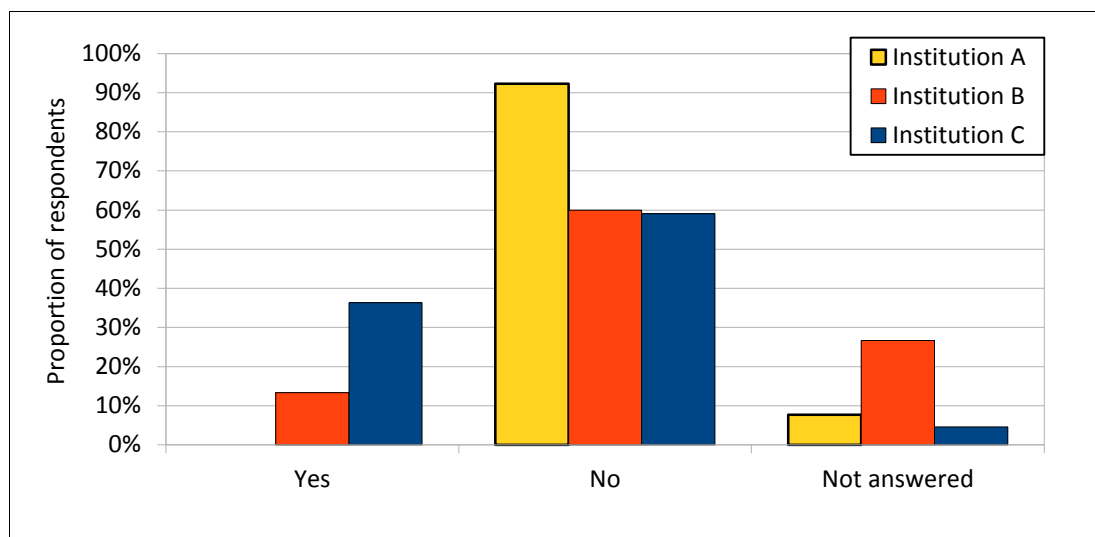


Figure 2.2: Reported use of any concordancer to produce examples

In the following two questions, those who had indicated “Yes” to having used concordancers were asked to indicate which software they had used in the past and how easy it was to use. Since none of the teachers at the High-school reported that they had used concordancers, the figures showing the results for these questions only show data for Institution B and Institution C. Those who had used software were asked to select software titles from a list containing *AntConc* (Anthony, 2004), *Lextutor* (Cobb, 2000), *The Sketch Engine* (Kilgarriff, Rychly, Smrz, & Tugwell, 2004), *WordSmith Tools* (Scott, 2010a) and were also given an option to specify software not listed. Figure 2.3 shows the reported use of concordancers with those provided by respondents as “other” marked with an asterisk.

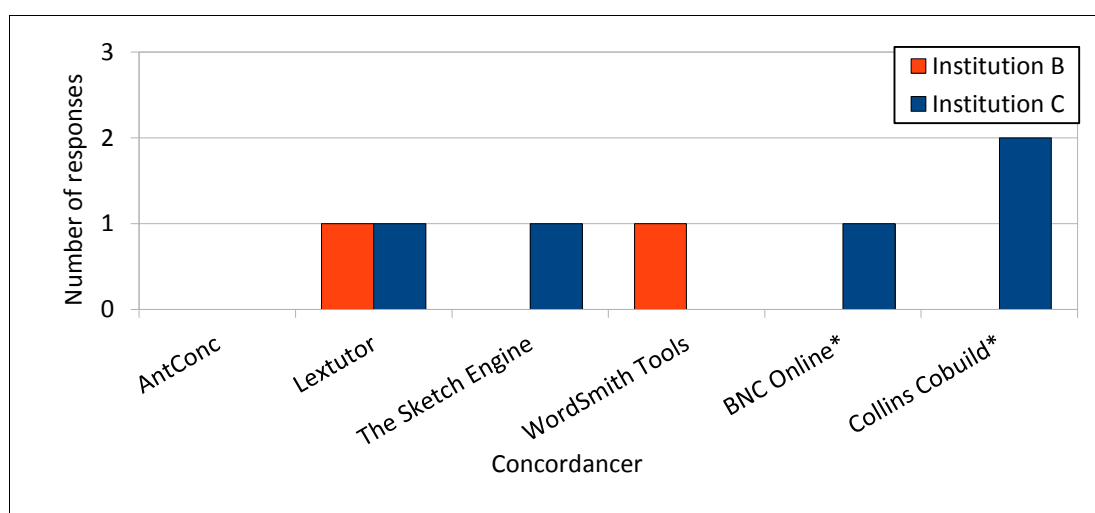


Figure 2.3: Reported use of different concordancers

Of the concordancers which had been used, bearing in mind the number of respondents in this category was very small, it can be seen that the teachers' experience was spread thinly across a few different titles, with web-based ones being more common than *AntConc* or

WordSmith Tools. Indeed, none of the teachers surveyed selected *AntConc*. Figure 2.4 shows that not all those who had tried concordancers felt that the software was easy to use.

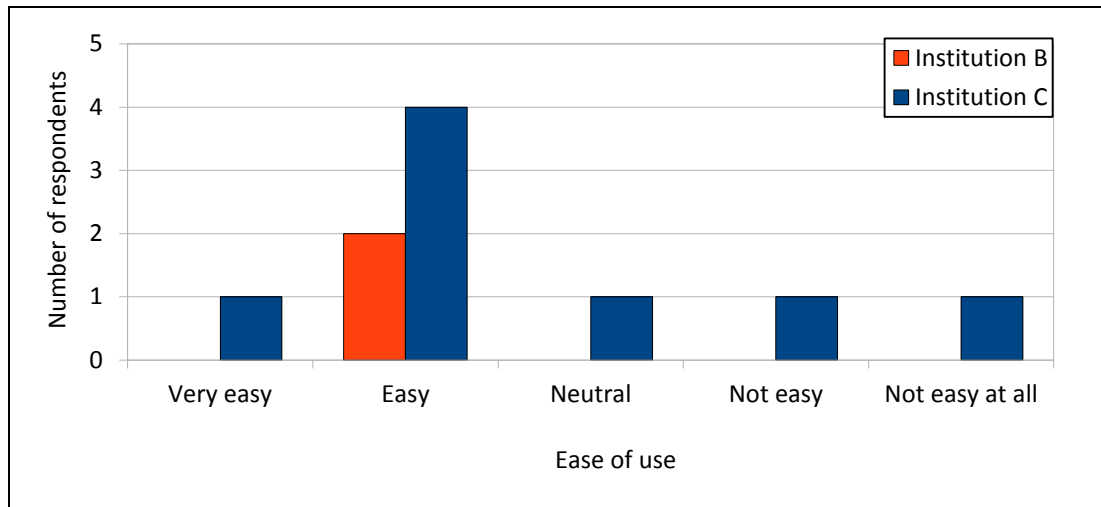


Figure 2.4: Teacher perceptions of ease of use of concordancing software

Figure 2.5 shows the responses from all teachers to a question about how easy it is to generate examples through intuition and reflection.

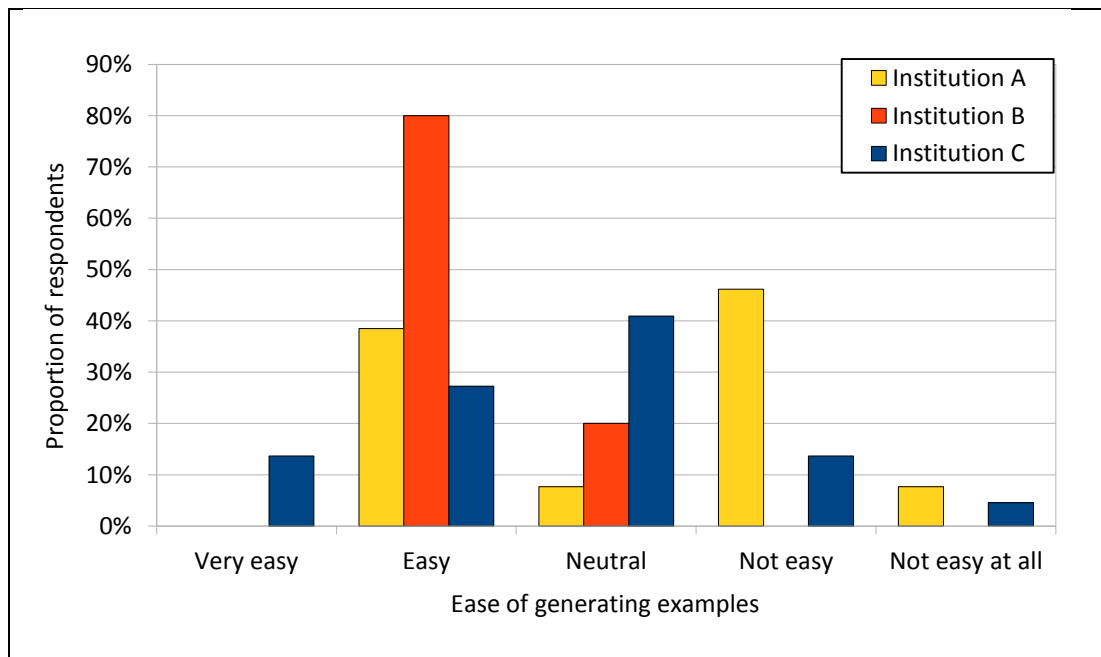


Figure 2.5: Teacher perceptions of how easily good examples can be generated through intuition and reflection.

Although at institution B, the 80% of respondents had confidence in their ability to generate examples easily, the High-school (institution A) teachers and the Sino-British university (institution C) teachers were fairly cautious with only 40.9% and 38.5%

respectively selecting “easy” or “very easy”. This suggests that concordancers or some other system for retrieving useful examples would be welcomed by many teachers.

Another question focussed on the importance of examples in language teaching, and the results are shown in Figure 2.6.

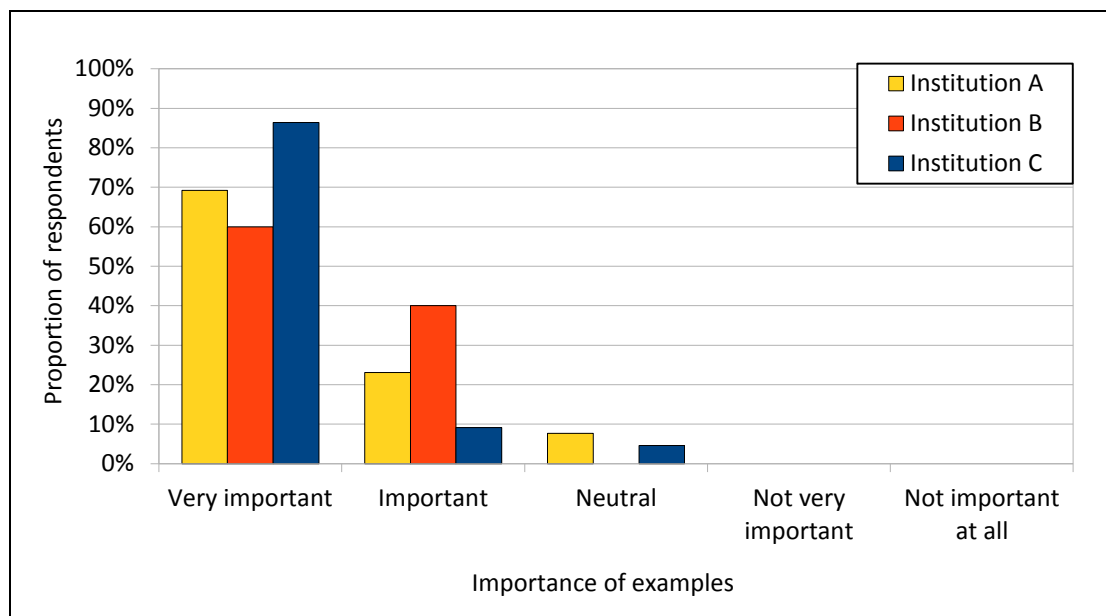


Figure 2.6: Teacher perceptions of the importance of examples

As can be seen, teachers at all three institutions showed an overwhelmingly positive awareness of the need for examples. Over 69%, 60% and 86% of tutors from the three institutions selected “very important”, and a combined total of 96% teachers selected “important” or “very important”. This is a strong indication that the centrality of grammar in English teaching in China may have been displaced by a focus on language as it is actually used.

Some differences between the teachers were revealed by a question about the main causes of mistakes in student writing. Figure 2.7 shows how teacher perceptions in the three institutions had slightly different patterns.

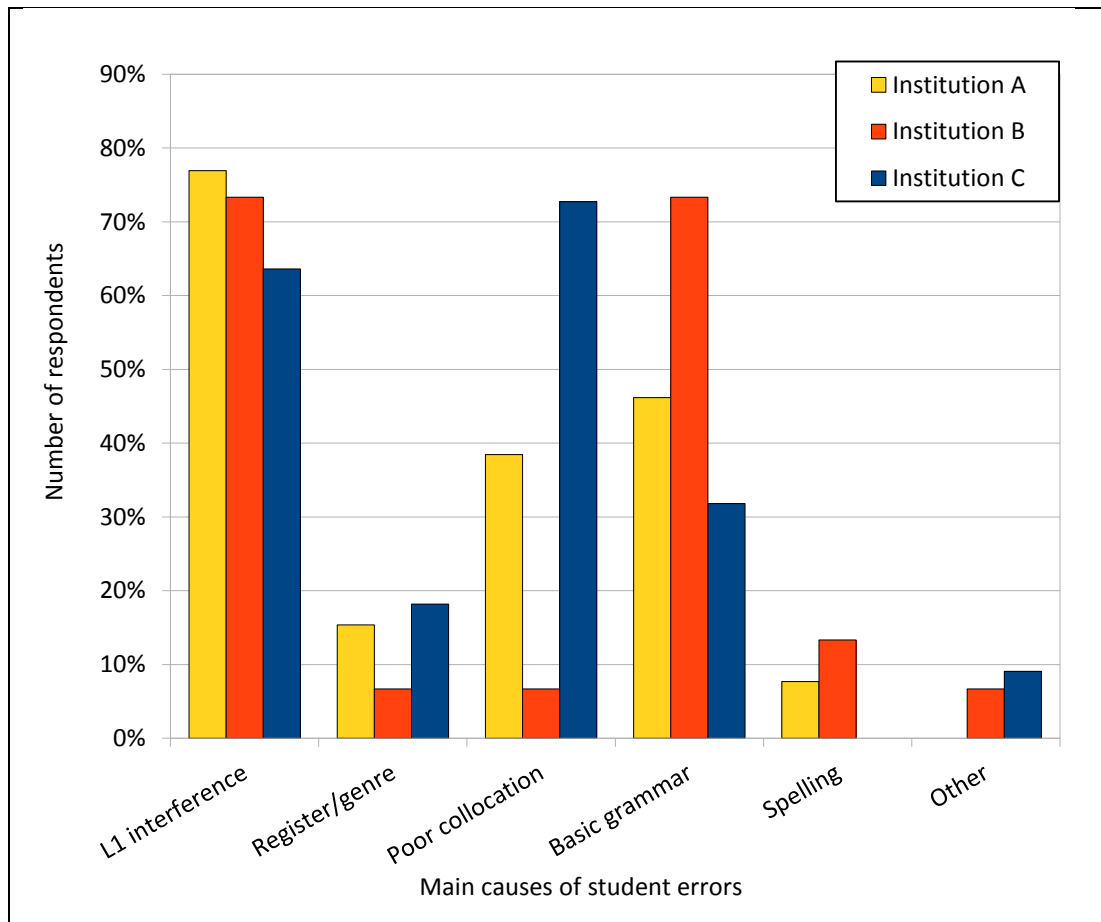


Figure 2.7: Teacher perceptions of the main causes of student errors

For the high-school, “L1 interference” was the most common choice with over 76%. “Basic grammar” was selected by a little under half of the participants (46.15%) with “poor collocation” following closely behind (38.46%). At institution B, “L1 interference” and “basic grammar” were chosen equally with 73.33% of respondents selecting these. Other influences were much lower with 13.33% for spelling and all others less than 7%. Institution C, perhaps due to the larger numbers of non-Chinese native speakers, had “poor collocation” as the commonest choice (72.73%), “L1 interference” proportionally a little lower than the other institutions at 63.64%, and “basic grammar” chosen by fewer than a third (31.82%).

2.3.2 Results of the student test and questionnaire

In the next section, the results of the student tests will be considered. In the test, as well as answering the questions, students were asked to indicate whether or not they were sure they knew the answer. If students did not know how to answer one of the questions, they simply needed to select “I don’t know” and they were not required to write an answer. Therefore, in the results each test item can be analysed according to whether or not each student believed he or she knew the answer and according to the correctness of the answer he or she provided. Obviously, the easiest questions would have been answered correctly by the most students and also be marked as showing the strongest levels of confidence. A correct answer with less confidence would suggest that the question was slightly more challenging. Areas of language which were difficult for the students would include those which they were aware were too difficult as well as those where they expressed some level of uncertainty but were unable to provide the correct answer. The areas of most concern might be those where the students believed they had written the correct answer but this confidence was misplaced. Figure 2.8 shows the differences in confidence and correctness across the different question types.

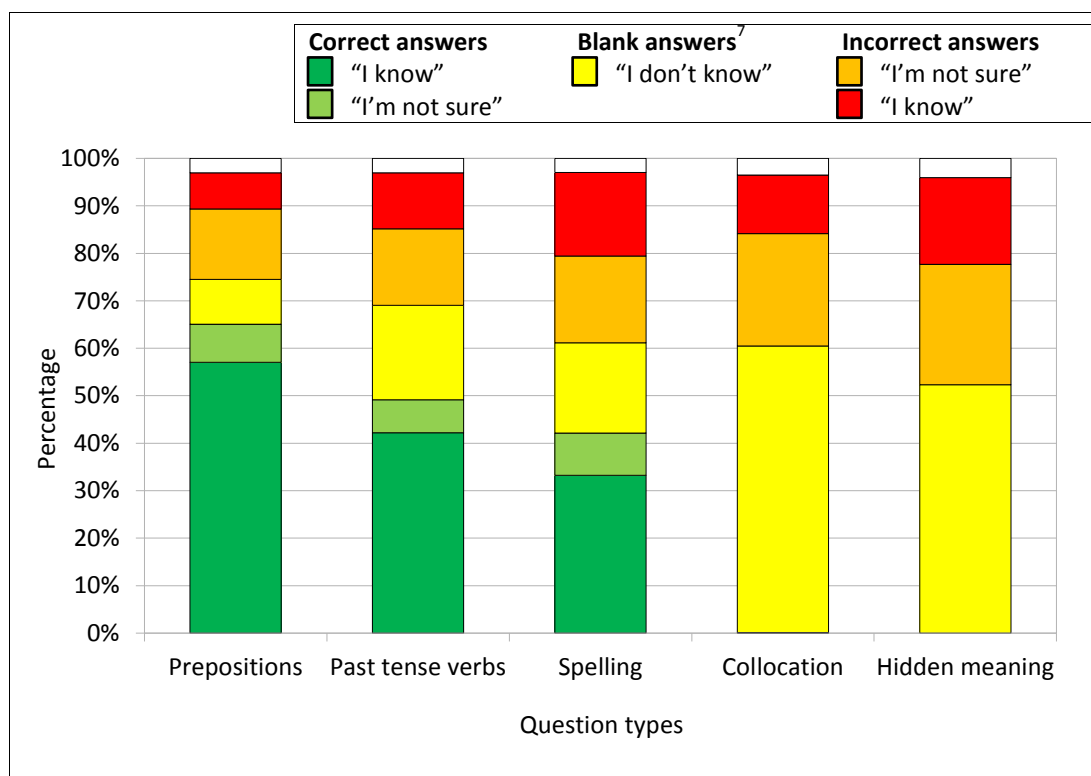


Figure 2.8: Reported knowledge and correctness grouped by question type

⁷ Participants were not required to write an answer if they selected “I don’t know”. However, some participants chose “I know” or “I’m not sure” but did not supply an answer. This explains why the stacks do not add up to exactly 100%.

As can be seen in Figure 2.8, the students found the 5 types of questions progressively more difficult and for “collocation” and “hidden meaning” (types 4 and 5) almost no students were able to answer correctly. However, for these two types the majority of students were aware of the gap in their knowledge. The overall tendencies of these results are similar to those reported in Frankenberg-Garia’s paper, where the Portuguese students found the “hidden meaning” question types the most challenging, closely followed by “collocation”. However, in her study more than half of the responses across each of the five areas were “not sure” or “don’t know”, while the results for students in China presented here show that it was only in the question types for “collocation” and “hidden meaning” where the majority of responses were “not sure” or “don’t know”. Nevertheless, like in her study the test provided an effective means for raising the participants’ awareness of some of the limitations of their linguistic knowledge.

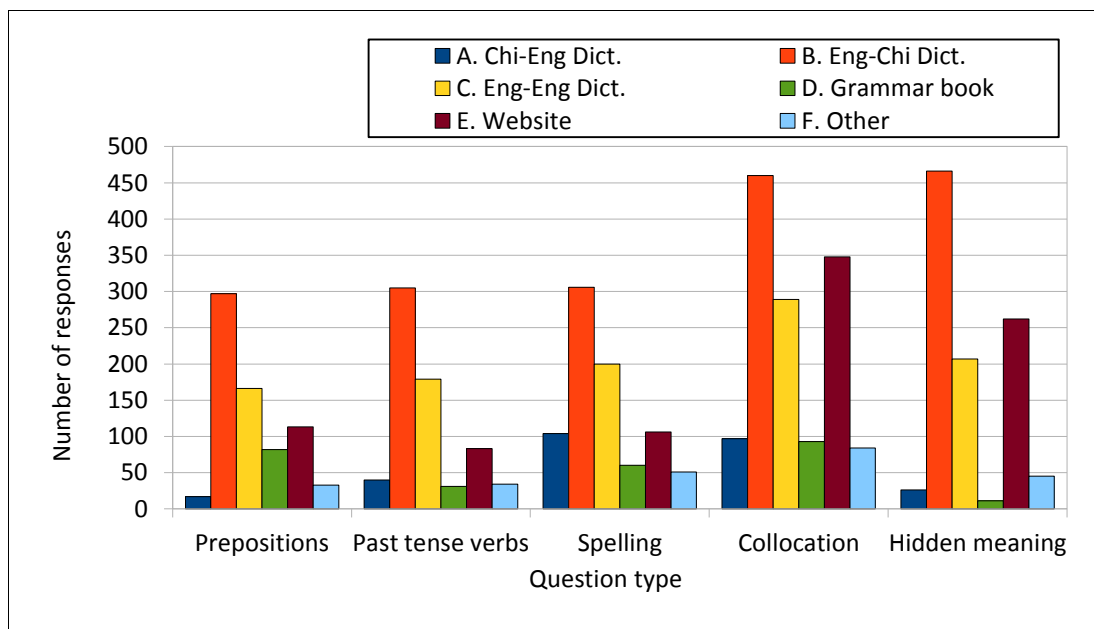


Figure 2.9: First reference choice grouped by question type

Looking at the first choice of reference given for all students (Figure 2.9), it is clear that the type of reference varied, with monolingual dictionaries and the internet being more popular for types 4 and 5. Limiting the data to institution C (as shown in Figure 2.10), however, shows that this preference for English-English dictionaries and the internet is much more pronounced. These results contrast sharply with those reported in Frankenberg-Garcia’s study, where for the first two question types the overwhelming majority of responses were to consult a grammar book. For the other question types, the results presented here show a preference for dictionaries over a grammar book which is

much stronger. It is also clear that for this new study in China, the new option for “website” was particularly popular as a choice for the last two question types.

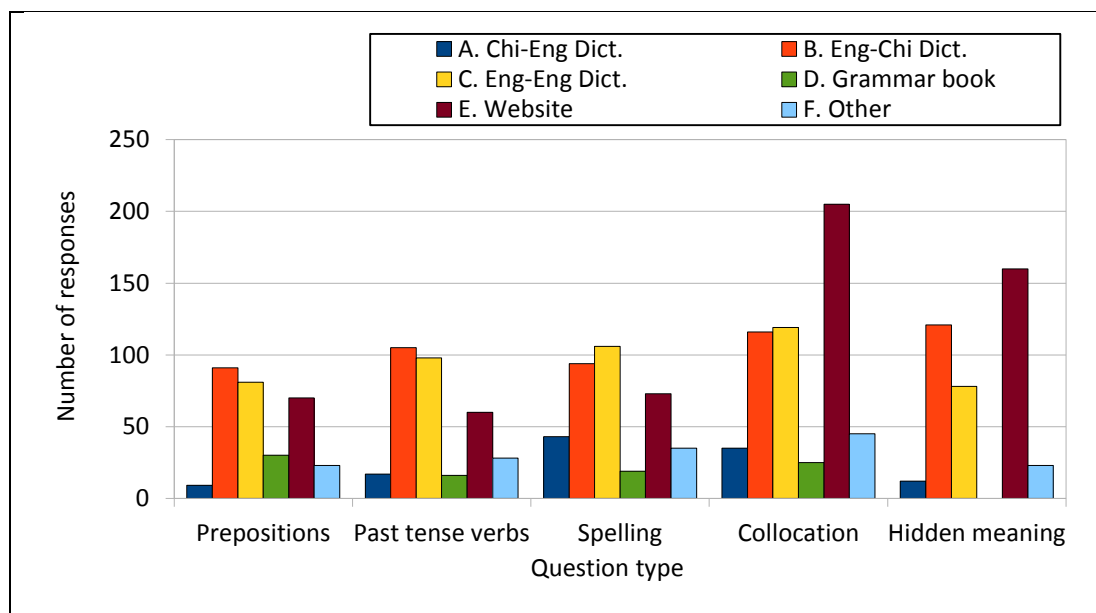


Figure 2.10: First reference choice grouped by question type (Institution C only)

One of the main differences between the test as it was developed by Frankenberg-Garica (2011) and the version used for this study was the added requirement in the new version of the test for students to specify which website they would use, and this proved to be very helpful. Figure 2.11 includes data from all three institutions, and shows the overwhelming popularity of search engines: especially *Baidu* (a Chinese search engine).

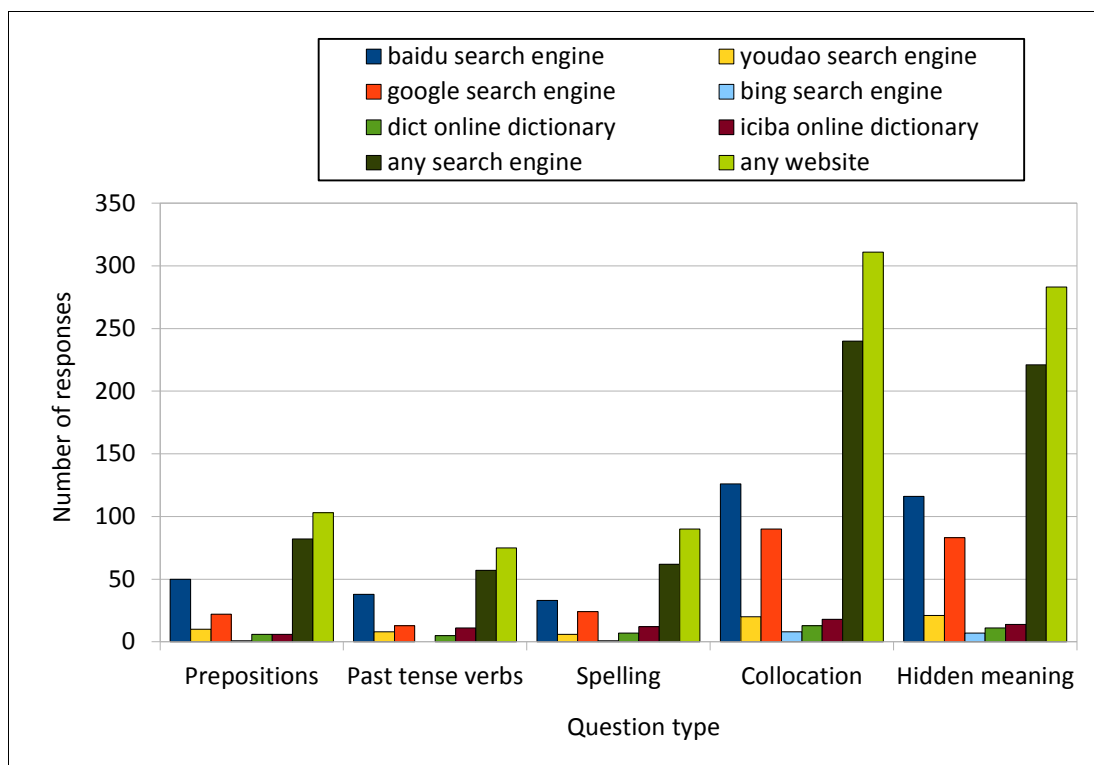


Figure 2.11: Websites where this was the first choice grouped by question type⁸

It really was quite surprising to find so many students listed search engines as their first or second choice of reference tool. It should be noted that for some short queries on a search engine, it seems that the top results may contain dictionary definitions or translations, but this does seem to vary considerably with the search engine and the item entered. Another important point is that many popular search engines are commercial enterprises and results are not sorted in an optimal order for linguistic use. Some of the techniques used in search engine design are explained by Croft, Metzler and Strohman (2010). In the literature, there has also been some discussion on the suitability of the web as a corpus for linguistic research (Kilgarriff, 2007; Kilgarriff & Grefenstetter, 2003; Robb, 2003). Nevertheless given that many students chose to painstakingly write out the name of the search engine for question after question, rather than just enter a number in a box to select one of the other reference sources, it seems that the preference for search engines over specific linguistic resources on the internet is very strong. Very few students responded “various”, although many did leave the choice blank. Of those who left the details blank for some questions in the latter part of the test, many wrote a search engine

⁸ youdao – www.youdao.com

as a response for several previous questions, perhaps indicating that they had tired of writing this multiple times.

In the questionnaire part, students reported how often they used the different resources.

Figure 2.12 and Figure 2.13 show the differences between A – the high school, and C – the Sino-British university.

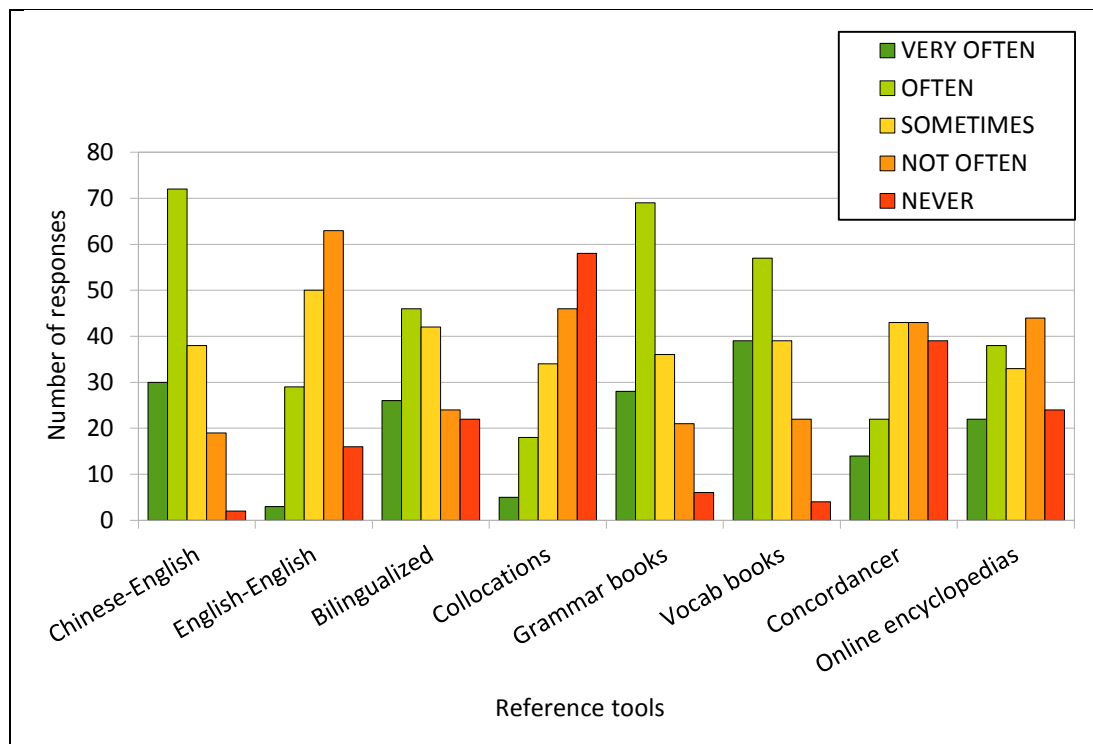


Figure 2.12: Reported frequency of use of different tools – students from Institution A

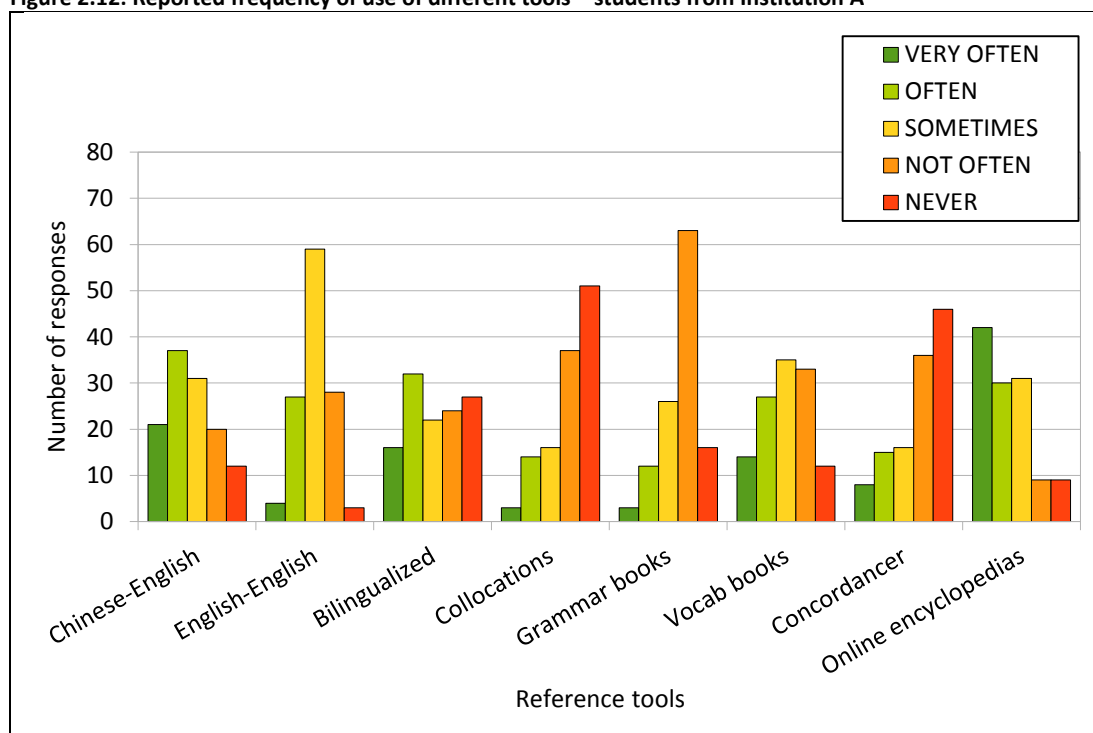


Figure 2.13: Reported frequency of use of different tools – students from Institution C

At the high-school (Figure 2.12), as expected, Chinese-English dictionaries, grammar books and vocabulary books show a trend towards more regular use, while English-English dictionaries and collocation dictionaries show the opposite. At the Sino-British university, as shown in Figure 2.13, however, the preference for grammar books is reversed, and there seems to be a greater reliance on online encyclopaedia resources. It is interesting to note how at the university, English-English dictionaries were used more frequently with the majority of students reporting use at least “sometimes”. For both groups it is also very clear that concordancers were not commonly used.

Another aspect of learner perceptions about resources is how easily they can identify the potential of each to provide different kinds of linguistic information. Using the hand-out with examples from different types of resources, the students were asked which they thought would be the most useful for six types of language problem.

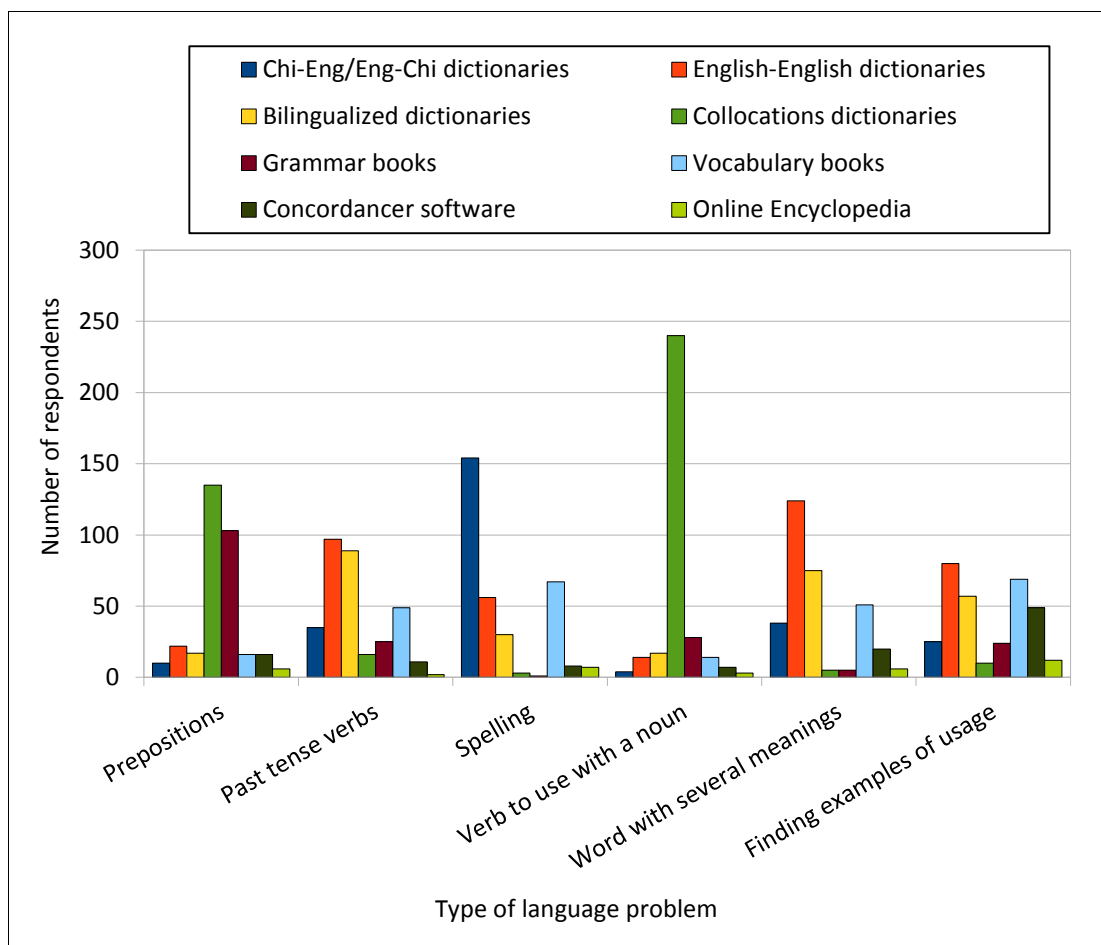


Figure 2.14: Best tools reported for different problems

Figure 2.14 shows the tools identified by students. Interestingly, collocation dictionaries were identified for prepositions and verb-noun combinations. The fact that the entry snippet from the collocations dictionary (*Oxford Collocations Dictionary for Students of English, 2002*) showed both prepositions and verbs collocating with “dread” may account for this, but it is interesting that so many students chose collocation dictionaries over other resources with which they were clearly much more familiar. Concordancers, though not frequently chosen, were strongest for words with several meanings and for examples.

As in the teacher version of the questionnaire, students were asked to indicate the importance of examples. Students at institution C showed the highest levels, but as with the teacher responses, examples were viewed as being “very important” or “important” by the vast majority of students at all three institutions. Students were also asked to respond to a set of four questions asking for strength of agreement to statements regarding the importance of the specific meaning of words, attention to rules of grammar, expressing something in a natural way and getting a reader to understand the message. It may be difficult to place a great deal of weight on reported views on what makes good writing, but “communicating a message” came across strongly as the most important aspect of good writing. Expressing something in a “natural way” was more controversial, but showed a slightly more positive response compared to “attention to grammar rules”.

The results show an interesting new slant on learner habits. Unlike Friedman's students (2009), there had been no explicit instructions or training to use search engines in this way, yet such large numbers of learners reported that they were making use of them. This demonstrates how attitudes to language learning have changed to the point where instant access to real life examples has filtered through to become a commonly accepted vehicle for learning. Given that the teachers at Institution C placed much greater importance on the influence of poor collocation as a cause for error in student writing, it could be that feedback from teachers has led to an increased awareness in the students as to the need to model use of language on real examples.

2.4. Conclusion

2.4.1 Limitations

The study presented here has a number of limitations. First, the study took place in one city in what is a vast country; it is not clear to what extent generalisations would apply to students learning in other parts of China or the rest of the world. However, the study does add to existing research and provides some interesting points of comparison and contrast between learner habits reported here and those investigated in other countries or at other times. Secondly, it is not clear how closely the strategies students and teachers reported in the test and questionnaires actually reflect their normal practice for different tasks as part of their academic reading or writing. In addition, although some of the student groups were quite large in size, the sample from institution B was much more limited. The findings here do, however, provide some important new insights into learner perceptions and suggest areas for future research.

2.4.2 Implications for EAP teaching and support

One of the clear findings of the study was the changing nature of the approach to language learning adopted by students as they move from high school to university foundation courses and on to English medium academic programmes. With the limited resources and strong pressure for high school teachers to ensure students are prepared for the University Entrance Examinations, it is unlikely that there can be a drastic change from the grammar-translation approach (reflected also in the preferences for translation dictionaries and grammar books), but with some additional training and practice with existing reference tools, learners could be better prepared for the real world of English communication or academic study where English is a vital tool for success. Many internationally published learner dictionaries and specialised reference books are available in China at reduced prices through agreements with Chinese publishing houses, and while it may not be practical for students to buy and regularly use more than one or two of these, if schools and universities could make more of them available and allow students to experiment and explore their strengths and weaknesses through classroom or self-study tasks, it is likely that some of the underlying problems teachers reported with language use could be overcome.

For teachers at all levels, but especially at institutions where the stakes are high such as the Sino-British University, the opportunities and limitations of various technological solutions need to be more fully explored. Both learners and teachers need to be more fully aware of

the differences in quality and function of hand-held devices as well as how and when search engines can be of assistance and when they cannot. As Frankenberg-Garcia (2011) argues, an important aspect of the use of reference tools is the kind of language problem that is used as a starting point. Experience and guidance with these needs to be in the context of real communication issues and real language difficulties, so learners can see how tools can be used as a solution to various problems.

2.4.3 Implications for design and scope of future tools

The results regarding search engines provide empirical support for some general observations about the use of search engines for linguistic information. For example, Kilgarriff and Grefenstetter (2003) have commented on the use of search engines for spelling by browsers in general, and Shaoqun and colleagues (2009) as well as Niño (2009) have made observations of language learners in particular. In terms of the development of future language learning tools, the overwhelming popularity of raw web search results and the demand for greater access to real examples cannot be ignored. Developers need to learn from the search engine model and consider how its speed and comprehensiveness might be mirrored or harnessed. Of course there is still an important place for all the reference books and tools mentioned, with learners showing some interest and experience with most of them, but with the changing nature of language and the uncharted waters of English for many specific academic disciplines, internet based or large datasets need to be a key area for exploration. If a linguistic web index is created for researchers along the lines argued for by Kilgarriff (2007), another potential group of users may be second language learners, especially if it can provide the speed and look and feel of a more familiar search engine. Projects to build learner tools based on web data such as those reported by Shaoqun et al. (2009) also hold great promise. The specialist EAP instruction for English medium institutions in the greater EFL setting is not going to be a big enough market to warrant professionally developed tools. Teachers and developers need to have less fear of exposing learners to examples of real authentic language and need to consider how software tools could be developed to support this. In some ways we have been held back from providing learners with direct access to larger language resources by fears of overwhelming them. The fact that search engines are used without any guidance shows that information systems are becoming a normal part of the study life of students.

2.4.4 Lexical Priming

The prevailing view of how language operates has been that grammar and lexis are separate systems and sentences can be constructed merely by choosing any syntactic structure and slotting in vocabulary. This view is still prevalent in many areas of language teaching. It is evident in China in materials designed to familiarize students with grammatical constructions following sets of rules, and in the wordlists of vocabulary which are frequently used to introduce isolated meanings of individual words, usually with just one or two word by word translations provided. Over the last few decades, corpus linguistics has presented challenges to this view of language, and by drawing on evidence which can be found in the patterning of language choices in texts, it provides both a means of narrowing down the range of items to be taught through an emphasis on the most frequent usage, and also a raising of the bar in the sense of demanding attention be paid to relationships between items in terms of collocation and colligation. From Firth (1957) through to Sinclair (1991), and in a wide variety of corpus linguistic research as well as Systemic Functional Grammar, the necessity for language educators to move away from a belief in a grammar separated from the lexicon is plainly evident. The theory of Lexical Priming (Hoey, 2005) makes a valuable contribution to linguistic theory by building on a range of insights gained from corpus linguistics and establishing a framework and evidence for the existence of other relationships which account for a sense of the naturalness or creativity of produced language. Building on the corpus linguistics tradition, Lexical Priming provides a challenge to the idea that words can be freely slotted into phrases and sentences merely according to their grammatical categories, and Hoey introduces the theory as providing a cognitive explanation for why collocation is so pervasive. However, it is clear that the other claims which Lexical Priming makes also challenge prevailing notions of how words and collocations can be used. The concept of textual colligation challenges the idea that words can be used freely in different positions in the sentence, paragraph or text for example. The concepts of semantic association and pragmatic association challenge the idea that words can be freely slotted into sentence structures purely based on some sort of inherent or isolated meaning.

While theories of language arising from corpus linguistics are clearly aiming to enhance a description of language and to thereby drive developments for language teaching, they are not particularly concerned with describing a model for the acquisition of language or the processes which underpin language learning. Lexical Priming, however, fills this gap by using insights from corpus linguistics and corpus data as evidence to explain how

individuals are primed through exposure and use of language, and explaining how this priming process is the basis for first and second language acquisition. In this sense it provides a bridge between arguments from corpus linguists who might be caricatured as focusing too readily on quantitative results from their datasets rather than on a consideration of how and why the patterns ended up in their collection, and the arguments of those who prefer to focus energies on describing language learning strategies for learners of a foreign language. From a pedagogical perspective, the theory could also be used as a powerful metaphor for explaining to adult learners why their understanding of language may need to be adjusted and how they might go about exploring wider relationships between words, context and meaning. Given some of the entrenched views about language which are held by students regarding vocabulary learning strategies, for example, which are evident in both the questionnaire responses explicitly, as well as the assumptions which must exist behind their reported preferences for different resources, it would be unrealistic to expect a piece of software to be able to completely shake and remodel their view of language and language learning priorities. Nevertheless, if a simple image of the human brain encountering words and phrases through hearing, reading and production and thereby building up patterns and expectations for how these could and should be used is presented, it could provide an impetus for encouraging language learners to look more deeply at the contexts of the language they encounter and the language that they produce. The idea that traces in the human mind of language which has previously been encountered are similar to concordance lines is a potent analogy, and also promotes a balanced understanding of how corpus resources can be used to find evidence but cannot ever represent the true priming of any one individual. In this sense, university students, who seem to appreciate having the rationale for the teaching focus being introduced clearly, can be assured of the relevance of corpus data as a way of gaining insights into real language use, while they are also encouraged to be critical and mindful of any resource's limitations.

The motivation for the development of the Lexical Priming concordancer which is presented in this thesis was twofold. As well as being deeply rooted in an appreciation of some of the struggles and difficulties faced by teachers and language teacher managers in terms of helping students in China appreciate their language needs and develop their language skills accordingly, the project was also designed to enable teachers and students to explore various features of the theory of Lexical Priming without needing to teach the

theory explicitly. It would not be desirable to replace the wordlists and sets of grammar rules which students and teachers currently use with a complicated exposition of Lexical Priming with all the technical and linguistic background knowledge which that would require. The software is designed, however, to encourage exploration of some of its features and to make it possible to see tendencies of words and phrases which are not usually apparent in either dictionary examples or the output from other concordancing software. The project is also in line with suggestions from two reviewers of Hoey's book on Lexical Priming: Garretson (2007) suggested that technology could provide ways to make analysis of Lexical Priming less time-consuming; Kaszubski highlighted the scope for the theory in the design of "learner concordancing practices" (2007, p. 292).

2.5 Summary

This chapter has provided an overview of the language teaching context in which the motivation and priorities for the development of *The Prime Machine* were formed. As well as providing background to the particular needs of students of English for Academic Purposes, it has also reported on the results of a survey of students and teachers in a city in Eastern China. The results showed some of the preferences for language learning reference resources which they have and gave an indication of some of their underlying beliefs about language. The chapter closed with consideration of the theory of Lexical Priming as a means of helping language learners appreciate the importance of exploring the contextual environments of words and combinations of words.

Some of the findings had a direct influence on the design of the software, with the students' overwhelming preference for search engines being a key influence on the choice of software architecture and the design of the user interface which is described in the next chapter. While consideration is given in the following chapters to various aspects of theories of corpus linguistics, the primary focus as each of the software features is presented will be on how the methods and presentation of results can facilitate and enhance language learning.

Chapter 3: Software architecture and data pathways

The purpose of this chapter is to provide background to the decisions regarding the underlying technologies, database design and software development for the project. The title of the chapter encapsulates its dual purpose, aiming to set out the technical decisions and use of programming languages and the design of the database, but also introducing the kinds of data manipulation which take place as a corpus is added to the system and the pathways and functions these data have in the end-user application.

First, the overall software architecture will be introduced. After that, details about the kinds of corpora available will be given. Storage of the corpus data will then be outlined, and an explanation given regarding the use of summary tables. Having explained the flexibilities and limitations of the system in terms of overall access and corpus data, some of the features available to users of the search interface will be introduced. Through the descriptions it will be demonstrated how features of the theory of Lexical Priming (Hoey, 2005) have been incorporated into many key aspects of the software design. It will also be demonstrated how the design has been influenced by the aim of providing guidance to learners about tendencies of language to occur in specific contexts and environments related to this theory, in a way which is accessible to them. The chapters following will build on this foundation, introducing other capabilities of the software with regard to collocations and concordance line ranking in Chapter 4, some of the main distinguishing features of Lexical Priming for the software in Chapter 5, and the handling of key words, tags and metadata in Chapter 6.

3.1 Fundamental design considerations

Modern applications in general can be divided into several types of software architecture, each having its own strengths and weaknesses, and the adoption of a particular model will be influenced by the expected data throughput and the numbers of concurrent users of the system, as well as the amount of pooled data and pooled resources it is anticipated they will use. Daily life in this computer age brings us into regular contact with each of these software types, and while users may be aware of limitations of access or some of the problems such as speed or disruption to service, for the most part the software architecture is transparent to the end user. Some applications run on more than one platform, so for example, a member of staff at a university might use a client-server based

system for email on their work desktop (e.g. *Microsoft Outlook*), while accessing the same email service through an internet browser at home or through a mobile device (like *Microsoft Outlook Web App*). The client version of the application can draw directly on a larger set of locally stored mail messages, and also perform searching and ordering quickly. The sending and receiving of new messages is processed in the background while the user can focus their attention on other business. It also integrates more flexibly with other applications (e.g. *Microsoft Word* and *Excel* for mail-merge operations), and attaching files is fast and easy. The web version displays a limited number of email messages at a time, pulling each message from the server as it is required. Attachments have to be uploaded to the mail server before the message can be sent, rather than dispatched in the background as in the client version. It is obvious to the user that some of the features of the web mail service are more limited, and that connection problems will lead to temporary suspension of the service, but sending and receiving email works reasonably well most of the time through either of these architectures.

Deciding the appropriate software architecture for a new concordancer is closely tied to the choice and size of the corpora, but corpus tools with different underlying technologies are already available. *WordSmith Tools* (Scott, 2010a) is probably the most well-known standalone suite of corpus tools. The user of *WordSmith Tools* installs the software on a single machine and can work with corpus texts without connecting to a network. The network version of *WordSmith Tools* runs through shared folder technology, with connectivity in order to check sufficient concurrent user licences remain available, and in order to be able to save data to a shared space, but the saving of data and the calculations made are performed by the user's own computer. *AntConc* (Anthony, 2004) works in a similar way to *WordSmith Tools*; the main difference in terms of underlying architecture being that its freeware status means network versions are not needed, and it is cross-platform so will work on a standalone computer running *Windows*, *OS X*, or *Linux*. Before using either of these standalone solutions, the user needs to download and install the application file on their computer, and then point the application to the text files containing the corpus they want to investigate. *WordSmith Tools* includes facilities to collect web pages to create a new corpus, as well as powerful file manipulation tools to transform various file formats into usable data, but for an apprentice user, the starting point for *AntConc* is the same as for *WordSmith Tools*: place a copy of the program and the corpus files on the target computer and then use this computer to search, calculate and display the results.

Two other highly specialized corpus tools which operate as standalone applications are *CenDiPede* and *ConcGram*. The former is designed as a research tool and it runs as a standalone application through a *Java* interface. For *CenDiPede* the corpus files to be used for analysis must be available on the machine in both parsed and unparsed form (Garretson, 2010). *ConcGram* requires smaller text files to be merged into one file in plain text format before it can be used in the main application (Greaves, 2009).

Some corpus work can also be carried out on the user's own computer through scripts rather than dedicated software tools. Meyer (2002) suggests that shared scripts could replace dedicated software for corpus research. However, a decade has passed since this suggestion was published, and although some research and book chapters describe script approaches, it seems that Scott (2008) is correct in arguing that use of scripts is probably beyond the programming skills of most corpus researchers. Despite this, Meyer's (2002) points regarding the flexibility of scripts and his encouragement to researchers to experiment with several different concordancing software tools to match the focus of a particular project are still relevant.

As well as standalone architecture, corpus tools are also available as websites. Although commercial web-based concordancers like *The Sketch Engine* (Kilgarriff, et al., 2004) allow the user to create and store their own corpora, the expectation with a web-concordancer is that either the institution providing the corpus will have its own online service, or that a range of well-known corpora will be available without any need for the user to set anything up. For example, there are web interfaces for general corpora like the *British National Corpus* and the *Corpus of Contemporary American English* (made available by Brigham Young University, <http://corpus.byu.edu/bnc/>), as well as highly specialist corpora such as the *Vienna-Oxford International Corpus of English* (Breiteneder, Klimpfinger, Majewski, & Pitzl, 2009) and the *Michigan Corpus of Academic Spoken English* (Simpson, Briggs, Ovens, & Swales, 2002). The processing load for a web-concordancer is hidden from the user, but the technology relies on a server, or a cluster of servers, receiving the user's query, retrieving the results, and then creating a customized web page to display them. The user's computer does not need to process the corpus data at all, merely running a browser to interpret the web page produced by the distant server. Rayson (2002) argues that a web-based system saves the user time in learning a new application and also saves time as no extra software has to be locally installed. As seen in the results from the survey of teachers in Chapter 2, it was these web-based concordancers which have been most widely used in

contexts such as Suzhou, China, but it should be remembered that the proportion of teachers who had used a concordancer at all was very small. The prospect of making a corpus tool available to any user on any browser is an attractive one, but as web-concordancers have become more powerful, it is clear that their user interfaces now differ as much as those of standalone applications. Two important limitations of web-based concordancing are also obvious to the end user. Firstly, the number of results made available is usually much more limited than on a standalone system. Secondly, the speed of access in a computer lab with multiple users is noticeably slower than when accessing the system from a dedicated internet connection. It is also worth noting that in terms of using software with learners, several studies have reported disruption caused by internet access problems during the data collection phase (e.g. Kaur & Hegelheimer, 2005; Vannestål & Lindquist, 2007; H. Yoon, 2008). Internet services can be switched off or blocked, or general internet access can become so slow that it becomes no longer feasible to use a service. Considerations regarding dependency on internet connection to links outside the country are particularly important in contexts such as China, where government restrictions, earthquakes or fishing boats can cause significant disruption⁹.

The most famous and flexible of the web-based concordancers is *The Sketch Engine* (Kilgarriff, et al., 2004). In terms of access for language learners, however, as well as connection issues, the costs of full subscriptions to this service are well above what institutions are likely to pay. Its flexibility, providing an integrated system for collection of web data, processing for POS and tree tagging and the production of Word Sketches, makes it a valuable resource for linguistic research and for lexicographers. It includes access to a wide range of corpora, with a growing number of languages, and if students at an institution holding a subscription need only access to these, they can explore collocation patterns using several different statistical methods and view the Word Sketches. For specialist, custom built corpora, however, as well as the initial subscription, there are other significant costs to consider if a user wants to work with a large corpus, as once the token count exceeds one million words, the annual fee per million words quickly adds up. The university-wide licences give access for students in their thousands to the ready-built

⁹ There were several fishing boat-related disruptions in 2001 and an earthquake off the coast of Taiwan in 2006 (see <http://www.china.org.cn/english/China/194131.htm>), as well as another major fault in 2009 (see <http://www.computerworld.com/article/2526855>). Incidents like these make access to overseas websites extremely slow, often for several weeks at a time. The “Great firewall of China”, internet censorship in China, is also well known at this time, with services to Google, Twitter and Facebook often disrupted (see <http://news.bbc.co.uk/2/hi/4587622.stm>).

corpora, but only make the corpus building tools available to fewer than 10 “full account” users. A few small corpora are available as open resources, without any subscription or login required, but it is clear that the Word Sketches and other data available for these are much more limited. With costs of over 500 pounds per year for each custom built 100 million word corpus ((9xGBP10)+(90*GBP5); see Sketch_Engine, 2013), projects based on *The Sketch Engine* wanting to match the size of the ready-made corpora are very costly. However, the cost to the service provider is also high. With the processing load for data transformation, storage, retrieval and calculations all falling on the servers, it is clear that the hardware demands for separately storing multiple custom made corpora are huge.

A third approach to software architecture is to use a small piece of software running on the user’s computer which connects directly or indirectly to a database server storing the data. This approach benefits from being able to share the processing load of calculating and displaying the results across both the user’s computer and the server, but also means that the data only need to be distributed as they are needed. Similar designs for university-based research were implemented before web browsers became commonplace. However, in recent years, this design concept has taken on a new form as the “App”. These small applications are quick and easy to install on computers or other devices and provide easier access and a more customized look and feel for frequently used internet services. The overall functionality of such applications is split between different tiers, with a server or layers of servers providing access to data and performing *Structured Query Language (SQL)* calculations. This approach can also provide more information to the provider of the service for security or software improvement schemes, as the local application is able to access and record more information than would be possible with a website cookie¹⁰, and periodically pass this information on to the server for remote storage and consolidation. The feedback benefits of this software model for the evaluation of this project are explored further in Chapter 7.

For a school or university setting, the design of a new concordancer for language learners might well see this tiered approach to software as a good solution. It could perhaps bring

¹⁰ Cookies are small pieces of information which a web site saves on a user’s computer. General information about the purpose of website cookies is available from: <http://www.microsoft.com/info/cookies.msp>. Regulations about the kind of information which can be saved and how it is used was recently changed in the European Union and details about this can be found on the UK Information Commissioner’s Office website: http://ico.org.uk/for_organisations/privacy_and_electronic_communications/the_guide/cookies.

the vision of Fligelstone (1993) closer, where the icon taking students in a computer lab to a dedicated corpus service is actually a client application which connects through to an institution's corpus server. Chambers and Wynne (2008) comment that with the current software options at typical institutions Fligelstone's vision is still far off. However, it is not uncommon for schools and universities to run servers for learning management systems (e.g. *Moodle*, *Blackboard*), email services, student records, and intranets. By setting up the server on a university network, access within the university would be faster and more stable than relying on external service providers. The majority of traffic would be handled within the local network, with smaller numbers of students accessing the system through the internet from off-site. In a setting where on-campus study modes are the norm, and where thousands of students use English as the medium of study, this multi-tier approach means relatively modest server hardware is able to provide sufficient levels of concurrent service. Small applications on computer lab desktops, or students' own hardware which is attached to the school's wireless internet connection (Wi-Fi), connecting to a dedicated institutional corpus server, would require little set-up for each individual, but allow the control of the resources to be managed centrally. A robust form of access management for corpus resources is important as copyright and commercial restrictions on corpora need to be honoured by the institution. This would also allow corpus resources to be available to all staff and students at an institution, rather than be an option explored only by a few teachers. Further considerations regarding the management of corpora in this kind of setting will be discussed in the section below.

3.2 Corpus selection

For researchers, the choice of corpus will be governed by the kind of analysis being undertaken and perhaps the availability of specialist corpora through their affiliated institutions and institutional subscriptions. It has been noted that when using corpora with learners there has often been a preference for smaller corpora (Chambers, 2007; C. Yoon, 2011). Reasons for avoiding larger corpora may be to help learners form a clear conception of what kinds of texts the corpus contains, and because of the difficulties software often has with handling large amounts of data (Chambers, Farr, & O'Riordan, 2011). Working initially with small corpora or even just a small number of text files may make it easier for learners to understand what a concordancer does, but the initial reward for learning how to use the software can also be much smaller, as the tools and statistics typically offer few results with limited discrimination between strong patterns of linguistic interest and

recurrent phrases which receive undue prominence as an accidental consequence of the sampling of the texts. In addition, the smaller the corpus, the greater is the risk that it will not contain all the words (types) and phrases which a learner wants to investigate. When concordancing is first being introduced to learners as a resource to assist them in language learning activities, it would seem especially important to try to avoid as much as possible the event that the concordancer returns zero hits for a specific search that they have made.

Another issue relating to the choice of corpora offered to learners is that students beginning to use concordancers are not usually in a very good position to decide which corpus is most suitable for their purposes. When Data Driven Learning (DDL) was in its early days, pioneers like Tim Johns were using microcomputers with postgraduate students and it was argued that the students would be responsible for selecting appropriate texts and feeding them into the computer (Johns, 1986). Some researchers have instructed learners how to build corpora themselves, and this would seem like a particularly fruitful way to demonstrate to international doctoral students the links between the academic texts they read and the development of their own lexical resources related to their academic discipline (Charles, 2012). At the lower end of the educational spectrum, Kilgarriff (2009a) suggests creating corpora from the web using a tool like *WebBootcat* (Baroni, Kilgarriff, Pomikálek, & Rychlý, 2006), so that younger learners have increased motivation by choosing exciting or more controversial topic areas as seeds for the corpus, rather than being given “safe” novels or other materials. However, for the growing number of undergraduate students using English as the medium of instruction (as described in Chapter 2) it seems more sensible to provide a few general corpora and a range of specialist academic corpora and take the difficult decision of selecting appropriate corpora out of their hands.

Copyright considerations are also important. While from a technical standpoint it would be quite feasible to distribute corpora on a USB stick along with the software itself, it is the copyright restrictions, or the obligations that corpus providers have to the copyright holders, that make such a system unlikely. By imposing limitations on the number of concordance lines which can be viewed, web-based concordancers not only restrict the amount of data their servers have to process, but also provide one way to meet copyright responsibilities. Issues of copyright and suggestions for how to manage resources to ensure only small samples are accessed are important (Anthony, Chujo, & Oghigian, 2011; Hemming & Lassi, 2003). Typically, permission for a concordancing service provider to hold

complete texts securely on a server while only giving users the facility to download small samples is much easier to negotiate with copyright holders. Whistle (1999) discusses the need to take copyright seriously, but also points to governments as a possible source for copyright-friendly resources. Unfortunately, for academic-oriented learners in specific subject areas learning through the medium of English there seem to be few readily accessible corpora available. Indeed, obtaining corpora suitable for the development of this project was not straightforward, and perhaps it could be argued that if gaining permission is difficult for a research degree project with specific outcomes over a specified time period, this issue could prove to be a serious disincentive for language teachers wanting to use corpora regularly with classes of students. Fortunately, the growing field of data-mining and the tendency towards open publishing mean that in future a wider range of datasets containing large numbers of quality texts from specific domains may be made freely available.

For the development and evaluation of *The Prime Machine*, a range of different corpora were selected. The *Guardian corpus* was used by Hoey (2005) for many of the examples introducing his theory of Lexical Priming, so it was also chosen in order to provide straightforward comparison, particularly during the early development of the project. Since the university where the author worked had a large number of students studying degree programmes related to finance and business, the *Financial Times corpus* (Harman & Hoffman, 1996) was also selected as a possible resource which could be suitable for them. The university also had a site licence for *the British National Corpus* (BNC, 2007), and this corpus was chosen both as a resource which could be used to demonstrate differences across different texts types and in order to demonstrate how highly structured corpus texts with part-of-speech tags and extended file headers could be integrated into the system. With a view to exploring ways of making a wider range of corpora available for specific academic disciplines, the collection of medical and biological academic articles, referred to in this thesis as the *SpringerOpen corpus* (SpringerOpen, 2011), and the *Hindawi corpus* (Hindawi, 2013) were included to demonstrate how open journal texts from data-mining could be imported and to show how successfully the system is able to highlight some of the features of academic texts. Finally, the system has also been tested with two corpora of learner writing: the *British Academic Written English corpus* (BAWE, 2007) and the written section of the *Spoken and Written English Corpus of Chinese Learners* (Wen, Liang, Yan, & Zhu, 2008), the *WECCL*.

The software architecture choices outlined in the previous section and the choice of corpora will not suit every teaching context, but the uses made of the data and the representations of these could be further developed to provide greater flexibility in the future. In the meantime, this project needs to be considered as working within a context where support for site licences for corpora such as the *British National Corpus* is available¹¹, and where production of specialized corpora for institution-wide access could be negotiated or sourced from open publishing organizations such as *Hindawi* or *SpringerOpen*. Alternative software architecture with a different work-flow from raw texts to the database could build on other aspects of the design and statistical methods, to meet the needs of users wanting to access self-compiled corpora. Similarly, with external support for server hardware, it would be possible to move to a web-browser based architecture, with much more powerful servers generating pages of results similar to the current client user interface. Some of these possibilities are explored further in Chapter 8.

3.3 Choice of programming languages

One aspect of handling large corpora for use with learners is the importance of speed. Learners need software which is both easy to use and fast (Mills, 1994). Scott (2008) has likened the viewing of concordance results to those of search engines, but the speed of a concordancer cannot usually compete with a search engine. Croft et al. (2010) explain that a response within 150 milliseconds is desirable to generate a perception of “instant” results for a user accessing a search engine. Even if other software is not able to deliver the results within this tiny time-frame, it is important to make the display respond in some way. Websites perhaps, particularly if accessed from overseas, bring a familiar waiting experience because the browser idly waits for the next web page to be retrieved or to eventually time-out. There is usually no way of knowing whether a slow query will eventually work, or whether a connection error message will appear. For applications such as flight bookings, one of the things which airline websites do is provide hints, advice and advertisements for related products as the user waits for confirmation of availability and bookings. A similar approach is taken on video games consoles, where for example *XBox*

¹¹ The announcement in 2014 that the BNC can now be downloaded at no cost from the Oxford Text Archive can be seen as a very positive move in terms of making resources more widely available, but it should be noted that some restrictions still apply and approval is needed (see <http://www.ota.ox.ac.uk/desc/2554>).

360 Kinect¹² displays health and safety reminders as well as tips for different aspects of games while the next part of the game is loaded. These features not only make it clear to the user that the application is still “alive”, but also try to make use of the waiting time to promote exploration of and experimentation with less intuitive aspects of the resource. In the field of information management, these features are known as “filler interfaces” and there has been some empirical evidence to show that incorporating filler interfaces, especially those containing both images and text, reduces a website user’s perception of waiting time and encourages a sense of continued engagement in the task (Y. Lee, Chen, & Ilie, 2012). The choice of programming language for the current project which is described below also included some consideration of how this waiting time could be handled effectively. Figure 3.1 shows some of the “please wait” messages from *The Prime Machine* which appear during waiting time.

¹² <http://www.xbox.com/en-US/kinect>

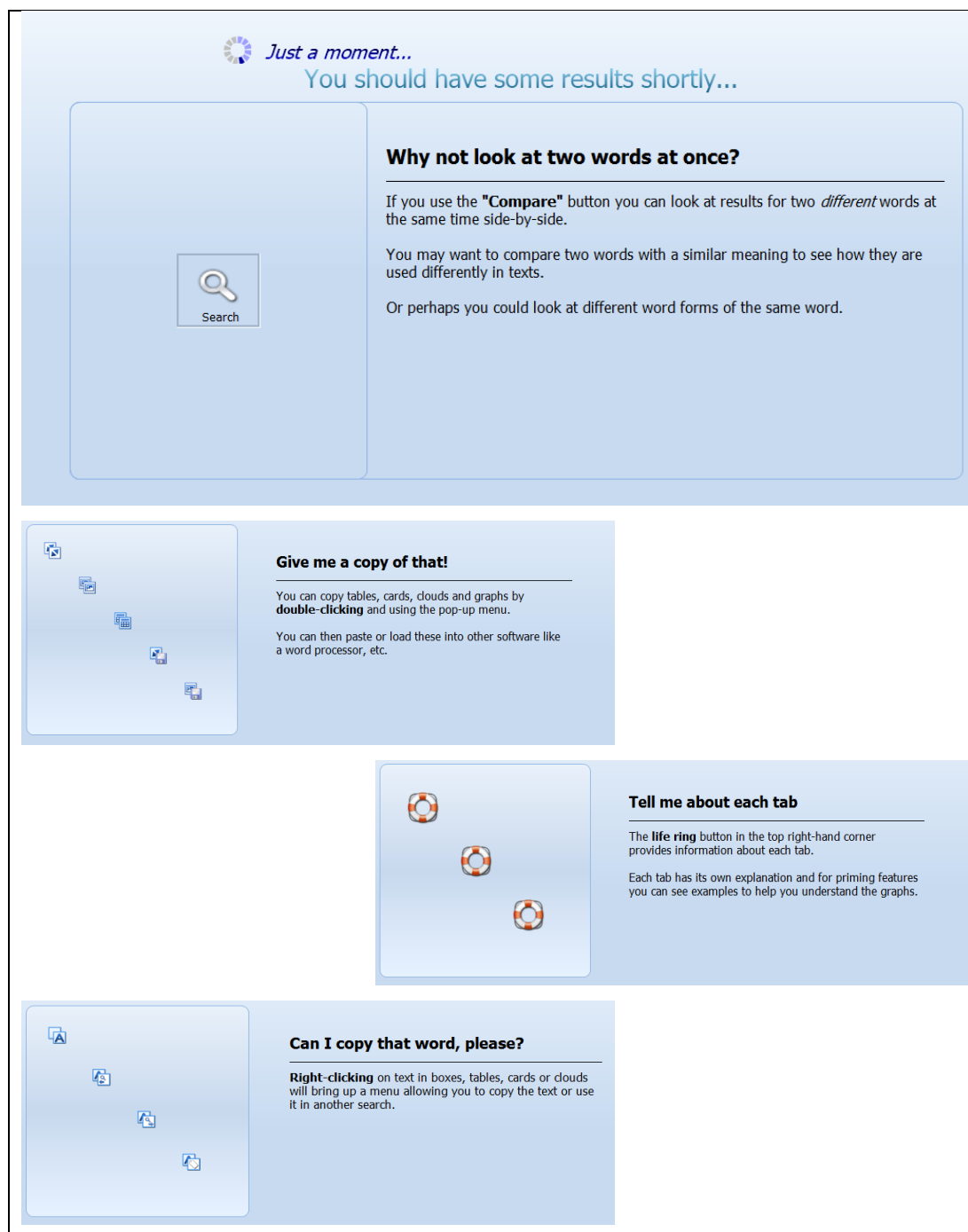


Figure 3.1: Screenshot of hints and tips which appear while results are retrieved (top) and some other examples of hints and tips (below).

Dedicated users of DDL in the classroom have shown that speed is not absolutely essential. In the early days of DDL, Johns (1986) explains that the computer would take about 50 seconds to produce a concordance. Simple concordance results are, usually, much faster than this, but extraction of other data from a corpus can take much longer. A list of concgrams for a long text produced by *ConcGram* takes about one hour, meaning it is best if these are produced by students independently before class (Greaves & Warren, 2007). In

the manual, it explains that producing a 2 word Concgram list for a corpus of a million words takes about one day (Greaves, 2009)¹³. The profiles produced by *CenDiPede* (Garretson, 2010) take around 2 minutes for each node, but produce so much data that further querying is needed in order to explore them. However, while researchers will be more tolerant of delays in producing corpus derived data, it could be argued that the technology most learners experience on a daily basis in other environments means that they will have very different expectations. Feedback from learners when using concordancers includes frustration with the time it takes (e.g. Chambers, 2007), and while some of this time could refer to what language teachers may see as productive engagement with the concordance output as learners search for patterns (i.e. time engaged in inductive learning), faster access to results ought to minimize dissatisfaction caused by having to wait for the software itself. The development of a system which delivers a wider range of statistics on features of lexical priming than is usually available, needs to take this question very seriously, as more statistics and more processing could lead to even longer delays.

Scott (2008) explains that *WordSmith Tools* has operated with on-the-fly processing since its inception, and that this approach has worked very well. *AntConc* also uses on-the-fly processing, and by slimming down the range of features with the needs of teachers and students in mind, it demonstrates that this approach could also work in the classroom. *The Sketch Engine* has also been used with learners. However, the aim of this project was not to replicate the functionality of *WordSmith Tools*, *AntConc* or *The Sketch Engine*, but rather to provide and draw on a wider range of Lexical Priming corpus information and to make both the retrieval and display of this information accessible and engaging for learners. *CenDiPede* can provide corpus-derived data for some patterns which are similar to this project, but Garretson's profiles are designed to be the basis, not the end results of research (Garretson, 2010). He proposes that one way to improve the speed of his *CenDiPede* system would be to store the results of pre-calculated profiles in a relational database. However, in the early stages of the current project, the possibility of using relational database queries to actually generate, rather than just store the summary data,

¹³ The hardware specifications and the operating system used in this example from the user manual are several years out of date, with *Windows XP* now beyond its lifecycle support end date (<http://support.microsoft.com/lifecycle/?c2=1173>), but it seems reasonable to assume that that even on modern hardware and a newer operating system the process would still take many hours.

was considered, and this has been the approach adopted. Since relational databases are not able to provide flexible text processing or a highly customized display of results, different programming languages were needed for different aspects of the system; a flexible programming language to transform raw corpus texts into rows of data in the relational database, *SQL* scripts to run on the database server to generate statistics, code for a middle tier server to take client requests and transform them into *SQL* queries, and a visually rich and fast programming language with which to develop the front-end client application.

Choices regarding computer language rely on some objective reasons such as the kind of data to be used, the devices and operating system platforms being targeted and support for suitable programming library functions. However, programming languages are also a matter of personal preference. Like Mike Scott, I took up computer programming initially as a hobby, and my first real software programs were written in *Pascal*. As Object Oriented Programming took off, I followed on with *Turbo Pascal 5.5* and then *Delphi 4.0*. Having been told as an undergraduate in the 1990's that *Pascal* was not the computer language to use for linguistic analysis, it was a relief and a joy to discover that the developer of the famous *WordSmith Tools* had also begun programming as a hobby, and that this famous software suite had been written in familiar *Delphi* code (see Scott, 2008). It is also worth noting that *The Sketch Engine* was developed using *C++* (not really that dissimilar to *Delphi*) and *Python*.

If cross-platform distribution was a primary concern, it would have been possible to use a programming language such as *Java*, where each computer running the software has a special layer between the application and the operating system to transform the platform independent code of *Java* into platform specific instructions. However, as well as the fact I had more familiarity with *Delphi*, there are arguments in favour of using a programming environment which generates native code as this provides greater speed and stability, as well as a native look and feel to the application. *CenDiPede* (Garretson, 2010) was written in *Java*, and the additional layer of interpretation required, coupled with the limited range of data manipulation functions available in *Java* programming libraries, perhaps accounts for some of the challenges the system faces in terms of speed. *AntConc 3.2.4* was written in *Perl* through ActiveState's *PerlApp* (Anthony, 2011). This transforms *Perl* scripts into an executable file which can be run without *Perl* libraries needing to be separately installed on the user's computer, but according to ActiveState, the source code is parsed and compiled

when the program is run and the speed is the same as it is for the source *Perl* script (ActiveState_Software_Inc., 2012). Version 3.4.3 of *AntConc* uses “various compilers” for the *Perl* script (Anthony, 2014a), making the appearance slightly more tailored to each operating system, but it is worth noting that the download page states that this new version “runs a little slower” than the previous version (Anthony, 2014b). While suitable for small collections of texts, both versions of *AntConc* suffer from the same speed issues as *CenDiPede*, and perhaps the much narrower range of visual components available for *Perl* accounts for some limitations in its visual design. Compared to the visually rich Apps being used by learners for word processing, email, social networking and media management, *AntConc*’s icon-free, static, largely monotonal user interface seems lacking in this respect. As well as issues with the time it takes to perform searches and interpret the results, qualitative feedback on corpus research with learners often shows that they are concerned with graphics and colour (e.g. Henry, 2007).

RAD Studio (which includes *Delphi* and *C++* “personalities”) has a much wider set of sophisticated programming libraries with which to build procedures, including many visual elements which are not available in *Perl* or *Java*. Embarcadero’s *RAD Studio* transforms *Delphi* source code into native application code for each target platform. This means that the procedures make direct calls to the operating system itself and they run without needing to be re-interpreted. The version of the programming environment used for this project was limited to generating *Windows 32 bit* applications, but newer editions of the programming environment compile code for *Windows 64 bit*, and *Mac OS* with growing support for mobile devices (*Android*, *iOS*, etc.).

When beginning this project, it soon became apparent that *Delphi* had undergone many changes and enhancements in recent years. *Delphi Enterprise 2010* (2010) provides support for a wide range of database connections, as well as DataSnap technology to manage three tier architecture. The three tier approach as described by Swart is where the DataSnap server is a middle tier connecting client applications to the database server (Swart, 2009). A DataSnap concordancer application developed in *Delphi* would mean that the application running on each learner’s computer would not need direct database access, but could interface through the middle tier seamlessly. Client applications designed in this way are known as “thin clients” because they do not need to have all the database access routines compiled within their own file, and the benefit of having a reduced size, coupled

with the security and access management benefits, led to the decision to purchase a license for *Delphi Enterprise 2010*, and this software architecture was adopted.

With *Delphi* as a suitable solution for developing the refactoring application to transform text files into the database, and with its DataSnap library for developing the middle tier and client applications, the question remained as to which database system to adopt. While *Delphi* has libraries to connect to a variety of database systems including Embarcadero's own system, *MySQL* was selected for two main reasons. Firstly, I had some knowledge of *MySQL* from setting up and working with *Moodle* (a Course Management System; see www.moodle.org). Secondly, *MySQL* has a community version which meant that licensing was not an added complication, at least during the development phase. However, the *SQL* scripts used in the refactoring process do not use *MySQL* specific functions, and very little *MySQL* specific syntax, so migrating to an alternative would be feasible. *Delphi* can also be used to create web-server applications, so with heavier investment in server hardware, an alternative web-based version of the software could be developed. In addition, the *Delphi 2010 Enterprise* license includes provision for smaller datasets to be manipulated through *SQL* running on the user's client machine without the need to set up a full *SQL* server. Such an approach would seem to be a good way forward if the application were to be extended to allow the creation and storage of custom built corpora up to around 10 million words in size. For the larger corpora, with the database running on a server and the corpus files kept securely in the institution's server room however, *MySQL* was a good choice¹⁴.

In summary, the software architecture for the project runs using a three tier model with (a) a *MySQL* database server to hold several hundred million words of corpus texts, respond to *SQL* requests, and to store logs of student activity during the evaluation; (b) a middle-tier server written in *Delphi* to process requests from clients and pass them securely to the database server; and (c) thin clients written in *Delphi* which have no direct access to the database, but use the local machine to process and sort results for display. The refactoring application, also written in *Delphi*, completes the transformation of raw text files into statements to be inserted on the *MySQL* server, and this takes place before the corpus is made available to users.

¹⁴ Version 5.1 of the Community version of *MySQL* was used for much of the development of the project, and this was later upgraded to version 5.6.

Figure 3.2 shows the data pathways which are available once the corpus has been refactored and made available on the system. Starting with the client computers, requests are sent through DataSnap technology to the middle tier server which transforms these into *SQL* queries for a range of different features and contexts. The middle tier also acts as a gatekeeper, adding default limitations to queries if some parameters are not specified and guarding against certain kinds of potential threat from database hackers. For example, the middle tier can prevent hacking attempts known as *SQL* injection attacks when a user inserts malicious *SQL* commands inside an input string¹⁵. The middle tier server communicates directly with the *SQL* server, but the database user account used for this only has write-access to the database for the user logs, so no corpus data can be changed once the refactoring process has been completed. The *SQL* server retrieves the required data and performs calculations if required and then sends these data back through the middle tier to the clients. Many of the procedures made available to clients require several *SQL* queries to be run by the middle tier and the results are sent back as *OLEVariants*. These are then passed on to *ClientDataSet* components in the Client application and processed using standard database routines within *Delphi*. The client application converts the rows from these tables into data structures for components such as TMS Software's *AdvCardList* or *AdvGrid*, and these are displayed on the screen.

¹⁵ See Tahaghogi and Williams (2007, pp. 458-459) for a simple explanation and examples of *SQL* injection attacks; see RAD Studio 2010 help file (Embarcadero, 2010) for a brief explanation of the advantages of multi-tier applications.

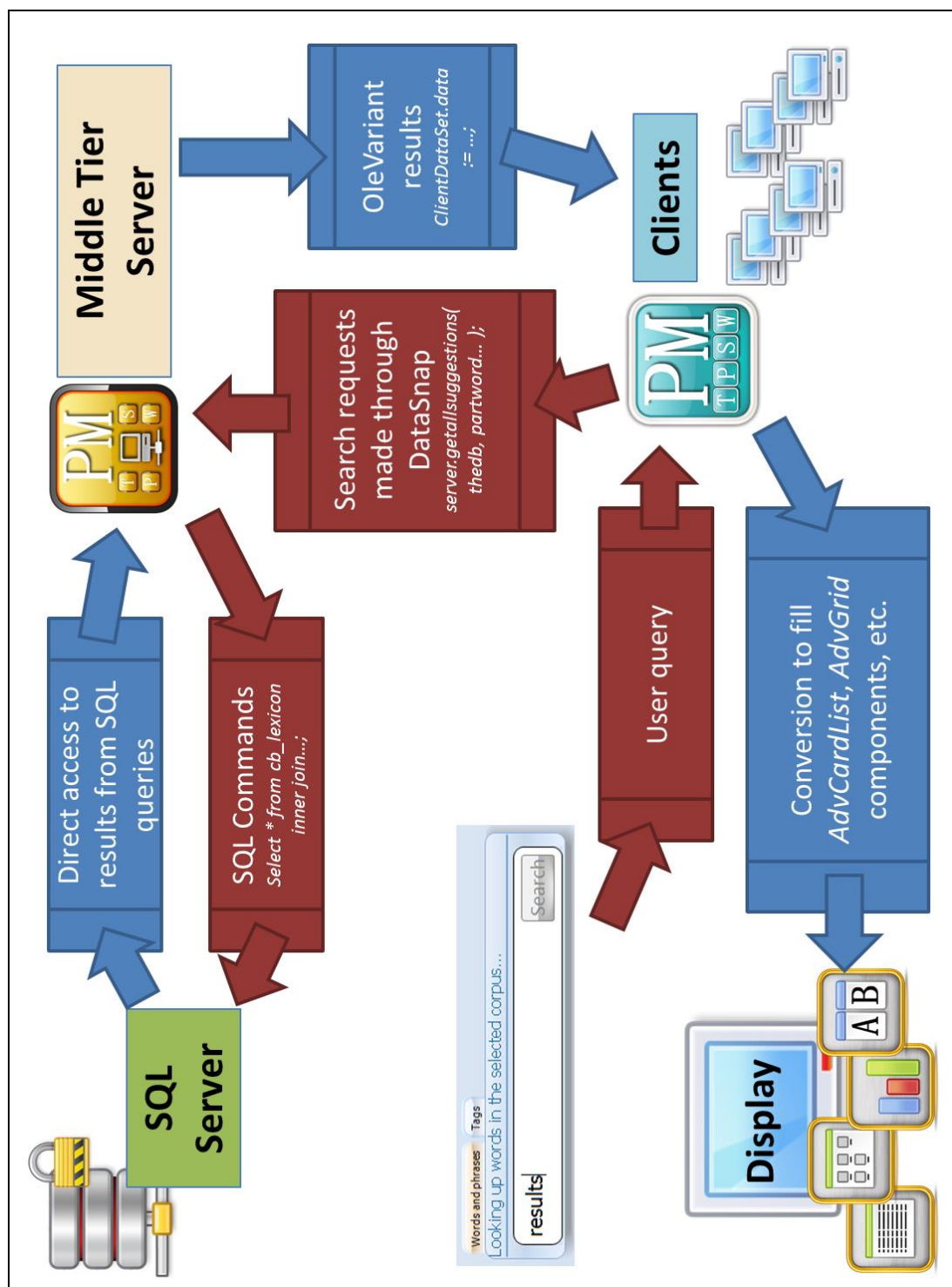


Figure 3.2: Diagram showing the data pathways for the three tier concordancer.

Having explained how the relational database serves this new concordancer system, the following section details the decisions made regarding how the data are stored in the database.

3.4 How do you store a corpus?

In this section, some of the basic design features of the database will be introduced.

Rather than providing a description of each column of each table in the entire database, this section will set out simply how texts are represented in the database. Details of the database structures used to hold additional information about the context and environment of each token, and summaries of tendencies for words and collocations to occur in different environments will be considered in later chapters as the function of these other data fields is explained.

3.4.1 Store data as they come

One option for storing texts in a database would be to keep the data in the original text files as much as possible. Indeed Sinclair (1991) presents a number of reasons for using raw text files which are still relevant to the design of corpus research tools today. As Sinclair argues, the decisions made regarding how text data are labelled and which linguistic theories are adopted can both limit the types of research which can be carried out by others, and also impose views of language which may mask discovery or extension of theories. Important theoretical issues regarding processes for parsing and tagging have been discussed widely (Meyer, 2002; Sinclair, 2004; Thomas & Short, 1996). While there is the danger of tagging text in a way which will simply propagate a pre-corpus view of language (Sinclair, 1991, 2004), there have also been some attempts to use corpus data as the basis for word categories used in tagging (Mindt, 2002). Sinclair argues that it could be important to consider the possibility of allowing even new categories of word class (Sinclair, 2004). Further points regarding the need for careful balance in the use of POS tags are introduced later in this chapter. However, although software designers need to be particularly mindful of such issues, for the current project the target linguistic theory was Lexical Priming (Hoey, 2005), and the range of calculations and statistics which the system needed to draw on meant that keeping files as raw text was not practical. Raw file storage would involve almost no pre-processing but would also mean the burden of transforming the data into new forms for manipulation and calculation would be transferred completely to the point in time when each user performs a search. While systems such as *CenDiPede* (Garretson, 2010) can allow sophisticated researchers with clear targets in mind to generate profiles and then query these data, for foreign language learning activities data need to be accessed much more quickly. Support for long text fields within relational databases is also fairly limited. Schwartz et al. (2008) explain that the standard options for full text indexes in *MySQL* only work on certain column types in one storage engine and

cannot include stop-words or words that appear in more than a certain proportion of texts. They suggest that full text searches on large datasets can also be unacceptably slow, and explain that it is possible to extend its range by adopting a storage engine with external functionality like *Sphinx*. Generating concordance data through a system like this would be possible, but the aims of the current project were different. Without additional information about the Lexical Priming features of each word and sentence in the corpus, searching back and forth within raw text files to determine the properties of each environment would be extremely costly in terms of time.

3.4.2 Store data ready for output

If storing data in a raw form would mean that gathering results would be too slow, another option would be to hold the data in the database in exactly the form they would be needed for the display of the results. In his conference paper at Corpus Linguistics 2011, Anthony, the creator of *AntConc*, hinted at this as he talked about the development of multipurpose corpora storage facilities (Anthony, et al., 2011). If web-based concordancers usually display KWIC results, it would be possible to simply store all the data in the database as strings of KWIC for each possible node word in the corpus. If KWIC lines with 5 word windows either side of the node would be stored in a database, it would mean each token would be stored in the corpus 11 times. Inefficiency in terms of duplicate data could be balanced against the ease with which KWIC lines could be retrieved. A complication for this approach in the current project, and perhaps a problem researchers would face if sharing a multipurpose facility, is that rather than just providing KWIC results, a wider range of output is needed for different applications. In order to help learners understand the context better and to create summary data, there needs to be a wider variety of ways to present the data. Increasing the window of text stored around the node to 10 words either side would mean tokens would be stored 21 times. It is clear that wider contexts such as whole paragraphs would lead to a tremendous increase in the amount of duplicate data held in the database, and this was not considered to be a suitable approach for the storage of corpus texts in the current project.

3.4.3 Store as XML

Some corpora are made available as XML files and this format brings the possibility of storing metadata and other information embedded in each corpus file alongside the language sample itself. Although different corpora have different XML elements, one approach could have been to enrich the existing XML files of each corpus by adding further information on the features of Lexical Priming being investigated. Rather than generating

XML profiles of a word to be stored separately from the corpus (Garretson, 2010), features could be marked up in each individual text. The structure of the BNC (BNC, 2007) shows how part-of-speech tags with word families can be wrapped around each individual token in the corpus, allowing concordancing software to search according to POS or lemma as well as type. However, it is worth noting that the indexes needed to quickly locate these different elements within a concordancer are not small. Even without the additional information about features of Lexical Priming, the indexes required for *Xaira*¹⁶ to access the BNC XML files are approximately 25% larger than the corpus itself (2007). One benefit of storing this information in a relational database rather than as a series of XML tags is that the database columns or tables linking to each token can act as both the data themselves and as indexes to those tokens. This information can be used to re-group or extract sub-corpora very efficiently using sub-queries and join statements¹⁷ in standard *SQL*. For this project, the aim was to develop a database which would allow manipulation of both the internal structural information of individual texts (as is available in XML), but also provide great speed and flexibility in generating summary tables based on common lexical environments for each target word or phrase. The use of summary tables to hold frequencies and strong tendencies of words and collocations to occur in different environments, and the functions used to measure these, are discussed in more detail in Chapters 5 and 6. It should be noted, however, that if the additional contextual information which is stored in the database for this project was to be exported for use in another system, XML and all the conventions associated with TEI guidelines (Burnard & Baumann, 2013) would seem to be the most appropriate standard to adopt.

3.4.4 Make the data as few as possible

Textbooks on database design set out principles whereby one aim is to normalize the data to avoid duplication, keeping the size of the database small and efficient. A simple example of one aspect of normalization of database schema is avoiding the repetition of supervisor names in a table of company employees by creating a separate table for the names of all the supervisors and using a numerical key in the employee table. This has the benefit of making the *SQL* instruction to generate a list of unique manager names a “trivial” task,

¹⁶ *Xaira* is the concordancing software which is distributed with the XML edition of the *BNC*.

¹⁷ Joins are a standard *SQL* operation to combine fields from a multi-table structure where data have been organized to conform to what are called “normal” principles whereby data are not held multiple times in a large table, but divided across tables using one-to-many or many-to-many links.

rather than needing to work through the whole employee table and match duplicates and then generate a list (Schwartz, et al., 2008). While computer storage and processing capabilities have increased by several orders of magnitude, with increased size and power has come expectations for larger amounts of data, so Sinclair's warning of the speed cost of "grossly overstuffed" marked up texts (Sinclair, 2004, p. 191) is still relevant today. Just as the database design textbooks take examples of the relationships between employees, line managers, departments and divisions or of songs, albums and musical artists as ways of explaining how to store information in separate tables to increase efficiency and speed (e.g. Beighley, 2007; Schwartz, et al., 2008; Tahaghoghi & Williams, 2007), these principles can also be applied to lexical items, words, sentences, texts and corpora. As shown in Figure 3.3, the occurrence of each individual word of each sentence in the corpus (i.e. each token) is the smallest item held in the database and the lexical item for this token is stored as an integer, pointing to a unique primary key for each type. Similarly, the sentence ID number links each token to a row in the table containing information about each sentence in the corpus. From the sentence table, a link is made to the "corpus_texts" table, and from there the "corpus_id" key points to a row in another table containing information about each corpus.

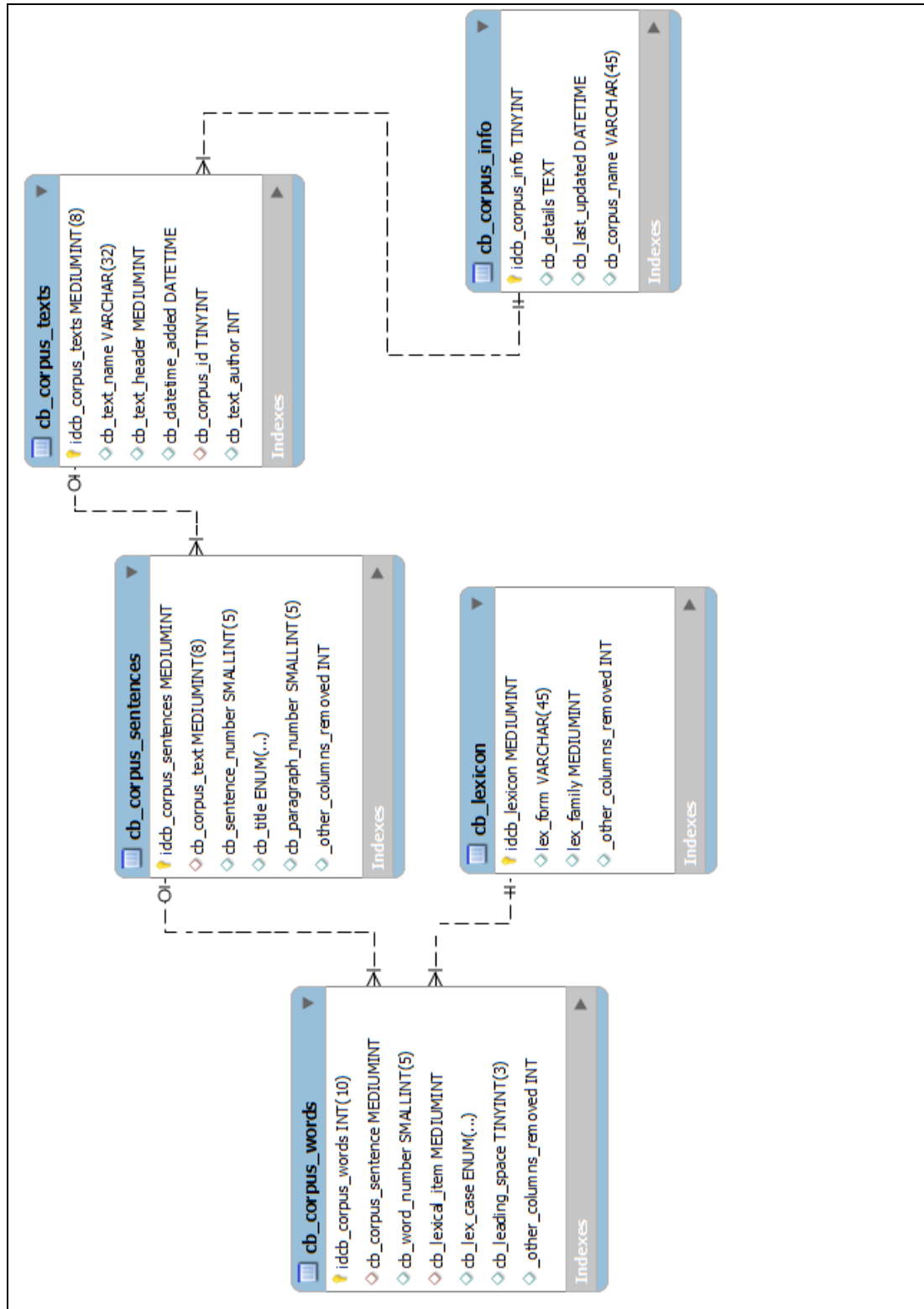


Figure 3.3: Simplified schema for ranks of items in the corpus database; note: “_other_columns_removed” indicates that some columns used in the full schema have been removed from this diagram to clarify the relationships being described.

The storage engine chosen for this project was initially InnoDB, but later it was changed to MyISAM. InnoDB storage engines have many advantages for some situations in terms of the way the indexes work (Schwartz, et al., 2008). MySQL MyISAM tables rely on the operating system to cache the data, so less disk access is required to retrieve frequently accessed rows, but indexes are cached by the server (Schwartz, et al., 2008). Therefore, for some of the pre-processing and for fast access to some of the other data, covering indexes have been designed, meaning that the index includes all the columns of data needed from a table for a specific query type. This means that the choice of MyISAM as the storage engine can lead to larger indexes and potentially more disk space on the server being used than would be required for InnoDB. Part of the reason for the change to using MyISAM rather than InnoDB, however, is that the size of the database changes considerably as it is processed from its initial state, so storing all the data for all databases in an InnoDB file which is not automatically shrunk means that a considerable amount of sparsely stored data can artificially increase the size of the data file on the server's hard drive and limit the effectiveness of operating system file caching. Because the data in the corpora are to be static when opened for users to access, there is no need to make use of the transactional capabilities of InnoDB, and the simplicity of being able to view the data structures within the operating system for MyISAM tables is also considered to be beneficial. One disadvantage of MyISAM tables which had to be considered in the software design is that they do not support foreign keys and so rather than being able to rely on the database server itself to check the integrity of primary key values in other tables, the application has to check that the links to other tables are valid (Tahaghoghi & Williams, 2007).

The way in which information about the author of each corpus text (shown in Figure 3.3 as an unlinked integer) and other metadata is stored and used in the database is described in Chapter 6. Given that the "corpus_words" table includes a unique sentence ID and a word number which is unique for each text, it would have been possible to construct a *natural key* as the primary key for this table rather than a separate integer value. However, the complications of the refactoring process described in more detail below, coupled with the range of ways in which rows in the table are accessed for the generation of statistics, meant that a decision was made early on to keep the primary key separate.

Looking at the "corpus_words" and "lexicon" tables in more detail, it can be seen that the design works with types rather than lemma, or looser groupings such as *word families*. The "lex_case" field permits a variety of different states for capitalization, so, for example,

“China”, “china” and “CHINA” all point to the same item in the lexicon. There are, of course, disadvantages with this, since the collocations, lexical priming tendencies and KeyTags are set up to work with the lexicon ID, and are therefore case insensitive. One reason for working in this way was that it was not anticipated that language learners using the system would want to compare forms of capitalization, and since position in sentence was one aspect under investigation, splitting capitalized and lowercase instances of a type would lead to confusing results. The decision to work with types was also influenced by the *Birmingham school*, where Sinclair and Hoey have shown that different word forms can have very different collocates and primings (Hoey, 2005; Sinclair, 1991). From the related field of information retrieval, the advice for search engine designers is to use types if there is sufficient storage capacity (Croft, et al., 2010). Nevertheless, search engines often expand queries by merging results for different word forms and tenses since they are primarily topic oriented rather than focussed on the linguistic differences which learners are wanting to explore. However, there are pedagogical reasons for preferring types. Learners of English from countries like China often find attention to word form challenging. A concordancer for learners should not be too pedantic by requiring exact word forms as if it were a password, but at the same time in a context where learners struggle with producing correct word forms, rather than clumping different forms together, it is likely that drawing attention to differences through side-by-side comparison would be helpful. For example, students frequently use *analysis* and *analyse* interchangeably in both presentations and reports without attention to word form, but seeing concordance lines for these side-by-side should help them select the appropriate form. In a similar way, tense and aspect can be a problem for Chinese students, so presenting summaries of differences between “analyse” and “analysed” could be a helpful way to remind them of the importance of marking tense in English. The design of the side-by-side compare mode in *The Prime Machine* is introduced in Section 3.6.2.

Although most of the scripts used to pre-process the corpus operate using types, there are a few routines which make use of links between words from the same word family. For these purposes, the lexicon table includes a `lex_family` column, which points to the ID of the shortest lexical item in the lexicon which matches the stem of each word. The stemming process is a simple implementation of Paice’s term conflation rules (Oakes, 1998; Paice, 1977). However, the *SQL* command to form these links could be adapted to use an alternative stemming algorithm based on different conflation rules or even a wordlist.

One further aspect of the lexicon table worth noting is the treatment of punctuation. Punctuation marks could be handled in three different ways; they could be ignored completely, treated in a special way, or treated like any other type. Ignoring punctuation did not seem to be helpful as treating it like white space would not permit the extraction of phrases which typically contain specific marks. The second option could have been adopted, but would have over-complicated many calculations and would probably require the users to be more mindful of punctuation as they query and explore the results. The third option was selected as it meant the co-occurrence of punctuation marks would need to “compete” on equal terms with all other types, but strong patterns would be made clear to the user. As such, a string meeting threshold requirements like *knock-on*, for example, is treated as three items in the corpus, but will appear as a collocation (see Chapter 4). The only exception to this is the handling of the apostrophe in contractions and as a marker of possession, as these are identified through pre-processing and *CLAWS* and stored with the part of the word as determined by the default rules in *CLAWS*.

Another early decision was regarding the range of integer values to permit for each of the various keys. *MySQL 5.1* has several column types for integers, ranging from those limited to values of 0 or 1, taking up one byte of storage, to “BigInt” integers which can hold values in a range just above $\pm 9.2 \times 10^{18}$, but require 8 bytes of storage (Tahaghoghi & Williams, 2007). With the schema described above, the size of these ID columns has a large impact on the overall size of the database since every token in the corpus holds its own ID, the ID of the lexical item in the lexicon and the ID of the sentence in which it is located. However, using integer values with ranges which are too small would mean that the overall size of the corpus and the number of types would be too limited. Word type estimates from the field of computer science (Croft, et al., 2010), as well as statistics from information about datasets such as *the British National Corpus* (Burnard, 2007a), led to the decision to use INT values for the tokens in the “corpus_words” table, MEDIUMINT values for most of the other keys, and TINYINT for the corpus ID. Originally, the plan was to keep all the corpora together in the same database, and so a primary key with a range of 0 to 255 seemed more than sufficient. However, as development moved from small test corpora to larger datasets of many millions of words, the speed began to slow down considerably, and this, coupled with the increasing amount of pre-processing required, meant that it became more sensible to split the corpora up into separate databases. A corpus administration database was added which contained text strings of the database names, providing easy control of access to different corpora as well as enabling them to be set in different

“states”, for maintenance, upgrades, etc. The separation of corpora into different databases on the same database server does not affect the experience of the end user; for a student using the system, all the corpora which are available appear in a single menu even though they are actually stored in separate databases.

3.5 The refactoring process

Johnson et al. (2007) use the term “refactoring” in their evaluation of the process used on the Protein Design corpus to change it from its initial structure into two additional formats. The term “refactoring” has been adopted in this thesis to refer to the process of transforming existing corpus files into rows in the database. Having made a decision about how the data would be stored in the database, the development turned to the refactoring process required to transform the corpus texts from their initial, raw form into rows in the various tables of the database.

3.5.1 Corpus formats

Corpora come in many different formats; while the newest corpora are likely to be made available to researchers in XML, several useful corpora from the past are still most widely available in SGML or plain text. Some large datasets primarily aimed for text mining applications in computer science may be (intentionally) made available in a very raw form, with compensating for “noise” and lack of structure part of the stated aims of the mining activity. However, other corpora like the *British National Corpus* have undergone several upgrades and transformations in order to keep up with latest standards (BNC_Webmaster, 2007). As well as variation in the amount and range of metadata, expectations of dataset providers with regard to character encoding at a more fundamental level have also undergone many changes in recent years. While Unicode brings opportunities to be more flexible it also adds a further layer of possible incompatibility between computer systems, especially when drawing on resources such as *CLAWS* which were developed as standards were changing. Early versions of *CLAWS* could not accept accented or special characters, and by *CLAWS* version 4, the industry standard was to accept plain ASCII and special characters encoded in SGML (Garside & Smith, 1997). One aim of this project was to ensure that development for languages other than English would not be too complicated and to make use of symbols such as opening and closing quotation marks (rather than just straight quotation marks) which are familiar to users viewing word-processed documents and emails. However, at the present time these require specific character encoding for

Unicode in order for them to be manipulated in software. Some of the measurements of Lexical Priming features rely on part-of-speech tags for identification, so a POS tagger such as *CLAWS* was needed. The refactoring process needed to take into account both the input demands of the *CLAWS* tagger as well as the output demands for insertion of rows into the database. It also needed to find ways to “cloak” metadata which *CLAWS* would otherwise simply discard, so that the post-POS-tagging process could draw on some of the raw data after POS tagging had been completed.

Given that this project was not going to focus heavily on POS tags, it may seem strange that a POS tagger was chosen for sentence segmentation and for identification of features such as Theme, passive voice and determiners. In fact, although the refactoring application would need further customization if a different approach was used to detect this information, alternative methods or taggers would be possible. *CLAWS* was state-of-the-art at the time of its development and it makes a good candidate for this project since its approach has been used to develop similar taggers for a number of other languages, and since it was written in *C*, it would not be too hard to imagine the future development of an integrated refactoring system written in *Delphi* which draws on the *C* programming language algorithms or sub-routines of *CLAWS* and is able to process text more quickly and more efficiently. *CLAWS* is also well known and therefore its strengths and limitations are also well known, and this was an important factor in adopting it. Additionally, having decided to use the *BNC* as one of the corpora in the system, *CLAWS* was considered to be an even more logical choice, since the XML version of the *BNC* already includes *CLAWS* C5 tags and straight-forward conversion can be made to find equivalent C7. Garretson (2010) argued for the need to take things even further and use tree relationships in order to analyse relationships between collocations¹⁸. The Word Sketches in *The Sketch Engine* (Kilgarriff, et al., 2004), also use tree relationships. However, both of these approaches take the user to complicated lists of results, and seem to make the patterns derived from tree relationships the primary object of investigation, rather than the raw data themselves. While exploration of POS tags or tree relationship data can be useful for certain kinds of linguistic research, within this project the aim was to store additional information about the contexts in which tokens or sentences occur, and then allow filtering and exploration of

¹⁸ Both *CenDiPede* and *The Sketch Engine* use a parser to annotate sentence structures using Dependency Grammar. Thus words in each sentence are tagged as being the head of a group of words, or dependent on other items in the sentence following this grammatical structural analysis.

the actual concordance lines, keeping these central to the user's experience. As explained in Chapter 5, *CLAWS* POS tags provide a means by which to identify features such as passive voice and to distinguish Theme from Rheme, and therefore are sufficient for the needs of this project.

Files read by *CLAWS* need to follow a strict set of requirements, including how special characters should be encoded, how prohibited characters should be removed and how the beginning and end of text sections should be marked. A further complication is that *CLAWS* flags unequal opening and closing quotation marks as an error. This was particularly challenging in the *Financial Times* corpus, since the same character is used in this corpus as an apostrophe, an open quotation mark, a closing quotation mark, and the opening and closing of quotations inside quotations. As shown in Figure 3.4, both *the Guardian corpus* and the *Financial Times corpus* use the apostrophe (ANSI character 39) as a quotation mark.

<p>'You'd think she'd be here somewhere, wouldn't you?' [Guardian 1990b: 10]</p> <p>'I have been authorised to say the following: 'Most people in the company would accept that Maurice has a value but not as chairman and not on the board'' [FT944_9]</p>
--

Figure 3.4: Examples of the use of the same symbol for apostrophes and quotes in *the Guardian* and *Financial Times* corpora

Prior to being fed into *CLAWS*, the refactoring application runs through a procedure to differentiate between quotation marks and apostrophes, being able to cope with the examples shown above, as well as instances where the quotation continues into the next paragraph but the apostrophe marking the end of the quotation does not occur. These quotation marks are transformed into Unicode for “smart quotes”, or in cases where the rules cannot automatically determine whether they should be interpreted as opening or closing quotation marks, they are left as Unicode characters for the apostrophe, and therefore treated as a “special character” by *CLAWS*. Another function of this process is to ensure that any “<”, “>” or “&” symbols in running text are encoded correctly in XML in order for *CLAWS* to accept the file for POS processing.

The output from *CLAWS* is three separate files. The main file is a vertical format list with each word or punctuation symbol on a new line, along with POS tag, and other statistics. Words longer than the allowed number of characters and any other tags or sections of the file outside the <text> </text> nodes are output to the supp file. Clearly, for a concordancer needing to access paragraph, heading and other information, being able to

integrate this back into the dataset is vital. The third file is an error file and *CLAWS* produces errors for a wide range of situations. While these error codes were very useful as the refactoring application was being developed, they have little relevance for a large corpus since individually analysing each one would take too long. Fortunately, since the use of POS tags was relatively conservative, most remaining errors would not be important. After *CLAWS* has finished tagging a file, the refactoring application opens each error log file and searches through to see whether any error types are found which are considered critical. If so, the file is marked as unsuccessful, and the first critical error found in the error log appears in the “error” column of the refactoring spreadsheet log. This log is simply a list of raw and output files, which enables the administrator to see if any raw files from the corpus have been rejected, and is especially useful for the development of refactoring rules for different corpus file formats and encodings.

Another issue which had to be considered was the way *CLAWS* treats punctuation and whitespace (new lines, tabs and spaces). These provide clues to the formatting of the text, and so it was important to try to encode them in a way which would mean they would not be lost once the texts were passed through *CLAWS*. The key to processing text files with *CLAWS* without losing text layout and special character information was to package the information up in a way which *CLAWS* could accept and then decode the information once it appeared in the *supp* files. Since the main focus of this project was to build and evaluate a concordancer, the software for the refactoring application, while important, was developed on a more pragmatic basis. By using custom-made milestone XML tags, and Unicode characters such as empty space (ࠏ), it was possible to “cloak” this whitespace in a way so that *CLAWS* would pass over it but the original layout could be retrieved from the output or supplementary file. Working with the XML files directly for the *BNC* refactoring process was not without its complications, since the XML reader component adopted (*NativeXML*) was designed to remove whitespace at the end of elements. A small enhancement was made to the source code of this component since handling whitespace as a separate element in *NativeXML* did not seem very straightforward, and would have over-complicated the process.

As can be seen from Table 3.1 below, the refactoring application had to deal with a wide range of standards. Looking to the future, it is likely that corpus resources will converge towards more standard formats, but nevertheless, different corpora have a wide variety of metadata and other structural formatting which arise from the different research foci, the

different sources and sampling methods used to collect the texts, and the different purposes which the distributors or publishers envisage users will have for the data.

Corpus	Raw format	POS and sentence markers in raw files	Compatibility with <i>CLAWS</i>	Post-tagging approach
<i>BNC</i>	XML	C5 Tag codes for each word	N/A	XML files can be manipulated directly
<i>SpringerOpen; Hindawi corpora</i>	XML	No tagging or sentence segmentation; no <text> tags	Transform XML into CLAW-readable form	Work through <i>CLAWS</i> "supp" file searching through <i>CLAWS</i> tag file.
<i>Financial Times</i>	SGML without attributes	No tagging or sentence segmentation	Transform original SGML into <i>CLAWS</i> -readable form	Work through <i>CLAWS</i> output file, referring to supp file
<i>Guardian</i>	SGML with attributes	No tagging or sentence segmentation	Transform original SGML into <i>CLAWS</i> -readable form	Work through <i>CLAWS</i> output file, referring to supp file
<i>BAWE</i>	XML	Some sentence and paragraph tagging ¹⁹ ; no POS tagging.	Transform range of text and heading formatting; Remove numbering from sentence and paragraph tags, and allow <i>CLAWS</i> to perform additional sentence segmentation.	Work through <i>CLAWS</i> output file, referring to supp file
<i>WECCCL</i>	Ad-hoc tagging system; not XML	No sentence segmentation; no POS tagging	Transform ad-hoc tags into XML ²⁰ (e.g.	Work through <i>CLAWS</i> output file, referring to supp file

Table 3.1: The format and refactoring required for various corpora

¹⁹ The sentence segmentation algorithm is clearly explained in the *BAWE corpus* manual, but manual examination of the files reveals problems with unexpected breaks due to page numbering in citations and some missing breaks between sentences.

²⁰ For example, a single <TIMED> was used in the raw text files to indicate task type; but this would be considered a structural error in XML since there is no matching closing tag. This tag would be transformed using the pre-processing tool into <TASKTYPE>Timed</TASKTYPE>, which can be read (and ignored) by *CLAWS* and then imported as metadata into the corpus database.

3.5.2 *The Prime Machine* corpus refactoring application






Managing the transformation process of text files in other concordancers can be a complicated and time-consuming operation. As has been explained, *The Prime Machine* has been designed as a system where the corpora which students and teachers use at an institution will have already been determined in advance. In this context, the refactoring application is a standalone piece of software which takes a corpus database administrator through the stages required to design transformation rules for the processing of text files before and after they are tagged by *CLAWS*. Some corpora require additional file splitting or text processing to enable the corpus files to be imported as XML. The software also includes features allowing nodes and other elements of the XML files to be manipulated before and after the files have been processed by *CLAWS*. With additional settings and lookup tables available for the identification of text categories (see Chapter 6 for more details) the software has a comprehensive but complicated range of features. In order to simplify the process, navigation through the stages are facilitated through the on-screen step-by-step guide, and templates can be stored and retrieved to hold all the settings for all the processing of a specific type of corpus. Figure 3.5 shows the main screen for the refactoring application with some of the available templates visible in the centre, and the step-by-step bar visible at the top.



Figure 3.5: The main screen for *The Prime Machine* corpus refactoring application.

Table 3.2 shows the text processing features and the corresponding icons which are available and gives some details about some of the steps in the refactoring process.

Table 3.2 The main steps in the refactoring process leading up to the importing of SQL dump files into the database server.

	<p>Convert text files to XML</p>	<ul style="list-style-type: none"> • Split large files containing multiple texts into separate files (<i>Financial Times</i>); • Perform copy and replace operations on files as text, to make texts XML compliant (<i>Guardian, WECCL</i>), or to simplify attribute elements inside XML tags (<i>BAWE</i>).
	<p>Select transformation rules</p>	<ul style="list-style-type: none"> • Create a list of tags from one file or a collection of files; • Assign pre-CLAWS and post-CLAWS actions to be taken when the tags are processed.
	<p>Confirm settings</p>	<ul style="list-style-type: none"> • Enter basic details about the corpus; • Load a lookup table for categories (<i>BNC, Hindawi</i>); • Make other advanced changes: <ul style="list-style-type: none"> ○ Customize acceptable CLAWS tags and text replacements for pre-CLAWS processing; ○ Make changes to the POS tags used to identify features used for Lexical Priming tendencies (see Chapter 5); ○ Customize handling of quotations.
	<p>Choose files</p>	<ul style="list-style-type: none"> • Specify the location of the folder containing the files to be processed; • Specify the location of the <i>CLAWS</i> application file.
	<p>Complete processing</p>	<ul style="list-style-type: none"> • Apply the transformation rules to all the files and create SQL dump files to be imported into the database.

As can be seen, the corpus refactoring application takes raw corpus files in a variety of different file formats and transforms them into SQL dump files. During the early stages of the project, once test corpus texts had been refactored, a decision needed to be made as to how the data would be imported into the database. Before using SQL dump files as output, an attempt was made to directly insert rows into the database as they were processed, with an early version of the *Delphi* refactoring application connecting directly to the database server. Working directly with a live database meant that the corpus lexicon, for example, had to be extended as each new word was encountered. The code used in February 2011 was very inefficient and took around 18 hours to import 1 million words.

However, it quickly became evident that the speed with which *MySQL* is able to match columns of text to the lexicon table is many degrees of magnitude faster than for a separate application to make individual *SQL* queries. This is because *MySQL* is able to

optimize matching using a range of approaches, while individual *SQL* queries need a connection to be established, the command sent, the query parsed and then the results processed and returned. Backing up and restoring the database using *SQL* dump files was much faster. The speed with which *MySQL* is able to import *SQL* dump files was impressive, so it became clear that rather than producing a set of *SQL* insert statements, updating each table at the same time, a more efficient approach would be to output the corpus texts in a form mimicking the dump files, and this is the approach which is taken in the newer refactoring application. It should be noted that the data in the *SQL* output files from the refactoring application are not in any way “normalised”. This means that the string of letters for each word appears as text rather than a numerical identifier linking to the lexicon. The data in these files are many times larger than the original files since they contain all the additional information about the context of each word and sentence without any optimization. However, transforming the database from its imported state to a more efficient form was accomplished using a set of specially designed *SQL* scripts. These scripts can be run on the server once for each corpus, and automatically generate the optimized and more “normalised” tables. Thus, while Rayson for the development of *WMatrix* needed to concern himself with the development of an efficient algorithm to identify word types (Rayson, 2002), in this project the request is made to the *MySQL* server in *SQL* script, and the optimization of this process is completed automatically by the server itself. Figure 3.6 is a diagrammatic representation of the refactoring whole process.

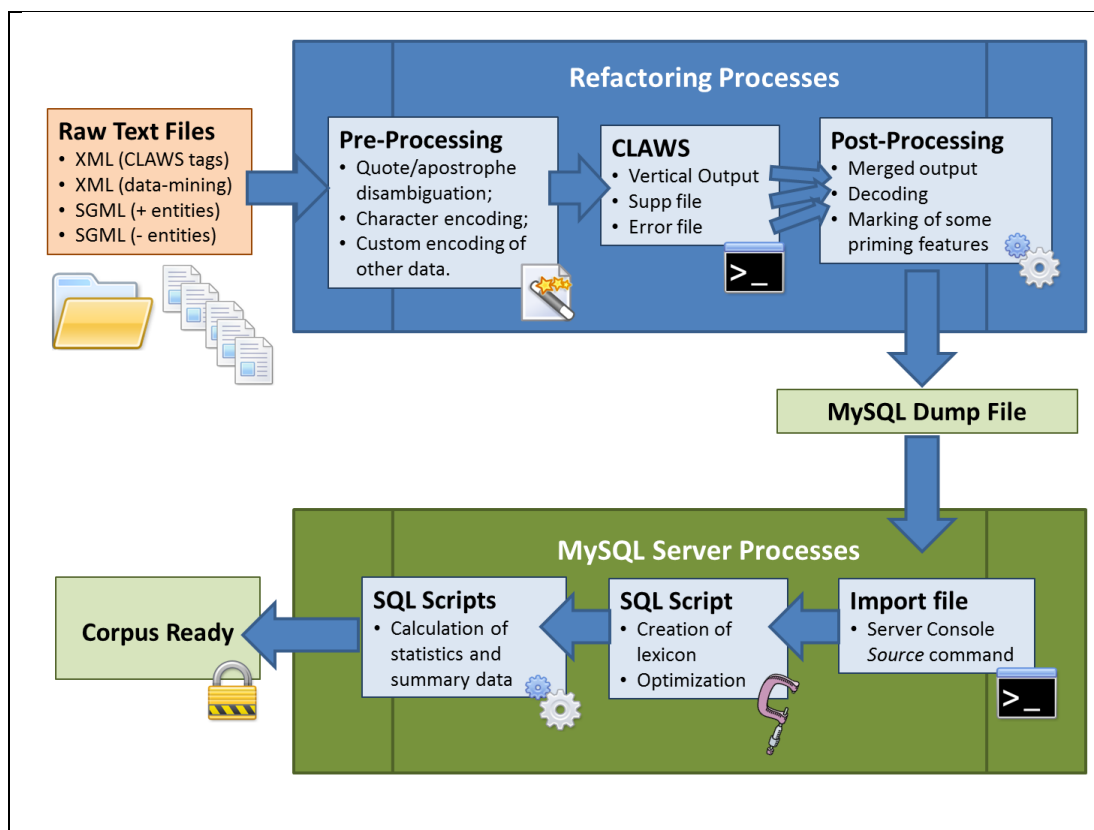


Figure 3.6: Diagram showing the processes involved in transforming raw text files for a corpus into its final state.

3.5.3 Summary tables

As the database optimization process was developed, it became evident that since the data in the database are fixed before learners start to access them, pre-processing of a wider range of features and using summary tables would be an excellent way to relieve the server from performing repeated expensive queries. From a strict database design perspective, storing additional information about the context of each word, based on features of Lexical Priming, as columns in the tables containing the tokens and sentences, however, breaks what is called the “normalization” of the schema since they are data which have been derived from other data in the tables. A break from fully normalized schema was made in terms of the mark-up of individual words and sentences in the corpus, so this additional information about the context could be retrieved for an item in its own row in the database. The current version of this makes use of bitwise values to compress the “on or off” status of a range of features within a small number of bytes for each row in the table. Since many of the calculations needed to identify the contextual environment of items are expensive in computation terms, use of additional columns in the table to hold this kind of derived data seems to be the only practical way of generating statistics at an acceptable speed.

Furthermore, it is recognised that summary tables are considered to be a good way to improve performance for data where they can be a little stale (Schwartz, et al., 2008). As mentioned above, corpus data are typically fixed, and so since texts are not being edited or changed, the data do not become stale. One of the simplest summary statistics held in the database is the frequency of words and size of the sentence contexts and text contexts where they occur. Summary tables are used for several statistics in the system and increase the amount of storage space required on the server's hard drive significantly. During the development of the *SQL* scripts used to generate information about these tendencies, the need for summary tables was not so obvious for test corpora of 2 million words as complex queries could be calculated in milliseconds. However, when a single calculation on a larger test corpus took up to a second or longer, the scalability of an on-the-fly approach became a concern. For a corpus like the *BNC* with 100 million words, a simple request to a dedicated *MySQL* server to pull out the words in the windows for a node which has a frequency of 100,000 can take several seconds. If this process is then repeated for several hundred thousand types in the corpus, the extraction process, without any statistical testing or storage of results, takes many days. By breaking away from on-the-fly processing, a whole range of feedback mechanisms also becomes possible, and a relatively modest server system becomes able to serve larger numbers of concurrent users. However, the cost is that alternative statistics or calculations on extremely frequent items is forced to the background, and indeed while the *SQL* commands required to generate on-the-fly results are not any more complicated than the pre-processing scripts, a "print queue" approach would seem the best option, whereby users could put a request for an "expensive" query into a queue, but the server would prioritize fast and indexed queries from other users. Currently, this functionality is not available in the system, and users are restricted to using the statistics and many of the settings which have been configured either at the time the corpus was refactored, or when the server was set up.

There is further discussion regarding the tendencies of words and collocations to occur in contexts related to features of Lexical Priming in Chapter 5, where the use and display of these data are also introduced. In the remainder of this chapter, however, some of the simplest sets of information, which are held in indexes or summary tables and provide several solutions to the requirements for a learner concordancer, will now be explained in terms of the user-interface.

3.6 Design of the user interface

3.6.1 Query syntax, auto-complete and spelling support

One of the first considerations regarding the user interface for a concordancer is what kind of query syntax will be used. For example, some concordancers allow users to enter simple wildcards, *AntConc* offers Regular Expression support and *The Sketch Engine* requires the user to specify the part of speech for Word Sketches. Within the context of a concordancer specifically for language learners, however, the main function of wildcards would presumably be to allow different morphology to be conflated into the results, while the role of POS tags in query syntax would be both to limit the data and to allow searches for patterns in multi-word units. For heavily guided tasks, wildcards and POS tags may be useful, and Regular Expressions may provide powerful ways to query concordance lines in the classroom. However, Stevens (1995) suggests that near native speaker knowledge is needed in order for corpus users to create new wildcard queries accurately. Qiao and Sussex (1996) argue that learners should be trained in understanding POS tags in order to perform successful searches. Given the popularity of search engines among students, it would have been possible to develop a query language for the concordancer following the operators commonly available for web searches, such as “AND” to merge results, “OR” for alternatives, “+” for required elements and “-” for elements to be filtered out. However, research on search engine user behaviour has shown that the vast majority of queries do not include any of these operators (Croft, et al., 2010). For Chinese users in particular, it seems that they are extremely rare, perhaps because of the fact that many of the operators are actually English words (Chau, Xiao, & Yang, 2007). It is also argued that neglecting operators, as most web users around the world do, is reasonable behaviour since the quality of results has been shown to be similar whether complex queries are used or not (Eastman & Jansen, 2003).

Further consideration is needed with regard to developing suitable query syntax for the exploration of the lexical primings of words. Garretson (2010) developed a new query language for the profiles his software generates so that highly specific contexts of words can be queried directly. His approach is powerful, but it is also very complicated and it was felt that for language learners a complicated query language for Lexical Priming information would be too much of an investment in learning about the software before they could get started with it. Therefore, rather than expecting the user to make any of these decisions before composing an all-embracing perfect search query with wildcards, regular expressions, POS tags or priming filters, a different approach was adopted for this

project. The system for simple queries is explained below, while the navigation and display of the main features of Lexical Priming for this project is described in Chapter 5.

The query box in this project is designed to aid the user in choosing between similar strings of words and forming correctly spelled queries through auto-complete as they type. Auto-complete is a well-known and familiar experience for users of modern websites and computer programs. It is particularly useful for database applications where the range of acceptable options is limited. Figure 3.7 shows how as the learner starts to type a search term into the box, the words in the currently selected corpus with the same first few letters appear, displayed in descending order of frequency.

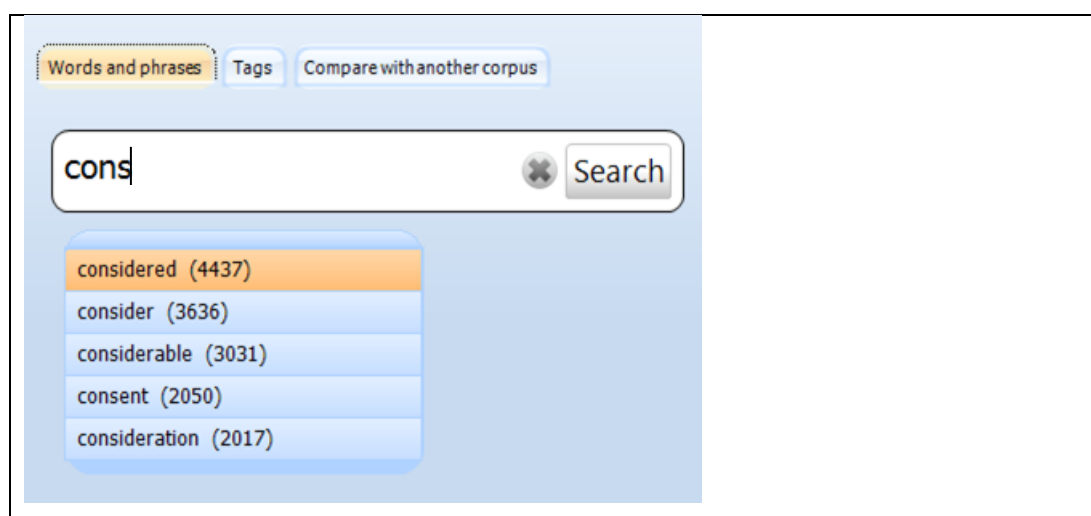


Figure 3.7: Screenshot showing auto-complete support for a query

As soon as the auto-complete list is empty, the client application “knows” that the word typed by the learner is not in the corpus and adds this to a locally held list of items with no concordance results. There is a delay between starting an auto-complete request process, the server receiving, processing and sending back the list of words and frequencies, and the display on screen, and while this is only a second on a fast local area network, it is certainly time for a faster typist to enter a word and move on to the next, or to hit the Enter key. As the space bar is pressed, in the background, the application sends another request to get the ID number of the lexical item in the database. If no record with that word form is found, the word is also added to the list of words not occurring in the currently selected database. Stevens (1991) highlights the importance of spell-check and spelling support in concordance software, noting that students reported frustration if a search was empty.

Although spell-checking functionality is built into newer web browsers²¹, students using online concordancers will not be provided with corpus specific support and in a foreign language learning context are likely to have a browser running a dictionary for a different language from the one which they are analysing in the concordancer. For software development, spell checking is a feature which is available as an add-on component for programmers, and there are several packages to choose from. At an earlier stage in the development, the possibility of holding the corpus lexicon locally as a spell-check dictionary file was considered, as well as the use of a standard spell check dictionary as a gatekeeper for search requests. It should be remembered that most corpora contain spelling mistakes and may also contain highly specialist or rare words, so restricting searches to words in a standard spelling dictionary did not seem appropriate. Instead of using the dictionary function of a spell-check component, the customized event handling feature of a text input component was used to make use of its curly red-line display and to create a spell-check process which balanced transmission of data to and from the server against providing relevant data for each specific corpus. When the search button is pressed, the application works through all the words in the search box and ascertains whether information is held locally about these types and their frequencies in the corpus. If there are any words where information is not known, they are packaged together and a single lookup is requested to the server before the client application moves on to the “please wait” screen. When the results come back, if there are any words not in the currently selected corpus, these will be stored in memory as misspellings and the spell-check procedure will mark a red line under the first misspelled word in the text box. However, when the user right-clicks on a word with red-curvy lines, unlike with a spell-check component’s default methods, two further database look-ups are requested. First, the system requests a list of words in the lexicon with the same *SoundEx* value, again ranked in descending order of frequency. *SoundEx* is a function in most database systems which converts English words into a code based on rules which ignore vowels and group similar sounding consonants together (Croft, et al., 2010), and many English language spell-check systems use *SoundEx* as the main method for ranking suggestions. Other spell-check ranking systems may also make use of “Edit Distance” (Jurafsky & Martin, 2010), and an edit distance function could have been written for the *MySQL* server. However, while *SoundEx* is a fast mapping of letters to a list of numbers which is executed at high speed, obtaining the edit distance requires the

²¹ *Microsoft Internet Explorer* has had spell-checking for user input since version 10; most other browsers also have spell-checking.

comparison of each candidate word against the misspelled string and therefore is much more demanding on the server. It was decided that *SoundEx* and the auto-complete functionality should provide sufficient support to users without needing to incorporate other spell-check algorithms.

Another principle of the design of the query interface was that learners should not be expected to know which the best corpus for a particular search would be. In order to try to avoid the possibility of a learner entering a search word in one corpus and not knowing that another corpus actually contained this word, the spell-check assistance routine also checks the string they have entered in each of the other available corpora. The corpora containing this word and the frequency in that corpus appears in a special box to the right of the screen.

Figure 3.8 and Figure 3.9 show how the results of these two spelling-related features appear on screen, with the first showing spelling suggestions based on *SoundEx* for “consequence”, and the second showing how a rare word such as “polymorph” is found in two other corpora.



Figure 3.8: Screenshot showing *SoundEx* suggestions for “consequence” in the *BNC: Academic sub-corpus*.

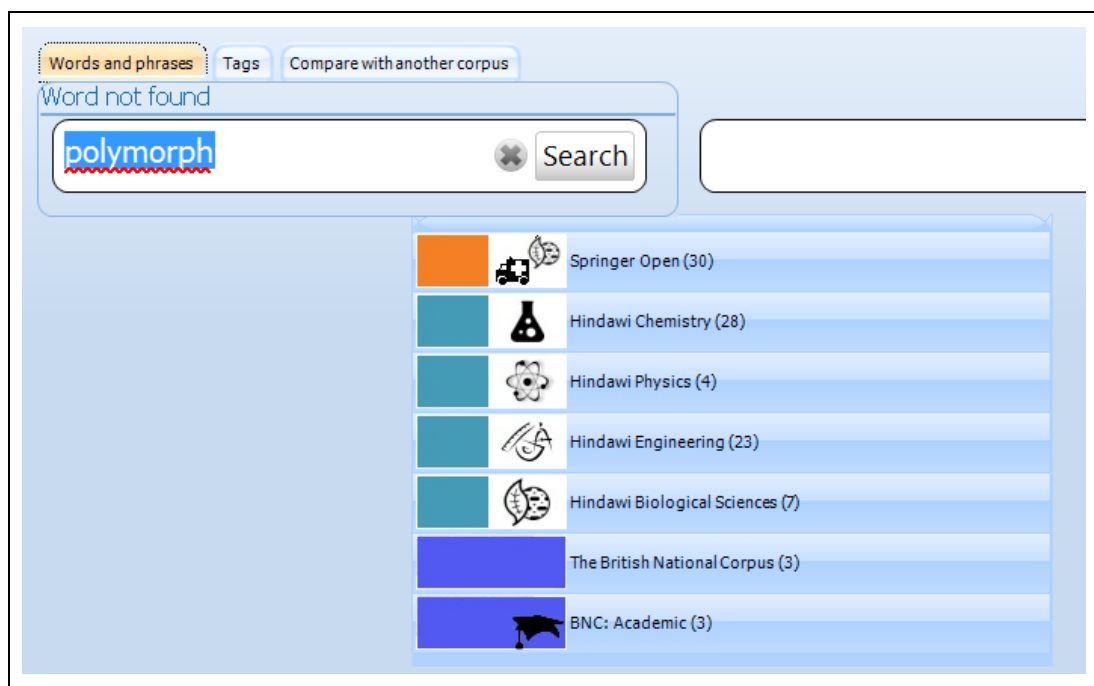


Figure 3.9: Screenshot showing the frequency of "polymorph" in alternative corpora.

Once the user has entered a word which exists in the currently selected corpus, alternative word forms and collocations appear on the screen. The detection and use of collocations is described in Chapter 4. The final section of this chapter introduces the way in which the other word forms and related words are used to prompt learners to perform useful comparisons.

3.6.2 Helping users compare words

When designing a concordancer for language learners, it is important to consider what the main reasons might be for them to perform searches. Chambers and O'Sullivan (2004) note that the software they used with learners would have benefited from having the capability of searching for lemma rather than specific word forms. However, as explained above, rather than giving the learner results with conflated word forms, it was decided it would be more appropriate to try to highlight these differences to the user by offering alternative word forms to display in a side-by-side comparison.

Looking through the literature on Data Driven Learning and studies which have evaluated corpus tools with language learners, there seems to be a consensus that comparisons of synonyms, as well as prompts to explore other word forms, would be particularly helpful. In one of the earliest papers on Data-Driven Learning, Johns (1991) explains that students often come to concordancers wanting to compare pairs of words. Corpora are thought to help demonstrate differences between synonyms clearly (Kaltenböck & Mehlmauer-

Larcher, 2005). All of the suggested activities given by Coniam (1997) for how corpora could be used in teaching require learners to compare. Three out of the six uses of corpora in the classroom given by Tsui (2004) involve different aspects of synonymy: near synonyms, words which are very close in meaning, and words which have the same translation in the learner's own language. However, student feedback from some studies has also shown that while it can be rewarding, learners find the discovery of differences between synonymous words both difficult and time-consuming (Yeh, Liou, & Li, 2007). There are several other obstacles which learners need to overcome. In order to see a pattern, learners may need to perform two or more searches (Gaskell & Cobb, 2004). Learners are not always ready to call to mind suitable words for comparisons. They may not be able to come up with further ideas on what to search for (Gabel, 2001). Sun (2003) notes that ineffective search skills also lead to frustration.

Given the importance placed by teachers and researchers on the power of comparisons in Data Driven Learning, it seems strange that little support is provided in most concordancing software to facilitate this. Both *AntConc* and *WordSmith Tools* require use of multiple windows or saved results in order to view two sets of concordance results or collocations simultaneously. While *The Sketch Engine* includes the *Sketch-Diff* function, only the summary Word Sketches are available in this view, and comparing actual concordance lines would require moving backwards and forwards between pages or having multiple tabs open in the browser. The *Sketch-Diff* query box also offers no suggestions or support and requires the same POS tag to be used for both nodes. Each of these tools can provide a rich variety of ways for a researcher to make comparisons between items but the pathway for making these comparisons can be complicated.

A further design feature for this project was to make comparisons between search terms as easy as possible. As well as prompting alternative word forms for comparison, the software also uses a junction box table to retrieve "related" words. Each row in the junction box table holds two ID numbers for lexical items, a language code which reveals the source of the link, and a ranking. In the current implementation, these links are based on the words being alternative English translations of Chinese words, retrieved from a freely available Chinese-English dictionary file, or on *WordNet* (Miller, 1995). To create the translation-based links, the *CC-CEDICT* database file (MDBG, 2013) was downloaded and imported into *Microsoft Excel*. The first column which contained the Chinese headwords was then deleted along with some function words such as "to" and "a" which regularly

appeared in the English translations. The columns were then imported into a *MySQL* database and a junction box table was generated so that strings (types) were linked together if they occurred in the same row in the original table²². In this way, a list of words which are alternative translations for Chinese headwords can be retrieved for any of the words listed as English translations in the dictionary. While it would have been possible to seek permission to incorporate a more advanced bilingual dictionary, it was thought that deriving the data from a simple, freely available glossary which had not been developed with a highly advanced lexicographical protocol would provide more simple suggestions and be more similar to the information provided in free mobile phone e-dictionaries. As explained in Chapter 2, the concordancer was developed specifically with Chinese learners of English in mind, but future versions could incorporate lists derived from dictionary mappings for a range of different languages or simple thesaurus data. For links based on *WordNet*, the script for extracting all semantically related words which is distributed with the *WordNet* database was adapted to generate short lists of related words. In order to extend these lists of links, the *SQL* script also links words where the stems of each pair match two new candidate words, and where the suffix of the two new candidate words also matches. A DICE mutual information score is then used to provide a ranking for the similar word pairs, so that they appear in the drop-down list with more mutually exclusive items towards the top, and words which occur with many other words towards the bottom. Collocations, alternative word forms and words with similar meaning are shown in Figure 3.10.

²² The database derived from *CC-CEDICT* is separate from the main corpus databases and could be distributed separately as a *SQL* resource, but for deployment purposes it would be important to check that this honours its Creative Commons Attribution-Share Alike 3.0 License.



Figure 3.10: Screenshot showing prompts which appear for *consequence* in the *BNC: Academic sub-corpus*.

3.6.3 Helping users compare results from two corpora

As well as looking at pairs of words, comparisons of a single item across different corpora can be a good way to show how use varies across different registers. Comparing the results of the analyses of two or more language samples is an important part of register analysis, since it is through comparison with other registers that the characteristics of one register become clear (Biber & Conrad, 2009). Just as most concordancing software does not provide an easy way to view and compare the results of two different items on the same screen, being able to view and compare results from two different corpora is also far from

straight-forward. *WMatrix* (Rayson, 2008) makes comparisons of two texts or two collections of texts very clear, by showing results of key words, key part-of-speech tags and key semantic domains for one collection using the other as a reference corpus. This is an excellent tool for researchers wanting to use differences in frequency between two corpora as a starting point for exploration of differences between the two collections of texts. If, however, a language learner wants to see how a word is used differently in two different corpora, none of the software packages really provide much support.

In *The Prime Machine*, a comparison between two corpora can be made easily using the “Compare with another corpus” sub-tab which appears on the main Search Tab. The search box on this screen looks and behaves as before, with auto-complete support at the word and collocation level. To the right of this box, a drop-down menu is provided which contains a list of all the other corpora which are available. When the user clicks on the “Compare” button, the application checks that the word or combination of words is present in both corpora at least once before the query is allowed to proceed. If the words do not appear in either of the two corpora which have been selected, feedback is provided. Figure 3.11 shows the search screen for comparing corpora. In order to allow access to the complete corpus as well as comparisons across its sub-corpora, texts from the *BNC* are stored in the database twice: once as part of the complete corpus and once in a sub-corpus determined according to the text categories (see Chapter 6 for details of the major text categories used for the *BNC*).

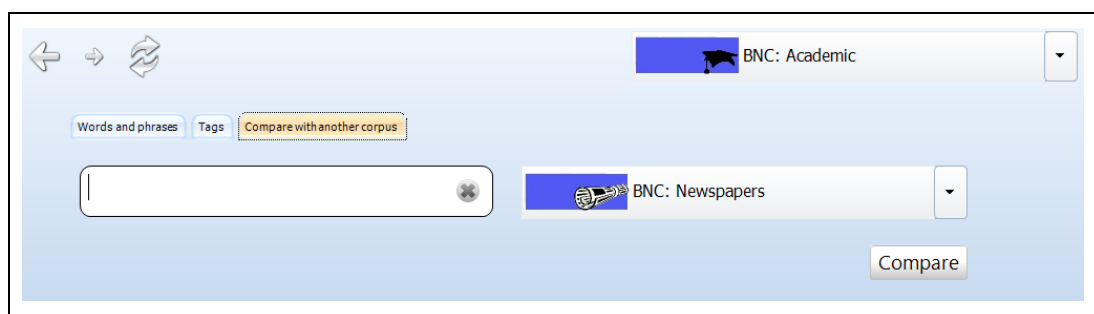


Figure 3.11: The “Compare with another corpus” sub-tab on the main Search Tab.

3.7 Summary

This chapter has outlined some of the innovations of the project with regard to the software architecture which balances hardware costs and corpus access requirements, the corpus refactoring processes and the spelling and query support in the concordancer. It has been suggested that the ability to present results for two searches side-by-side on screen, along with the other search query support, makes a highly productive foundation

for a concordance application aimed at language learners. The next few chapters explore collocations and concordance line ranking, the main features of Lexical Priming and the handling of key words, tags and metadata. After that, the results of an evaluation with language learners and teachers are provided in Chapter 7.

Chapter 4: Collocation

Perhaps the strongest argument for the use of corpus linguistics in the description of language has been the patterns of collocation which can emerge through analysis of texts. The biggest impact of corpus linguistics on dictionary design and also on language teaching has been through the prevalence of this accepted notion of collocation. From Firth's memorable statement "You shall know a word by the company it keeps!" (Firth, [1951]1957, p. 179) and his seeding of a new approach for lexicographers, the influence of collocation as a linguistic feature of interest has grown and modern textbooks for language learners and dictionaries contain collocation information for the lexical items they present.

4.1 Collocation and language teaching

Unlike some of the other features of language which have been uncovered through the approaches of corpus linguistics, collocation is a term with which language teachers are certainly expected to be familiar, and from the widespread use of the term in section headings and dictionary panels it is clear that students are being encouraged to gain an understanding of it too. The power of teaching collocation in language learning can be seen in some of the literature. As the pillar of the Lexical Approach, collocation is claimed to be a way to break through the "intermediate plateau" (Morgan Lewis, 2000, p. 14). Hill puts the lack of knowledge of collocations down as the root of many errors in learner language which are caused "because they create longer utterances because they do not know the collocations which express precisely what they want to say." (Hill, 2000, p. 49). In a book giving guidance to teachers on English for Academic Purposes (EAP), Alexander, Argent and Spencer make the following claim about the importance of collocation for a non-native speaking student's acceptance into the academic community:

In Academic English particularly, where writing is expected to conform to predictable patterns, mis-collocation can be one of the most distracting advertisements that the writer is not a competent writer in English, and can lead to a different meaning from that intended...

(Alexander, Argent, & Spencer, 2008, p. 163)

While not all language teachers will be following methods connected with the Lexical Approach or teach EAP, the ubiquity of the term across different aspects of mainstream

language teaching is obvious. For fifteen years or more, collocation activities have formed part of general English course books. *Cutting Edge* is a widely used course and it includes exercises matching verb-noun combinations and reflections on how best to note “words that go together” (Cunningham & Moor, 1999, p. 53). The *Touchstone* series of books is heavily promoted as being corpus-informed and understandably puts access to collocation information as one of the key aspects of the selection process for items and examples (McCarthy, 2004). From Level 3 in *Touchstone*, a main section is devoted to teaching students how to “Learn new words in combination with other words that often go with them” (McCarthy, McCarten, & Sandiford, 2006a, p. vii). From Level 4, the term “collocation” is introduced and subsequently used again in the skills summary of the scope and sequence pages (McCarthy, McCarten, & Sandiford, 2006b, p. ix). Moving to Business English, in the very popular course, *Market Leader*, students have exercises where they mark “word partnerships” (Cotton, Falvey, & Kent, 2006, p. 9), while in the teacher’s book they are called “collocations” and it is explained that one collocation for each word in the exercise can be found in the text (Mascull & Heitler, 2006, p. 12). In EAP, the use of the actual term “collocation” and teaching about its meaning seems to be more common. In an exam preparation book for the *International English Language Testing System* (IELTS) written by one of the key figures behind the exam itself, the need to “Choose words that go well together” is listed in the assessment criteria for both writing and speaking and reflective tasks based on this are provided (Jakeman & McDowell, 2008, pp. 92, 138). The public band descriptors for this test (Speaking, Writing Task 1 and Writing Task 2) all include “collocation” as a requirement for Band 7 “Good User” (available from www.ielts.org)²³. Other leading courses for Academic English incorporate information about collocations as well as activities encouraging students to notice and record them. The *Academic Encounters* series has reading activities where students are given an explanation of collocation, told how knowing collocations makes reading easier and instructed to scan the text to match nouns to verbs (Brown & Hood, 2002, p. 89). *EAP Now!* includes a definition of collocation explaining that they “just fit together” and emphasising there is no “special reason” for these combinations, encouraging students to ask their teacher if words form a “good collocation” and whether they “sound natural” (Cox & Hill, 2011, p. 31).

²³ While these public descriptors are located in the “Research” part of the website, direct hyperlinks are placed in the “Information for Candidates” pages, effectively embedding them in the student-oriented section too.

As well as being fairly well represented in teaching materials, learner dictionaries also seem to draw an increasing amount of attention to collocation information. As the pioneer of corpus-driven lexicography, the *COBUILD* dictionary has always emphasised collocation by design. The *Collins COBUILD Advanced Dictionary of English* includes collocations in its full sentence definitions and also has prominent “Word Partnership” boxes “giving the complete collocation with the headword in place to clearly demonstrate use” (*Collins COBUILD Advanced Dictionary of English*, 2009, p. viii). The *Macmillan English Dictionary for Advanced Learners* has “Word Partnership” boxes showing full collocations with word class information. Words which have “many collocations” have an additional “Collocation Box” grouping collocations by sense and word class combination (*Macmillan English Dictionary for Advanced Learners*, 2007, p. x). In the Second Edition, there were more than 500 entries with these boxes (Michael Rundell). A similar two tier system is used in the *Longman Dictionary of Contemporary English*, with collocations shown in bold type in the main block and an additional collocation box for those words which have “a lot of collocations” (*Longman Dictionary of Contemporary English*, 2009, p. xiii). Obviously, there is not sufficient space to include collocation information for most headwords and publishers also have dedicated collocation dictionaries. In the learner dictionaries listed above, although longer explanations of the meaning and importance of collocation are available, the short usage guides typically explain collocation in simple terms as being: “words that are often used with a particular word” (*Longman Dictionary of Contemporary English*, 2009, p. xiii); “how words combine” (*Macmillan English Dictionary for Advanced Learners*, 2007, p. x) or “... high-frequency word patterns” (*Collins COBUILD Advanced Dictionary of English*, 2009, p. viii).

4.2 Defining collocation

From a language teaching perspective, definitions of collocation can be fairly broad with the aim being to make learners more aware generally of the patterns of words around them. Developers of the Lexical Approach have introduced the concept to learners by describing the variation in the strength of relationships between words as being similar to the variations found in relationships between people (Hill, Lewis, & Lewis, 2000). The same authors have introduced collocation to students from a more practical perspective, comparing the ease of putting together a model aeroplane kit from “smaller pieces pre-assembled into recognisable chunks” (Hill, et al., 2000, p. 89) rather than from the smallest pieces separately. Other approaches include drawing on expectations of their mother tongue through reflection or translation of technical phrases (Conzett, 2000; Hill, et al.,

2000). With all of these approaches, the aim is to help learners start noticing collocation in the language they encounter and to encourage them to invest time in this enterprise because of the practical advantages collocations can offer. It seems that many learners find it unusual at first to examine language in front of them in units beyond individual words. In order to encourage students to think about language beyond individual words, Hill (2000) suggests asking students to underline verb + noun collocations, with the aim of making this kind of analysis become more natural with practice. Indeed, Conzett (2000) claims that explicitly making students aware of the term “collocation” will speed up class activities based around collocation. Woolard argues that definitions based on statistical information do not “guide my students’ attention to specific elements of text in a clear and directed way”; saying that for the purposes of teaching a definition focussing on expectation is more useful (Woolard, 2000, p. 29). He explains that he guides his students to “... those co-occurrences of words which I think my students will not expect to find together” (Woolard, 2000, p. 29) and also restricts the patterns covered by the term to specific combinations of word classes.

While the introduction of the term “collocation” in the classroom may be made through metaphor, reflection on translations in the mother tongue, practical applications or repeated methods of annotation, more formal definitions of collocation in linguistic theory and computational linguistics have developed over the years. The differences in the qualification and scope of collocations which mirror the different purposes that different teachers have in mind are part of a general tendency for different researchers to specify the meaning of collocation and similar phenomena in different ways. Linguistic descriptions include a wide variety of ways of limiting what should count and how it should be understood to operate, including strings of characters in raw text (Sinclair, 1991), lexical phrases (Nattinger & DeCarrico, 1992), lexical bundles (Biber, Johansson, Leech, Conrad, & Finegan, 1999), motivated or unmotivated collocations (Hunston, 2002), lexical networks across sections of a book (Phillips, 1985), within Pattern Grammar (Hunston & Francis, 2000), and as a major contribution to the identification of norms (Hanks, 2013). Each of these stipulate such aspects as whether collocation-like phenomena operate on word forms or lemma, only in specific grammatical relations or freely, and for any kind of word or only certain parts of speech. According to Hoey (2005), at times the very definition of collocation has been tied up with the methodological approach for their retrieval. To demonstrate this point, he quotes his own statistical definition of collocation from an earlier book:

“... the relationship a lexical item has with items that appear with greater than random probability in its (textual) context”

(Hoey, 1991, pp. 6-7 quoted in Hoey, 2005, p5)

He then introduces a definition which fuses the psychological importance of collocation with the means of detection and evaluation:

... So our definition of collocation is that it is a psychological association between words (rather than lemmas) up to four words apart and is evidenced by their occurrence together in corpora more often than is explicable in terms of random distribution.

(Hoey, 2005, p. 5)

Hoey introduces collocation using the example *inevitable + consequence* (2005, p. 2).

Further examples of collocation can be found in this chapter in Figures 4.3, 4.6, 4.7, 4.8, 4.9 and 4.10.

As a piece of software purposefully designed to support the examination of the kinds of relationship between words which are introduced in Hoey’s theory of Lexical Priming, collocations are defined in this project based on his 2005 definition²⁴. In this thesis, collocation will be used as it is in the software to refer to combinations of two, three, four or five words in a four-word window either side of a node. The term “multi-word unit” will occasionally be used where combinations beyond two words in length are the focus of the discussion. In *The Prime Machine*, collocations are referred to as “collocations” rather than “partnerships” or any other less technical term. This is partly because of its widespread use in textbooks and dictionaries, but also because as it has been argued in Chapter 2, Chinese learners in particular are likely to benefit from exploring language in a way which is new to them, with collocation playing an important role.

There are several tip screens dedicated to collocation and information about the Collocation Tab is also brought up by clicking on the life ring. Figure 4.1 below shows some of the information provided to learners in these ways as well as the pedagogical definition of collocation which was created as a result of trying to bring together Hoey’s revised definition (2005), simplifications inspired by the dictionary and teaching materials writers mentioned above and notions of the topic sensitivity of collocations.

²⁴ C.f. Hoey (2014) on collocation operating over greater spans.



Collocations show the connections between words which exist in the minds of people who use a language.

Dictionaries usually include some information about very common Collocations (also called “Word Partnerships”). Collocations for different forms of a word are often different.

The connections between words in the minds of experts from a specific academic field are often different from the connections between words in other fields.

Computers can help find collocations by going through millions of words from different sources and counting how often other words occur near a particular word, and measuring how likely it is that this would happen by chance.

Usually, only words up to 4 words before or 4 words after are counted.

You don’t always need a computer to show you collocations; as you read texts and listen to other people, try to notice which words often seem to go together.

Figure 4.1: Information provided for language learners about collocation on the Life Ring screen for the Collocations Tab.

Before going on to consider provisions for collocation extraction and display in other concordancers, and presenting details of how collocations are calculated and used in *The Prime Machine*, it is important to consider the role collocations can play as a resource in the classroom and for self-study activities for students of a foreign language. For all aspects of *Lexical Priming*, Hoey (2005) emphasises that associations will be specific to the domain and genre. Although it has been illustrated above that dictionaries and text books highlight collocation fairly prominently, when considering learners’ needs for collocations across different disciplines, resources are still very limited. Alexander, Argent and Spencer (2008) give a case study in which students ask whether “arrive at” and “come to” are suitable to be used with “conclusion” in academic writing and the teacher responds in the affirmative immediately. They discuss the fact that when giving students information about collocation it is best to check resources first and that a dictionary would probably not be much help. They recommend looking at concordance lines and presenting these as evidence to learners in the next class. Since collocations are often discipline specific, although textbooks and dictionaries do include some useful information, concordance lines and concordancing software are important resources for EAP. O’Keeffe, McCarthy and Carter (2007) also argue that corpus data can provide teachers with strong support for explaining the collocations for more “banal” or “everyday” words which are less easily retrieved through intuition, adding that providing learners with evidence from multiple

texts in a corpus gives a teacher much more confidence than just looking at one example in a class text.

Within the theory of *Lexical Priming*, Hoey also provides evidence for nesting effects, or the way in which combinations of words may be primed for use in ways which differ from those of the individual words. The term “nesting” is introduced by Hoey as a property of words “... where the product of a priming becomes itself primed in ways that do not apply to the individual words making up the combination” (Hoey, 2005, p. 8). Having 500 or more collocation boxes in a learner dictionary is good for those words, but concordancers are able to provide detailed lists for all items beyond a minimum frequency and also give insights into collocation in specific domains or genres. Concordancers can also provide examples containing specific combinations of words so learners can start to explore how these relate to specific uses and specific meanings: that is to explore evidence for how these combinations are primed differently.

4.3 What do concordancers offer in terms of collocation?

Some proponents of hands on learning activities with concordancers claim that producing lists of collocations is straightforward. When version 3 of *WordSmith Tools* was current and before *The Sketch Engine* or *AntConc* had been developed, Woolard asserted:

... concordancers like *WordSmith* ... are not complex and it only takes one short induction lesson to train students to use them for collocation exploration.

(Woolard, 2000, p. 42).

However, when introducing *AntConc*, Anthony (2004) suggested that some of the more basic information shown in concordancers including collocation tables can be confusing for learners. While developments in all these packages have made them more accessible and they can be used with learners as a collocation reference resource, they are best suited for rather different situations. Even if the learning session is set up so students are expected to act “like a researcher” (Johns, 1988, p. 14), the user of concordancing software needs to be aware of several principles and have made several research design decisions before clicking the button and getting the results. First, they need to decide on which corpus or which sub-corpora to use. Corpus selection is the first screen presented to the user in *The Sketch Engine*, and each of the tools in *WordSmith Tools* requires the user to first select the corpus text files to be used for the analysis. In *WordSmith Tools* and *The Sketch Engine*, collocations can be calculated after a set of concordance lines has been retrieved. The

pathway learners need to follow in order to view collocations is from (1) the selection of texts to (2) the formulation of a search query to (3) the display of concordance lines, and then from the concordance list to (4) the display of a list of collocations, with options to filter down the original list of concordance lines to see positive or negative evidence.

AntConc provides a slightly different route, with all the different types of concordancing analysis visible in tabs across the top of the program's window. Learners first have to (1) select the texts, then (2) generate word lists and then (3) create collocation lists. As with some of the other tabs, if other processes are a prerequisite for a calculation, *AntConc* tells the user what needs to be done and changes the view to the necessary tab. For example, a word list must be generated before collocations can be calculated, so if this has not been done, a message appears and the screen jumps to the corresponding tab. The default collocation settings in *AntConc* are for a minimum frequency of 1, just one word window either side of the node and for matches to be case sensitive. Teachers may find that undergraduate English majors focussing on corpus linguistics as part of their course may need assistance in using *AntConc* even with a worksheet of instructions²⁵. No doubt students from other disciplines who do not have the prerequisite linguistic and software knowledge needed in order to use concordancers to produce collocation lists will find these steps to be quite an obstacle.

The list of collocations for these packages, whether the bare-bones list of words and statistical measures, or the grammar-function detail of a word sketch in *The Sketch Engine*, all lead towards further questions which the researcher-user of the software needs to consider. It is easy to see why electronic or online dictionaries seem to offer learners a rather more straight-forward query process for the casual user of a system who is looking for answers on how to use a word or phrase. All three software packages cater well for advanced users, with many more options available for statistical measures and window size in *WordSmith Tools* and *The Sketch Engine* than currently implemented in *AntConc*. Someone aware of the limitations of each of the statistical measures may try multiple searches and interact with the results.

As explained in Chapter 3, *CenDiPede* is rather different from the other packages in that it is a system designed for the user first to create long detailed profiles for a search term and then to query these using its special query interface.

²⁵ Anecdotally, this has proved to be a fairly common in classes of Chinese students.

In terms of the software design of a new concordancer, the choice of statistical measures to make available is an important consideration. One of the earliest and still highly cited measures for collocation is the “association ratio” of Church and Hanks (1990) based on mutual information scores. Interestingly, they note that this was implemented so as to be asymmetrical, so different results would be produced according to the order of the two words being studied. It was proposed as a method to help dictionary writers as “an index to the concordances” (Kenneth Ward Church & Hanks, 1990, p. 29). However, by the time Oakes provided a summary of collocation measures in 1998, of the dozen measures listed only two were asymmetrical. Other than the “relative frequency” measure as provided in *The Sketch Engine*, the collocation measures which are available today in the main corpus packages give symmetrical results and are not sensitive to position. The issue of direction and symmetry of collocations is taken up further in Section 4.7 below. A list of statistics from Oakes, and *The Sketch Engine* is provided in Appendix 1.

WordSmith Tools, *The Sketch Engine*, *AntConc* and *CenDiPede* provide different coverage of statistics based on mutual information, but the summary in Table 4.1 clearly shows MI, T Score and Log-Likelihood to be the most common.

Table 4.1: Collocation measures available in concordancing software

Measure	<i>WordSmith Tools</i> (Scott, 2010a)	<i>The Sketch Engine</i> (Kilgarriff, et al., 2004)	<i>AntConc</i> (Anthony, 2004)	<i>CenDiPede</i> (Garretson, 2010)
MI	✓	✓	✓	✓
MI3	✓	✓		
T Score	✓	✓	✓	✓
Z score	✓			
Dice coefficient	✓			
Log DICE		✓		
Log-likelihood	✓	✓		✓
Relative freq.		✓		
MI-log-prod		✓		
Minimum sensitivity		✓		

In terms of the complexity of the presentation of the results, the packages also have some notable differences. With *WordSmith Tools* and *AntConc*, the name of the statistic is hidden on the results page, while with *The Sketch Engine*, clickable columns are provided for each statistic chosen. With *AntConc*, students may be confused about the meaning of the “stat” column. As well as the disassociation on screen between the statistical measure

and the general term “relation”, perhaps one element contributing to the complexity of *WordSmith Tools* is that it shows counts for different positions (L4, L3, L2, L1, etc.). This information is, of course, very useful to someone familiar with the language and able to do the mental gymnastics of creating phrases in their mind to fit each pattern.

Word Sketches in *The Sketch Engine* make the grammatical relationship between collocates and the node word accessible, but take demands on mental processing to a higher level. With practice, no doubt advanced *users* can conjure up the appropriate phraseology for “object_of”, “subject_of”, “modifier” and “modifies”, but it is hard to imagine how high intermediate or advanced *language learners* could. These Sketches are very powerful, but in order to see how these are used in order and position, the user would need to click on the frequencies of each to reveal concordance lines.

It is possible to use the query language in *CenDiPede* to retrieve asymmetric values (Garretson, 2010, p. 78), but the default ranking procedure of collocations in *CeDiPede* is based on a combined weighting from the three measures.

It can be seen that packages offer different statistics but they are almost all symmetrical and do not take into account ordering or positioning. Through working repeatedly through the collocation settings and generating multiple pages of results, it is possible through these packages to obtain some word order information. For example, the user could change the default window size or limit searches to left only or right only by setting one of the window size values to 0. Unlike the learner dictionaries which promote their collocation panels as clearly showing the word partnerships in the order in which they appear, results from the packages introduced above show lists of collocates isolated from the node.

In summary, while some techniques require tagging of grammatical relations, those using raw data usually give symmetrical results for word pairs and ignore their positions. However, heavy tagging processes are not always possible or desirable, and the starting point for a language learner exploring collocations is probably a specific node, and often a specific node in a specific position in relation to the other words they want to use.

4.4 Multi-word units of more than 2 words

Having considered the importance of collocation from a teaching perspective, its representation in textbooks and dictionaries, definitions and means of calculating two word collocations, this section will consider multi-word units of more than 2 words. The frequency of linguistic items is often used as a basis for judgements about their importance in teaching. In terms of evaluating the importance of a multi-word unit, it has been suggested that the frequency of the multi-word unit can be measured against individual word frequency (Sorell & Shin, 2007). This, of course, means that very few multi-word units obtain a level of frequency needed in order to displace medium frequency vocabulary items from the curriculum. However, extending collocations beyond two words is considered to be a valuable goal from a language teaching perspective. Shin and Nation (2008) created a list of collocations of two or more items using the spoken section of the *British National Corpus*. They found that in spoken data many multi-word units would reach sufficient frequency to be included in elementary curricula based on raw frequency, but found in the results many culture-specific and colloquial items. They argue that the criteria for including units of two words and more in a list for teaching should be very strict and not solely based on raw frequency but consider “learner need, range of use (for example in both spoken and written use), difficulty, teachability, and suitability for the age and background of the learners” (Shin & Nation, 2008, p. 346). Following the generation of the Academic Word List (AWL) (Coxhead, 2000), there have been attempts to find ways to generate lists of academic phrases. Just as with the AWL, one way of reducing the number of multi-word units to include in the teaching of academic English would be to filter out multi-word units which are frequent outside academic texts. Simpson-Vlach and Ellis (2010) produced a list of multi-word units consisting of three, four or five words which were used in academic texts. They developed an approach using log-likelihood in a key word manner to determine which multi-word units occurred to a statistically significant level in academic texts more than non-academic texts and then a combined metric based on both mutual information and frequency in order to rank the results. They used teachers to judge the usefulness of phrases and then used a weighted version of MI and frequency to create what they call the FTW (formula teaching worth). The last part of this project was to group phrases according to common use. In terms of achieving the goal of creating a list of multi-word units which are relevant across academic disciplines, their results are interesting and could provide possible lexical clues for a range of specific functions in academic speech and writing. However, just like the AWL, the list excludes multi-word

units specific to a discipline, so while it has importance in terms of application to EAP courses catering to students in mixed disciplines, it does not bring to the forefront the expert-like constructions which would meet the needs of language learners wanting to grow in their linguistic expertise in their own academic field. Based on his own exploration of ways to create a list of academic collocations for EAP teaching across academic disciplines, Durrant (2009) questions at what point collocations of increasing length may cease to be useful across disciplines, since they are likely to become more and more subject-specific.

As was seen in Section 4.3, collocations of two words are available in major corpus software packages, but for multi-word units longer than two words, the options are much more limited. One approach to multi-word units in corpora and language teaching is through producing “Concgrams” which Greaves and Warren (2007) claim help show learners the “aboutness” at either text or corpus level. *WordSmith Tools* offers Clusters as another way of viewing the data, and while the large number of papers published looking at n-grams or clusters shows that this can be a rich field of research, ordered multi-word units based on statistical measures are not widely available in concordancing software. While Concgrams, Clusters or n-grams can be calculated and displayed, there are difficulties in finding appropriate statistical measures to filter or rank these and to merge the results with bigrams (two word collocations). Danielsson suggests there is a “mismatch between what computer programs can easily do and what mwus are like” (Danielsson, 2007, p. 18). From a methodological perspective, it appears difficult to outperform raw frequency when extending current collocation measures to extract units of more than two words (Kilgarriff, Rychlý, Kovár, & Baisa, 2012; Wermter & Hahn, 2006). Within the field of information retrieval, when multi-word units are entered in a search engine, pages containing these in order will be weighted more heavily than if they are unordered (Croft, et al., 2010, pp. 275-277). Extraction of collocations for linguistic analysis is different from the methods used to obtain web sites which have good coverage of a topic requested by a search engine user, but the importance of the ordering of the words for informational retrieval does suggest that the ordering of the elements in a multi-word unit could be important.

Appendix 2 shows a table containing some of the statistical formulae which have been used and evaluated for multi-word units. The method adopted by Shimohata, Sugio and Nagata (1999) focussed on measuring the range of different environments in which a multi-word

unit occurred, arguing that finding the extremes of a unit could be achieved by detecting many different items around it; that is to say there would be a wide variety of lexical items either side of the multi-word unit. The other measures shown in Appendix 2 calculate multi-word collocations by extending bi-gram measurements in a number of different ways. Petrović, Šnajder and Bašić (2010) explain that the measures differ in which elements of the n-gram are compared, as well as the configuration of stop-lists.

In *The Sketch Engine*, multi-word units beyond two words have a very limited role. When evaluating measurements for collocation, Kilgarriff et al. relate an anecdote which demonstrates how two word collocations which are elements in longer phrases may be difficult to judge depending on the familiarity of the respondents with different topic areas. They explain how the relevance of the collocation of “world” and “finals” was questioned by one judge, but another judge was able to quickly add “cup” to make a very common and recognisable use of these words in a set phrase. In order to find a pragmatic balance between over stating the importance of multi-word units of more than two words on the one hand and missing out on very common phrases on the other, *The Sketch Engine* (Kilgarriff, et al.) displays the commonest match for each collocate to show one longer phrase with specific types (not lemma) in their lemma-based word sketches. Multi-word Sketches are also possible in *The Sketch Engine*, but they note that “Word sketches are usually only interesting if based on several hundred data instances” (Kilgarriff, et al., 2012, p. 3), so lower frequency multi-word units and users of corpora of millions rather than billions of words would find this quite limiting. The pathway for the user to explore multi-word items in Sketches is explained using the example of “take advantage”, with the user first needing to select “take” and then selecting “advantage” from the list of collocates. The other way in which multi-word units are presented in *The Sketch Engine* is through multi-level tokenization which is a system to mark names like “New York” as a single collocate rather than two separate items.

As a state-of-the-art corpus tool catering for lexicographers and researchers, as well as gaining some popularity in language teaching, *The Sketch Engine* does not incorporate the extensions to two word collocation statistical measures, basing the “commonest match” on raw frequency. Kilgarriff et al. (2012) cite the evaluation by Wermter and Hahn (2006) where it is argued that compared to the other statistical extensions, raw frequency is a good measure, and even more so if grammatical information can be incorporated. Wermter and Hahn (2006) took collocation and automatic term recognition lists and then

compared the results they obtained based on raw frequency compared to two other measures.

In a language learning setting, whether in dictionaries, text books or computer aided language learning, the problem with measures for multi-word items seems to be more to do with being selective and suitably stringent, rather than not being able to measure anything. The approach for *The Sketch Engine* is very pragmatic and the chances are it will work well for many situations, but the context in which I expect *The Prime Machine* to be used is rather different. A lexicographer using *The Sketch Engine* to look at collocations will be able to draw on other resources (corpus-based, corpus-informed, as well as their highly tuned intuition) to be thrilled or irritated that a long phrase qualifies or does not qualify for inclusion in the results. The main task for them is also to produce sub-entries for specific headwords, since dictionaries usually focus on multi-word units as a means of supporting or explaining a particular sense of a word. But the situation for language learners is very different. Firstly, if two word collocations are actually just pieces of a much longer phrase, it is highly unlikely that a language learner would be able to guess what the full form would be. If language learners are presented with a list of single collocates and in actual fact two or more of them form pieces of a highly frequent phrase, it would also be very difficult for them to understand how to put them back together.

4.5 Priorities for Collocation in *The Prime Machine*

A number of aspirations related to collocation and multi-word units shaped the design of *The Prime Machine*, and these can be summarized as follows:

- to provide language learners with a means of exploring collocations and multi-word units in a range of different corpora by themselves;
- to help the learner not only see relationships clearly, but also to find and select useful starting points and to avoid unfruitful starting points;
- rather than giving learners lists or clouds merely containing isolated collocates, to present full collocations in typical word-order with the aim of improving understanding and retention;
- to make the concordance lines central to the user's experience with the software, being a way to help explain the collocation list, and also for strong collocations to offer one way of sorting concordance lines;
- since learners are unlikely to read exhaustive lists, to implement a collocation measure which could offer multi-word units of more than 2 words where this is

could be helpful, but also to show common words around collocations and how these form part of larger units;

- to address the tension between a computer-science oriented approach where multi-word units are felt to perform better with stop lists against the fact that function words are often part of recognizable chunks and that language learners need to see how function words operate as important elements in longer structures;
- to build into *The Prime Machine* the facility to demonstrate to learners the differences between the primings of words and the nesting of those words.

There were also a number of practical considerations. Given the treatment of punctuation as separate types in the database, consideration was needed regarding how to piece back together hyphenated multi-word units such as “knock-on effects”, which depending on the tokenisation method adopted could theoretically count as two, three or four tokens. Another practical issue was related to the amount of storage space needed in the database for collocations and other summary data about the collocations. In order to provide an acceptable response time for language learners using the system, summary information about the typical environments of words and multi-word units would be pre-calculated and stored in the database. However, if too many multi-word units were included this would start to affect the speed of both the generation and retrieval of data in the summary tables. The collocation measure needed to provide a suitable cut-off point for determining which multi-word units would be included.

Section 4.7 of this chapter presents a newly developed approach using log-likelihood contingency tables, designed to capture directional strength and be extendible to longer units. Rather than using joint and separate frequencies as proposed by Dunning (1993), this new measure compares the proportion of slots surrounding the node which are accounted for by a collocate in a specific set of positions. The results are displayed in clouds or tables showing full forms and indicating open slots. An example of this kind of word cloud is provided in Section 4.8.11.

The first part of this chapter has introduced several key influences and objectives for the handling of collocation in *The Prime Machine*. Ideas from language teaching, dictionary and textbook resources, other concordancing software and linguistic theories have been presented. The focus of the chapter will now move to specific design features and functionality related to collocation which has been put into the software. First, the way in

which collocations are calculated and stored in the database will be explained. Then, the display of collocations and the contribution they make to several other processes will be explained; specifically, their use in text prediction, in attention focussing captions, in the calculation and display of semantic associations and in concordance line ranking and selection.

4.6 Mutual Information measures used in *The Prime Machine*

Although the aim was to create a concordancer for language learning and teaching which would be simple to use and demonstrate clear differences between words, it has to be recognised that many users, and teachers in particular, may already be familiar with other software packages and the collocation measures which they offer. Before introducing the new collocation measure, a brief summary will be given of how mainstream collocation measures based on mutual information are implemented.

Procedures for extracting these collocations are well established. Shin and Nation (2008) hold that collocation should not cross established boundaries within a text, and support others in the argument to treat different word forms separately to allow patterns of collocation for each type of the same word to become apparent. While many queries on *The Sketch Engine* will produce conflated results for inflected forms and *CenDiPede* only stored collocates in specific grammatical relations to the node, as has been introduced in the previous chapter, *The Prime Machine* calculates results based on types rather than lemma. Keeping word forms separate and constraining the measurement of collocation within sentence boundaries would seem to be a good starting point for a tool designed for language learners tackling academic writing.

As was seen in Table 4.1, the common mutual information (MI) measures include T-Score, DICE and the cubic association score (MI3). It would have been very straight-forward to add the log-likelihood statistic to these three, but since this statistic was the basis for the new collocation measure, it seemed better not to offer two metrics with the same name but based on different calculations. However, for the new measure for log-likelihood collocations (explained in Section 4.7 below), it is necessary to use the frequency of joint occurrence, and since this is the same value which is used for all three MI collocation measures, in the refactoring and optimization process collocations are extracted for all lexical items in the whole corpus before the other procedure to calculate collocations using the new method is run.

A simple function was created in *SQL* for each of the three MI collocation measures and a stored procedure was written so that when a corpus has been refactored and summary statistics are processed, the database will run through each lexical item extracting and grouping words contained within the same sentence in a four word window either side of the node. The top line of the MI3 formula can cause an out of range error in *MySQL* since cubing a joint frequency of over one million with a very large total corpus size can exceed the permitted values for the *BIGINT* variable type. Therefore, the order of operations was altered in the *SQL* function to avoid this. As totals for the joint and separate frequencies are counted, all three values are stored along with the frequency for all items with a joint frequency greater than 2.

Following *WordSmith Tools*, it was felt that displaying information about the relative importance of each position of the collocate could be useful. In this system, since collocations are stored in summary tables in the database and are not calculated on the fly, to minimize storage space percentages rounded to the nearest whole number are stored rather than raw frequencies. Thus the database table for these collocations has columns containing *TINYINT* variables for L4, L3, L2, L1, R1, R2, R3 and R4. This does, of course, lead to some rounding errors so not all the positions will add up to exactly 100, but it does give very clear information about the relative importance of each slot.

The MI Collocations for each corpus are stored in a single table in the database. The structure is shown in Figure 4.2 below:

Column Name	Data Type
cb_node1	MEDIUMINT(9)
cb_node2	MEDIUMINT(9)
cb_freq	INT(11)
cb_node1l4	TINYINT(4)
cb_node1l3	TINYINT(4)
cb_node1l2	TINYINT(4)
cb_node1l1	TINYINT(4)
cb_node1r1	TINYINT(4)
cb_node1r2	TINYINT(4)
cb_node1r3	TINYINT(4)
cb_node1r4	TINYINT(4)
tscore	FLOAT
mi3score	FLOAT
dicescore	FLOAT

Figure 4.2: Table structure for MI Collocations

As can be seen, the two MEDIUMINT columns link to the lexicon table in the database and operate as a primary key. An additional index is added at the end of the process to facilitate fast access of rows based on “cb_node2” followed by “cb_node1”. Since the three statistical measures are symmetrical, there is no need to store all the information in the database twice, so when going through the lexicon only collocates which have a primary key lower than the node are stored (i.e. cb_node1 must be greater than cb_node2). When extracting the collocations, the results of two queries using each order are merged and the values for each slot are given aliases to reverse the order as required, so in the client application L4 is always 4 words to the left.

For the user, collocations based on MI statistics can be displayed on the Collocations Tab. The default is to download the top 20 collocates for each measure and then display these in a merged list, ordered by the currently selected statistic. It was decided that giving users a combination of word clouds and tables would be a good way to draw attention to strong collocations in the clouds, while providing additional data and a longer list in the table. The collocate clouds are based the visualisation technique of Tag clouds, which are common in other computer applications and considered to be one way to make relationships between words and tags more attractive (Croft, et al., 2010, p. 407). However, within concordancing software, this kind of visualization has not been widely adopted although it is used for some specific purposes in some programs. *WMatrix* (Rayson, 2008), for

example, presents alphabetically ordered clouds for key words and semantic categories. Since Version 6, clouds have been introduced into *WordSmith Tools* and the visuals for these in the online manual suggest that they fill rectangular boxes. In *The Sketch Engine*, clouds are available for the “thesaurus”²⁶ and like *WMatrix* and *WordSmith Tools*, the clouds fill the screen with many items. *AntConc* does not currently include Word Clouds and they are not currently on the Development Roadmap for *AntConc*²⁷.

For a concordancer designed for language learners, it would seem desirable to use clouds to draw attention to some of the strongest collocates, rather than to fill a cloud with an exhaustive list. One of the free word cloud generators that produces attractive clouds and is available on the internet is *Wordle* (Feinberg, 2013). In response to a programmer’s forum posting, Feinberg (2009) provided some information about how *Wordle* works. Rather than plotting the items using an algorithm to fill all available space in the panel containing the cloud, a method based on the *Wordle* spiral effect was adopted and procedures created in *Delphi*. Essentially, the most significant item is centred in the cloud and other items are plotted in a spiral leading out from the centre. When drawing word clouds using computer programming languages, one important consideration is how to determine the optimal font size to use for each element in order to suggest a difference in importance while ensuring that a sufficient number of elements are of a size which is legible. For all the collocation clouds in *The Prime Machine*, the size of the font used for each item is based on the statistic rather than the raw frequency. In order to reduce the magnitude of differences between the statistics when they are displayed in the cloud, square-roots are used²⁸ and a multiplier for each cloud is calculated by placing the highest ranked item first, and then determining the scaling needed to transform the square-root of

²⁶ The “thesaurus” in *The Sketch Engine* is based on a measure of similarity between the word sketches of potential pairs of words (Lexical_Computing_Ltd., 2014), and this approach is different from traditional thesauri (Kilgarriff, 2003). Such an approach which is essentially based on shared collocations in particular grammatical relations could be questioned given the evidence that Lexical Priming provides for differences in priming between pairs of synonymous words.

²⁷ http://www.antlab.sci.waseda.ac.jp/antconc_index.html accessed 27 December 2013

²⁸ Unlike the log-likelihood measure, the square-root is linear and so performing this operation is simply a pragmatic approach to mapping a wide range of values to the range of sizes of text which are legible to the human eye. However, other approaches are also possible and, for example, a metric based on the square-root of the frequency was used for text size in *the Word Tree* (Wattenberg & Viégas, 2008).

its measure of collocation strength into the desired font size. Further work could be done to make the clouds more attractive and further aspects of the processes used in *Wordle* could be implemented, but for the purposes of providing a quick view of collocates, they seem to be fit for purpose. Figure 4.3 shows how these appear in the application.

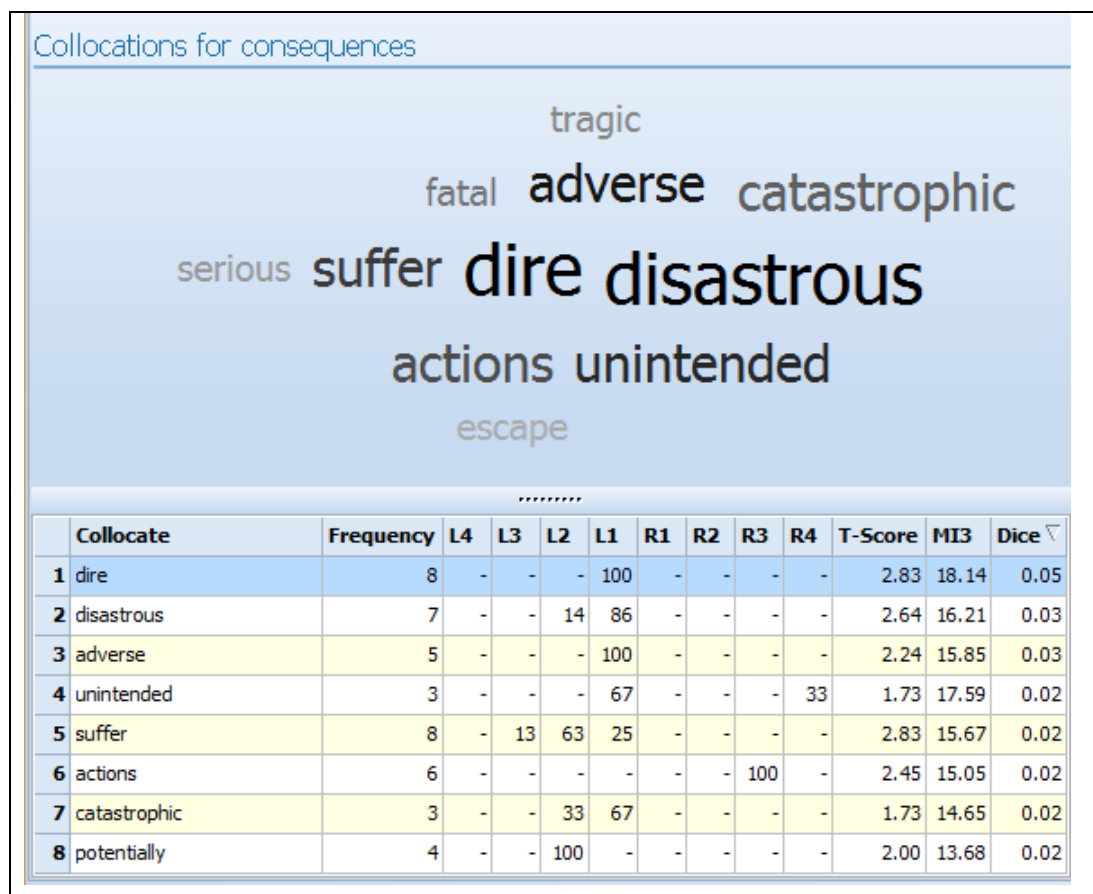


Figure 4.3: MI Collocation cloud and MI Collocation table for the node *consequences* in the *BNC: Newspapers* sub-corpus, sorted by the Dice score.

One of the trade-offs with database architecture such as that used in *The Prime Machine* is that the more data which are returned, the longer the server will take to access and transmit the results. During the development, I was mindful of previous criticism of collocation lists such as that of Lewis (2000b) who reported that only being able to see the top 20 collocates with the CoBUILD Collocation Dictionary CD-rom was insufficient. A plus button designed to look like buttons used in applications on mobile devices to request more results allows the user to obtain the top 100 collocates for each measure and to view these in the table.

With suitable indexes set up, retrieving collocates for an item is extremely fast. Since all the data are downloaded at the same time as the concordance lines and other information,

switching between collocation measures is also very fast. MI Collocations can also be used in the software as a concordance ranking measure, but this is discussed in more detail at the end of this chapter.

4.7 A new approach for collocation

It has been explained above that there are already a wide variety of collocation methods, so it may seem rather bold and perhaps unnecessary to introduce a new one for the purpose of this project. However, a new approach has been developed for several reasons. Firstly, it was felt that directional collocations were preferable. Since it was planned to provide the user with a short list of collocations as part of the auto-complete functionality described later in this chapter, it was felt that the best approach would be to develop a statistical measure which could match the user's perspective; the node word entered in the search box should conjure up a list of phrases oriented towards the specific node, not a mutual measure. In order to display collocations in the order and sequence in which they are typically found, calculating collocation strength between two items could have been done using a mutual information statistic producing symmetrical results, and then the commonest position could have been chosen for display, but with the method proposed here collocates which occur significantly in more than one position will be displayed in both forms in the results. Secondly, initial exploration with this method seems to show that the multi-word collocations which are generated are more intuitively suitable than those generated based on raw frequency. That is to say, the same method can be used to measure the strength of relationship for collocation groups of more than two words. Thirdly, as will be evident in the following two chapters, using log-likelihood in this way provides a unity for all the main statistical measures used in the package, so by grasping one statistical measure, an advanced user who is interested in the statistics behind the results will be able to understand all of them. Understanding of the log-likelihood contingency tables is likely to be of interest to and within the grasp of some of the university level language learners who use the software, particularly those studying more technical or mathematical programmes.

Early work on collocation considered the possible importance of the proximity and sequencing of the elements of a collocation. Writing well before the mutual information statistics used in modern concordancers had been applied to language, Sinclair rejected the idea of attaching greater importance to collocates nearest the node and stated:

But as the theory stands at present, the primary structural criterion is that of co-occurrence, in any sequence, with or without intervening material; features such as preferred sequences, or habitual interventions, are secondary in structure.

(Sinclair, 1966, p. 414).

He goes on to argue that “The same collocation has a different significance to the items involved” (Sinclair, 1966, p. 428). This difference is picked up in his later work where he describes it as a subject of “enduring interest” (Sinclair, 1991, p. 115). Nevertheless, as Gries (2013) points out in a recent paper, asymmetrical collocation measures have not had much of an impact on corpus linguistics so far. His proposal to use *Delta P* was published after my own work had also independently led me to the development of a means of bringing together measures of directional association strength and multi-word units beyond two words. In future, it would be interesting to explore whether *Delta P* could be used as an alternative or complementary measure, but the software described in this thesis was already at too advanced a stage for detailed attention to this measure to be given.

The approach adopted in this project also ensures that results are stored in summary tables looking at each collocation pair in both directions. The sequence of the two items within the window does not affect the ranking, but the co-frequency is calculated for specific slots rather than the total co-frequency in the window. In response to statistical objections about double counting of items, Sinclair (1991) argued that each token was considered once as node and at other times as collocates, but never both at the same time. With the approach presented here, the contingency tables make it obvious that the whole corpus is accounted for once for each collocation pair under investigation.

The log-likelihood measure has been applied to collocation before and is one of the common statistics available in modern concordancers. Dunning (1993) proposed the use of Log-Likelihood, as a way of balancing the bias towards low frequency items which exists in many of the other measures. It is a measure which is also very popular in key word analysis, and has received attention over the years due to questions about appropriate cut-off points for minimum statistical significance and the risks of over reporting relationships when applied to very large corpora. Rayson, Berridge and Francis (2004) demonstrate why in corpus applications the log-likelihood scores should be considered significant when greater than 15.13. Wilson (2013) recommends the use of Bayes Factors in keyness and other calculations in corpus linguistics to distinguish between very strong evidence and less strong evidence based on the overall size of the corpus, and Table 4.2 shows how these are

calculated and to be interpreted. This approach is used in order to standardize the cut-off point for collocations, the other priming features which are introduced in Chapter 5 and key tags which are introduced in Chapter 6.

Table 4.2: Approximate Bayes Factors and Equation for BIC approximation

Approximate Bayes Factor (BIC)	Degree of evidence against H0 ²⁹
0-2	not worth more than a bare mention
2-6	positive evidence against H0
6-10	strong evidence against H0
>10	very strong evidence against H0

$BIC \approx LL - \ln(N)$
 Formula from Raftery (1986) and Kass and Raftery (1995) given in Wilson (2013).

Using the BIC approximation it is possible to calculate the minimum size of a corpus required in order to satisfy the 15.13 level proposed by Rayson, Berridge and Francis (2004) above, while still reaching a BIC of 2. The natural log of 500,000 is 13.12, so subtracting this from a log-likelihood of 15.13 or more will generate a BIC value of at least 2. Thus, following Wilson’s proposal means that a BIC cut-off point of 2 on a corpus of half a million tokens or more will provide a level of stringency equivalent or greater than the 15.13 level and provides scalability to make comparisons between corpora of larger sizes possible.

When Dunning (1993) proposed applying log-likelihood as a collocation measure, the values used in the formula (F4.12 in Appendix 1) were the combined frequency, the separate frequencies and the total corpus size. However, log-likelihood can also be applied to contingency tables, where frequencies in one corpus can be compared with frequencies in another corpus, and the usual configuration and formula used to measure keyness is shown in Table 4.3 below. With this well-known technique, “Corpus One” and “Corpus Two” could be a study corpus and reference corpus, or two sub-corpora formed by splitting a larger corpus.

²⁹ H0 stands for the null hypothesis. In the context of collocation, the hypothesis would be that the items occur together more frequently than would be expected by chance. The null hypothesis is the opposite of this: that the items do not occur more frequently together than would be expected by chance. In other words, H0 is the hypothesis that the words do not form a collocation.

Table 4.3: Contingency table and formula for key words

	Corpus One	Corpus Two	Total
Freq. of word	a	b	a+b
Freq. of other words	c-a	d-b	c+d-a-b
TOTAL	c	d	c+d

$E1 = c*(a+b)/(c+d)$
 $E2 = d*(a+b)/(c+d)$
 $LL = 2*((a*log(a/E1))+(b*log(b/E2)))$

Table and formulae from Rayson and Garside (2000); citing approach from Read and Cressie (1988).

If the key word way of looking at a corpus and forming a contingency table is used to extract collocations, it means that a sub-corpus is created by taking the words surrounding all the instances of a specific node word. The remainder of the corpus is used as a reference corpus with which to compare the relative frequencies of each collocate. The number of words either side of the specific node word could be very limited or extend to complete sentences and even whole texts. Before describing this contingency table in detail, it is necessary to first consider the size of window used.

During the development of my system, I tried working with three levels: the traditional 4 word window (adapted slightly for sentence boundaries), complete sentences and whole texts. Issues arising from the application of this method to whole text environments will be discussed in more detail in Chapter 6. Beyond initial testing, sentence-level collocations were not taken further for three main reasons. Firstly, having such variable distances between the items meant that any visualisation of these relationships would be very complicated. With collocates in close proximity, an ellipsis might represent a gap of a few words, but it was hard to imagine how a long gap of tens of intervening words might be clearly shown in a way to help language learners see how they differed from or were related to nearer combinations. Secondly, the limit of 4 words either side of the node is well established and also supported in Hoey's coverage of collocation in his theory of Lexical Priming (2005) and the definition provided earlier in this chapter. Thirdly, wider windows meant longer processing time, and examination of the preliminary results while the *SQL* scripts were being developed did not seem to merit the investment.

Another issue which Sinclair (1991) also raised and is an important question for windows is whether punctuation should be considered and whether sentence boundaries should be imposed when collocation is investigated. With four-word windows where sentence boundaries are not taken into account, the result may be more *raw* or *untouched*, but

sentence breaks mark a clear grouping and ordering of words. Collocation measures usually work with the frequencies rather than the number of slots available in the collocation windows. If windows are restricted by sentence boundaries, however, the span will be uneven for many instances where the node is located less than 4 words from the beginning or end of the sentence. That is to say, by limiting the measurement to words within the same sentence, it is obvious that some of the windows will not be a full 8 words in length. If the node word occurs at the beginning of a sentence, for example, the window for that occurrence will just be 4 words long. Similarly, if a word occurs at the end of a sentence, or if the word is in a very short sentence, the windows will also be smaller. In his thesis, Collier (1999) mentions this problem and explains it as one of the reasons for choosing +/- 4 for his concordance line selection. He says that since some of the lines will be -4 but only +3, it will lead to statistical problems. Keeping the window smaller than 5 avoids an ever increasing number of these problems, but also lessens the measurement of the position of words in the sentence. If other tendencies of words related to features of Lexical Priming were not also being measured in *The Prime Machine*, this could lead to some loss of potentially interesting information, but the tendency for particular words to appear in Theme or Rheme and any tendency to appear towards the beginning or end of sentences are captured in the measures which are outlined in Chapter 5.

With a relational database, counting the total number of words actually taking positions in available slots is trivial, although it is time-consuming when repeated for each *type* as a node in a large corpus. The method used takes the actual sum of the available window slots rather than simply multiplying the node frequency by 8. What this means is that the contingency table can be seen as measuring the relative difference in frequency between a potential collocate in a specific position (or range of positions), balanced against all the other combinations of words in those windows, and its frequency outside the windows. In this way, the frequency of each potential collocate for two word collocations is divided between:

L4/L3/L2 vs. L1 v.s R1 vs. R2/R3/R4

This provides the learner with up to 4 significant collocations for any node and collocate pair:

“X .. Y” X Y Y X Y .. X

The measurement is effectively asking “Is the proportion of windows where collocate Y occurs in this position significant statistically compared to its occurrence outside the windows”. The contingency table for “on the left with a gap” is shown in Table 4.4 below.

Table 4.4: Contingency table for Log-likelihood Collocations for a specific set of slots

	Corpus One	Corpus Two
Freq. of word	<i>A = In slot L4, L3 or L2</i>	<i>B = Outside the +/- 4 word window</i>
TOTAL	<i>C = Count of all slots in +/- 4 word windows</i>	<i>D = Whole corpus – C</i>

Through a stored procedure in *SQL*, the value of C is calculated for each node by counting the actual number of words in the corpus within a window of up to plus or minus 4 words in the same sentence, and the value of B is calculated through subtracting from the frequency in the whole corpus the joint frequency in all slots which was stored as part of the calculation of MI collocations earlier. Collocations are stored in a summary table if they meet the minimum threshold of a BIC score of 2.

In order to reduce the collocation processing time required for extremely high frequency items, rather than using a fixed stop-list, collocates are not calculated for lexical items which have a frequency greater than 0.25% of the total corpus word count. This default setting was established through experimenting with a small range of corpora and sub-corpora of sizes ranging from 1 million to 150 million tokens, but this can be adjusted before a corpus is processed. However, it is very important to note that although these items above this threshold are not stored as the node of a collocation, they are not used as a stop-list, and they are considered as potential candidates as collocates for other items. As shown in Table 4.5 below, for the *BNC* (BNC, 2007), this means 49 items will not have any collocations stored in the system with the item as a node, and this list includes 8 punctuation symbols.

Table 4.5: Cut-off points for storage of collocations for high frequency items as node in the complete *BNC* based on frequency

Excluded Words	Frequency	Included Words	Frequency
1 the	6,046,727	50 all	281,629
2 ,	5,017,057	51 —	272,494
3 .	4,715,167	52 do	271,170
4 of	3,046,937	53 been	259,891
5 and	2,623,420	54 :	257,173
6 to	2,600,676	55 has	256,644
7 a	2,170,498	56 their	254,379
8 in	1,947,880	57 if	253,396
9 that	1,119,624	58 will	251,261
10 it	1,055,324	59 would	246,218
11 is	991,057	60 so	242,449
12 was	881,703	61 what	240,539
13 for	879,955	62 can	232,575
14 i	870,516	63 no	230,072
15 's	784,710	64 up	221,426
16 ‘	770,022	65 #pause#	216,353
17 -	759,014	66 when	209,661
18 ’	752,179	67 more	209,584
19 on	734,650	68 #unclear#	202,994
20 you	667,997	69 ;	202,921
21 with	659,218	70 out	202,881
22 as	654,472	71 who	200,826
23 be	651,502	72 <i>said</i>	195,327
24 he	640,188	73 about	191,968
25 at	522,328	74 some	167,166
26 by	515,055	75 them	167,144
27 are	464,689	76 <i>time</i>	162,256
28 have	460,956	77 two	161,287
29 this	453,808	78 its	160,359
30 not	451,843	79 could	159,909
31 but	446,187	80 into	157,709
32 from	425,326	81 then	154,663
33 had	420,354	82 other	154,118
34 they	419,905	83 him	153,548
35 his	409,599	84 <i>like</i>	151,820
36)	408,167	85 well	151,254
37 #Overlap#	407,013	86 only	148,991
38 (402,181	87 my	146,706
39 ?	387,955	88 than	144,897
40 or	369,284	89 !	141,764
41 which	365,468		
42 she	351,975		
43 we	350,849		
44 an	337,155		
45 there	319,439		
46 n't	316,412		
47 were	313,268		
48 one	305,411		
49 her	303,264		

Arguably, for a corpus the size of the *BNC* the cut-off could be stricter, as items qualifying for inclusion include some punctuation marks and the only items in the next 40 rankings with lexical meaning are “said”, “time” and “like”. However, it is worth noting that these lists are not visible to the user and the list is designed to limit the amount of pre-processing time and storage space for collocations. The statistical measures should ensure that high frequency items are only visible in the results when the co-occurrence is very frequent, and as will be explained below, the high frequency items will also appear in results showing “extensions³⁰”.

For two-word collocations, this approach may seem interesting or perhaps eccentrically novel, but it does have further power when applied to multi-word units. With the system for calculating multi-word units which I have developed, a particular ordering of 3, 4 or 5 words has to compete with all the other possible orderings of those words as well as the occurrence of the words away from the main node. This is rather different from the approach adopted by Danielsson (2007) where initial selection is more open and ordering is only considered after the extraction has been completed. Using the approach proposed here, the result is that in order for a multi-word unit to make it through the “barrier” of significance, it needs to account for a greater proportion of the combinations in a window. Since BIC values are obtained for all of the multi-word units stored in the summary tables, these can be directly compared.

The sequences considered are limited to consecutive words, so for 3 item multi-word units, for example, the slots are L2+L1+node, L1+node+R1 and node + R1 + R2. It was felt that presenting language learners with a cloud showing X .. Y and Y .. X is reasonably self-explanatory and the words at the top of the list are going to be quite eye-catching, but it would be harder for them to make sense of other possibilities such as XY .. Z, X .. Y.. Z, X .. Y Z, etc. Furthermore, in my system the collocation clouds are there to support access to the concordance lines, rather than to be looked at in detail independently, and the caption feature (where collocates are highlighted in the title bar of the card box) explained below is a complementary way of showing how longer combinations appear in each concordance line.

³⁰ Extensions are additional elements which occur with a collocation and are displayed when a collocation forms the basis of a search query.

Once 3, 4 and 5 word collocations have been extracted, the question then is what to do with the part-phrases which also exist in the tables of shorter collocations. One option would be to use some sort of mechanism to delete from the list ones which occur in longer forms above a certain frequency, like the approach adopted by Shimahata et al. (1999). However, a comprehensive list of two word collocates is used as part of the process for generating captions for collocation lines which is described in Section 4.8.2 below, and it was thought that it might also be helpful to provide learners with the opportunity of obtaining summary information and statistically significant extensions for any two word collocation they enter as a search term. Therefore, the approach adopted was to use a procedure to mark items which occur frequently as part of longer units, thereby permitting *SQL* requests to be made for lists with or without the part-phrases. Figure 4.4 shows the table structure in the database for two, three, four and five word log-likelihood collocations which are named “2mwu”, “3mwu”, “4mwu” and “5mwu” respectively. As can be seen, for each element in the multi-word unit, a link is made to the lexicon table and the primary key consists of the complete combination plus the “cb_type” column which includes information about the position of the node with respect to the other elements and the strength of evidence based on the BIC score. Of course, the structure of the database and naming conventions used are not visible to the user.

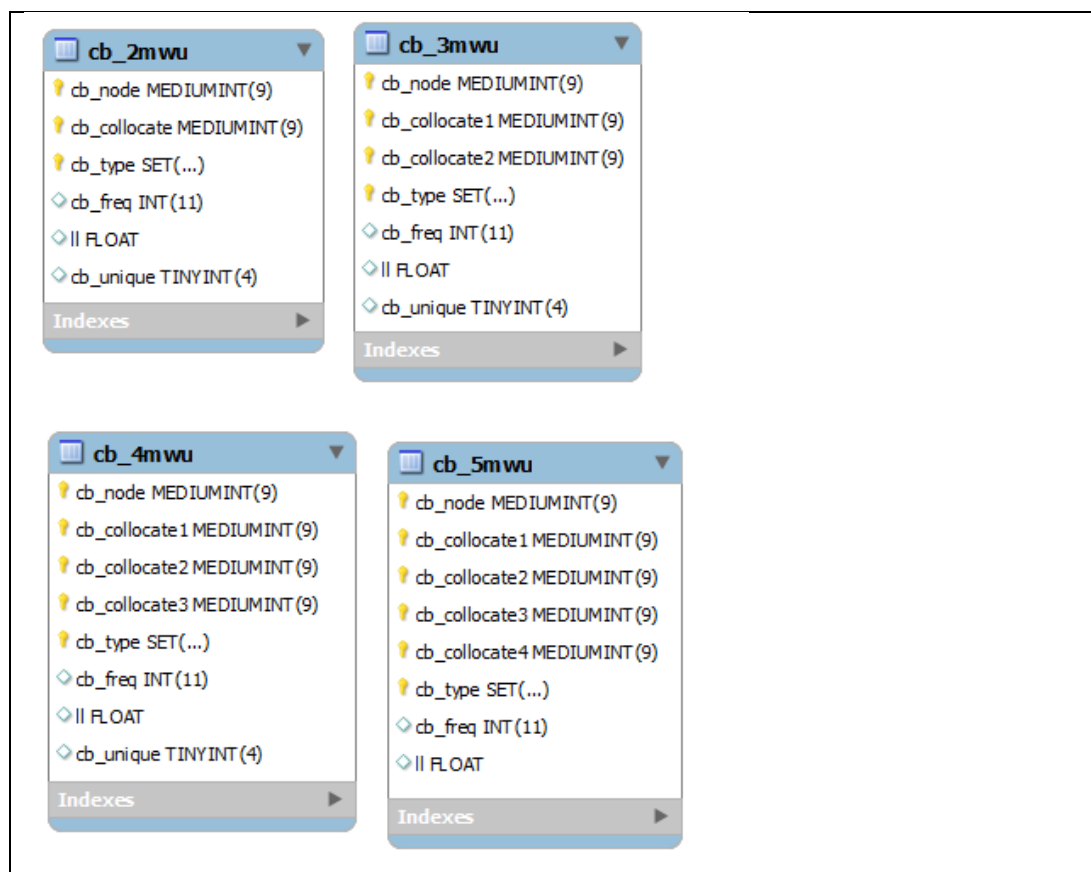


Figure 4.4: Table structure for Log-Likelihood Collocations

Since the search query box in *The Prime Machine* allows users to enter a multi-word unit as a search term, one consideration was what should be displayed on the collocation tab for such queries. If any of the three MI collocation measures are used, without implementing the complicated measures for multi-word units which have been mentioned earlier in this chapter, the only viable option is to show other collocations for the node word. However, with the log-likelihood method, the elements of multi-word units of less than 5 words in length could also be stored in the database in longer combinations. It may also be desirable to show users which words are likely to occur with statistical significance around the chosen multi-word unit if the fact that the words in the shorter combination have been selected together is a given. The contingency table for this calculation is rather less stringent than the main measure, since the proportion of cases accounted for is based on the frequency of the pre-selected multi-word unit, rather than the complete context window for the node. In simple statistical terms, the extensions to multi-word units are looking at the likelihood of encountering each additional element given that the others have been selected. The contingency table for extending two word multi-word units to three word multi-word units is shown in Table 4.6 below.

Table 4.6: Contingency table for extending collocations

	Corpus One	Corpus Two
Freq. of word	A = 3 * frequency of the sequence	B = frequency of word outside this sequence
TOTAL	C = 3 * the frequency of the 2 word sequence	D = Whole corpus – (3* the frequency of the 2 word sequence)

After collocations have been extracted and stored for each order and length in the *SQL* collocation calculation procedure, extensions are immediately calculated too. Thus, two word collocations with no gap between them are used as the basis to extract three word collocations containing the two words together, while two word collocations with a gap are used as the basis to find statistically significant candidates to occupy the middle space.

Figure 4.5 shows the structure of the tables holding these extensions in the database.

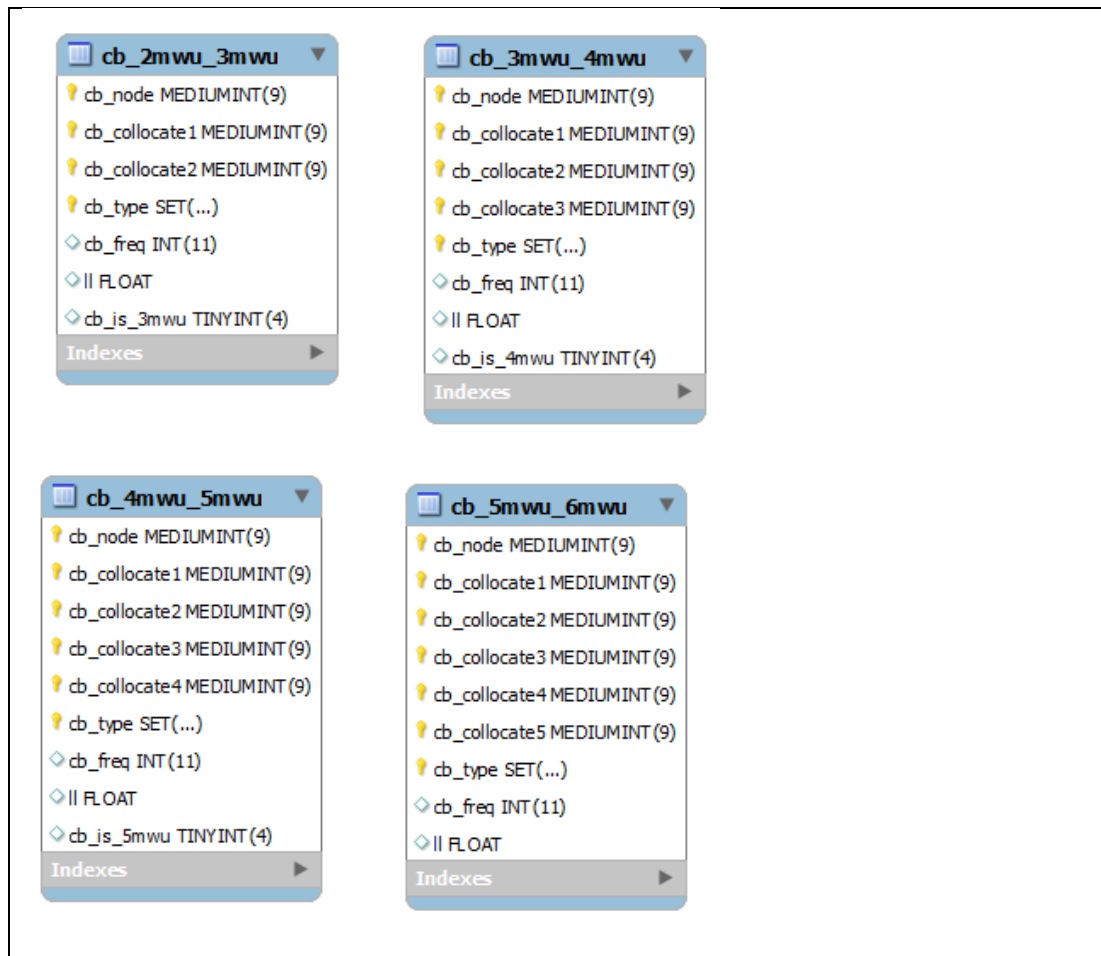


Figure 4.5: Table structure for Collocation Extensions

Once all the “extensions” and MWUs have been extracted and stored, the extensions are also marked according to whether or not they are also stored as collocations in the regular MWU tables. This last step is necessary in order to filter out these extensions from the results, and longer collocations which meet the threshold level in and of themselves will also be displayed in the table. These extensions are not used in the text prediction routines and do not have other summary data, but they do appear as items in the results if multi-word units are used as the basis of a query. At the top of the table of results for longer collocations of a multi-word unit query, the query string, the log-likelihood and BIC score are shown above the list of extensions and longer multi-word units containing the query in order to emphasise the strength of association for the initial query.

4.8 Uses of collocation results

4.8.1 Collocation clouds and tables

On the Collocations Tab in *The Prime Machine*, collocation clouds and tables are available for the Log-likelihood collocations just as they are for those using the mutual information collocation measures. However, there is an important difference. Since the log-likelihood collocations are based on specific ordering and proximity of the collocates, it is possible to present each as a complete collocation rather than isolated words. In this way, the items in the cloud should provide a stronger impression and provide learners with the opportunity to experience the phenomena introduced in another of Firth’s memorable assertions:

A word in a usual collocation stares you in the face just as it is.

(Firth, [1951]1957, p. 182).

The point is that learners may need to see the words together for these visual representations of the collocations to have an impact. The design is based on the proposition that if only the usual collocation word clouds were presented, the same information may be retrievable, but the user would need to be thinking about the node word and formulating a plausible ordering or grammatical relationship for each link. However, if the node is plainly visible in each element in the cloud, it makes the cloud rather “thicker” but ensures the whole relationship can be seen. One of the common things in Chinese dictionaries in particular is to replace the headword with ~ in phrases and all the examples, presumably to save space (and in a paper dictionary ink and paper). However, the ~ symbol probably does not really facilitate the sub-conscious linking of the

two words in the reader’s mind. Figure 4.6, Figure 4.7, Figure 4.8, Figure 4.9 and Figure 4.10 show examples of the clouds and tables for multi-word units for the node *outcome* in several different corpora.

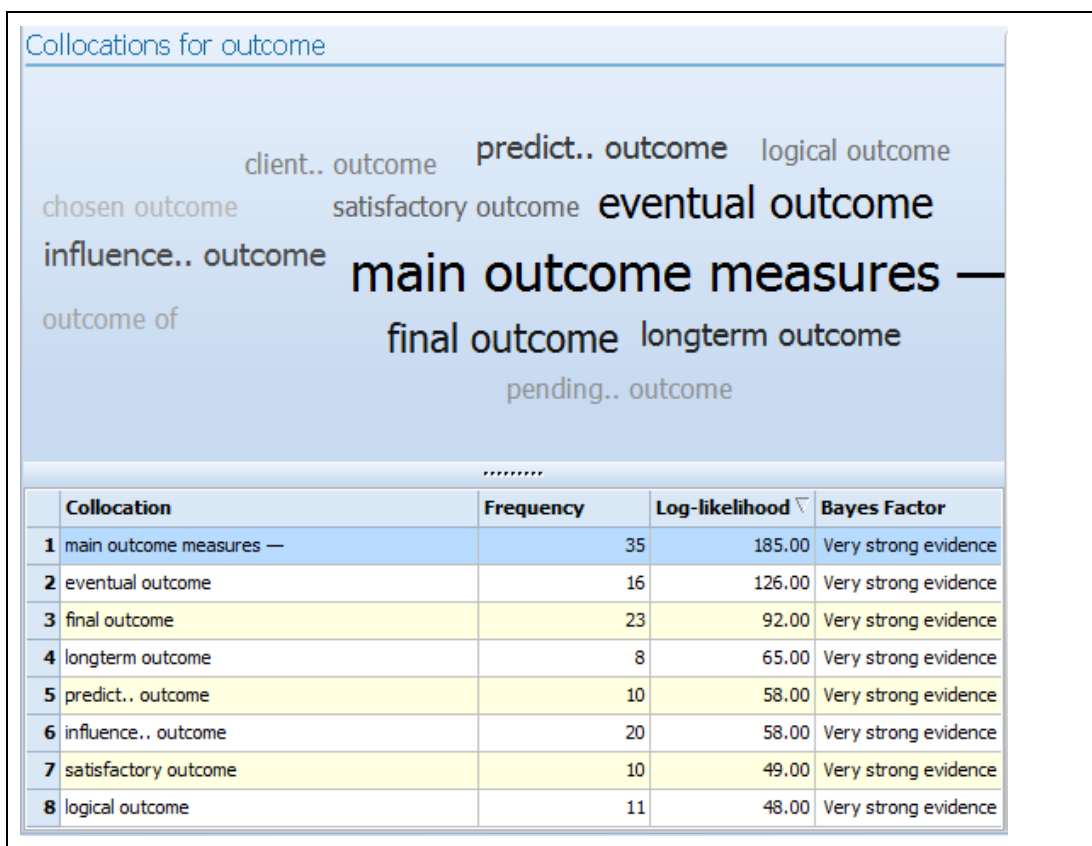


Figure 4.6: Log-likelihood Collocation Clouds and Tables in the *BNC: Academic* sub-corpus for the node *outcome*.

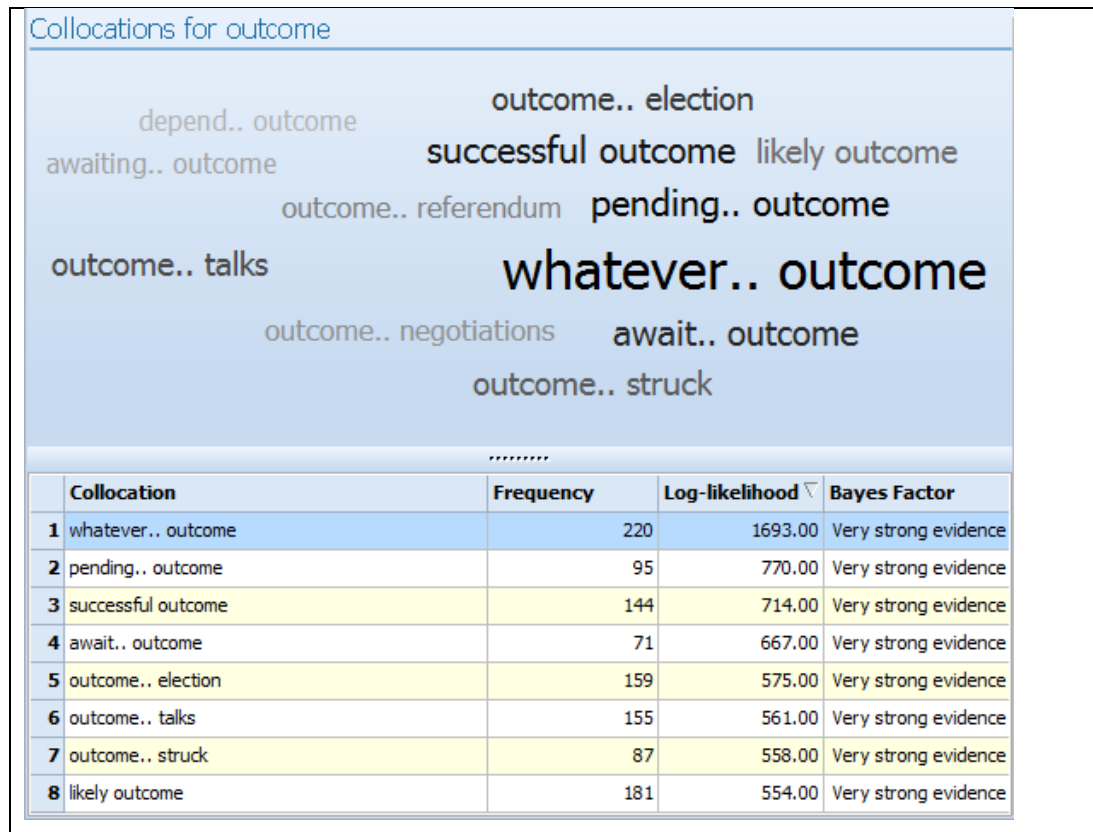


Figure 4.7: Log-likelihood Collocation Clouds and Tables in the *Financial Times* corpus for the node *outcome*.

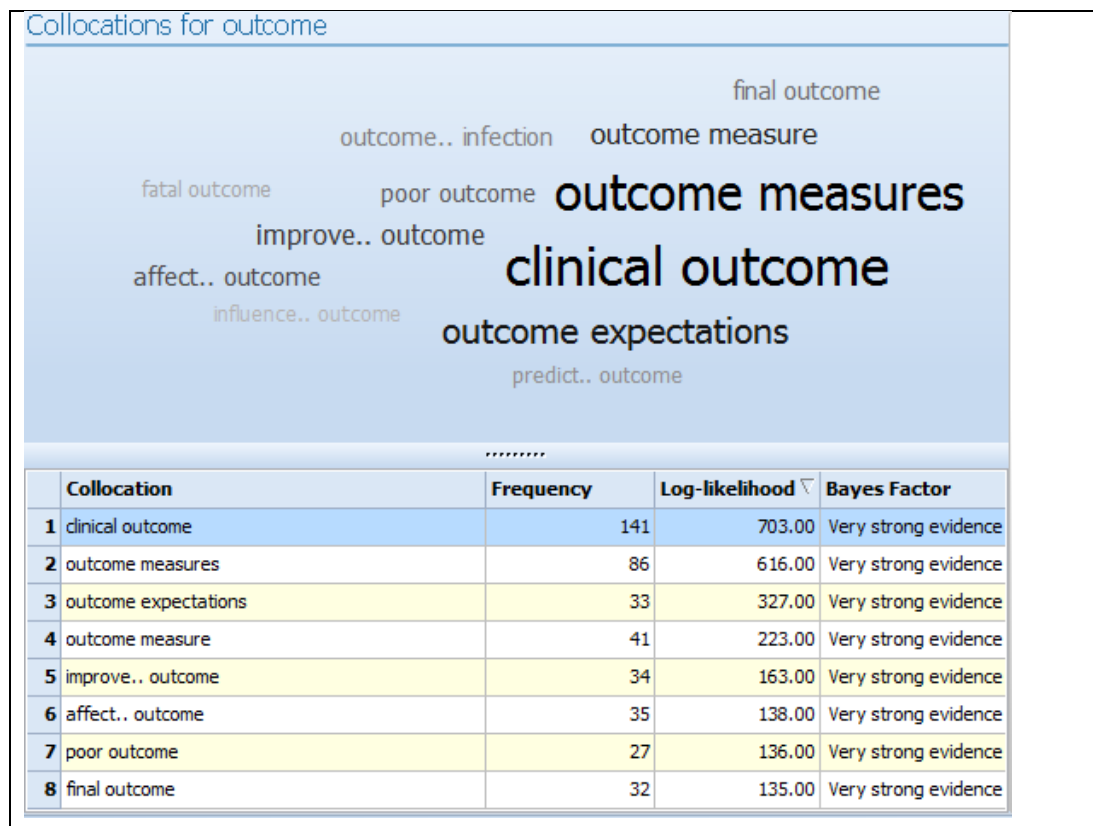


Figure 4.8: Log-likelihood Collocation Clouds and Tables in the *Hindawi Biological Sciences* corpus for the node *outcome*.

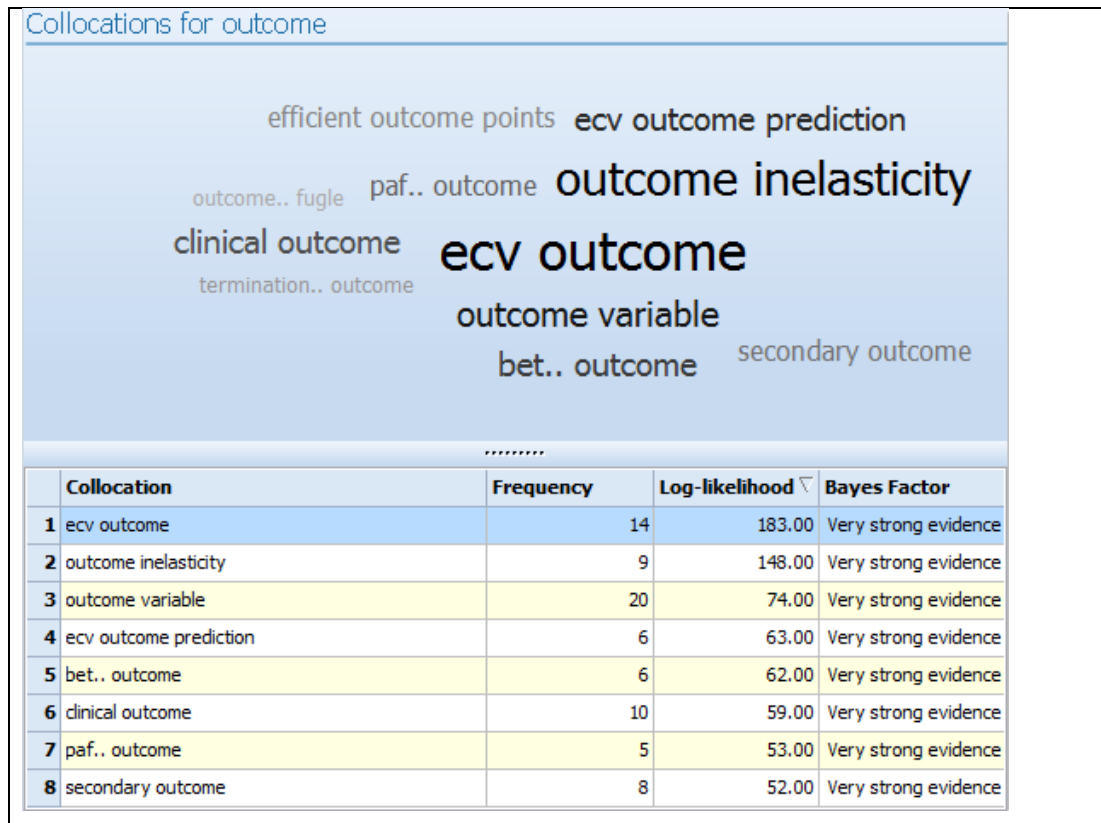


Figure 4.9: Log-likelihood Collocation Clouds and Tables in the *Hindawi Mathematics* corpus for the node *outcome*.

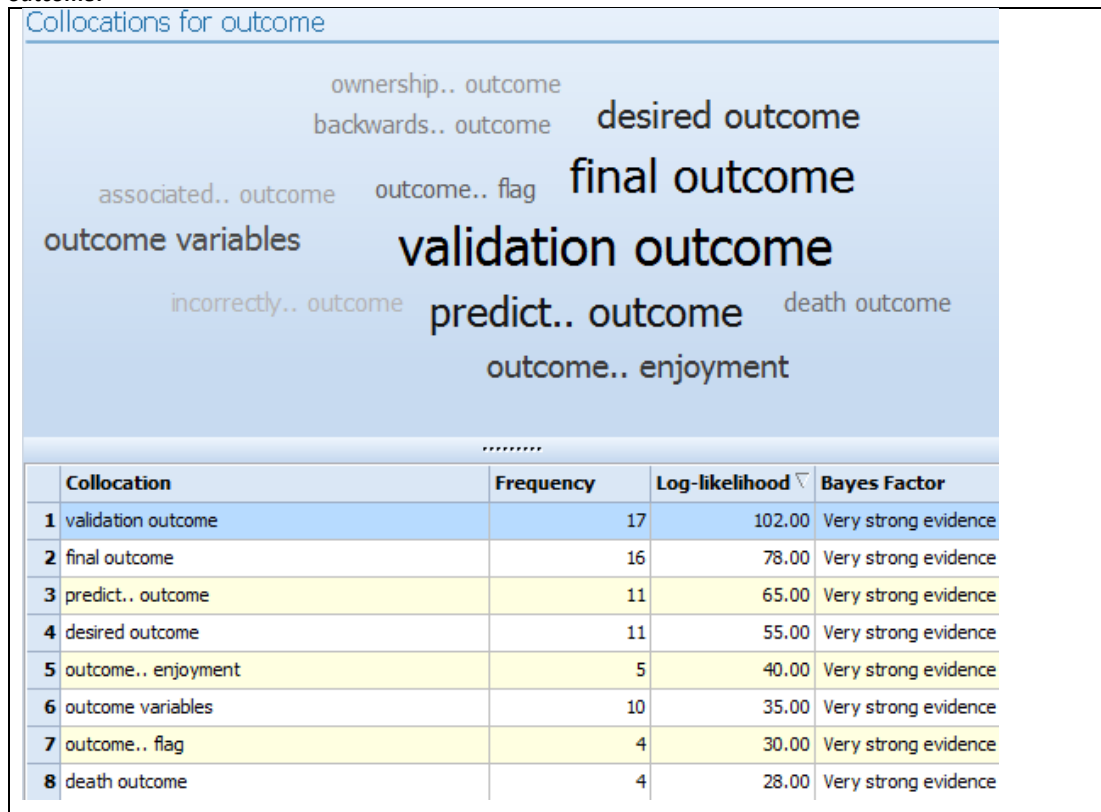


Figure 4.10: Log-likelihood Collocation Clouds and Tables in the *Hindawi Computer Science* corpus for the node *outcome*.

4.8.2 Collocations and concordance lines

The log-likelihood collocations are also used for a number of other purposes in the software. One of the benefits of looking at collocation lists in a concordancer rather than as a separate resource is that the user can explore the actual concordance lines which were the basis of evidence for the relationship. Other concordancing software tries to make links between collocations and concordance lines clear, and in *WordSmithTools* and *The Sketch Engine*, as explained earlier, the user first requests concordance lines and then moves on to generate lists of collocates. *The Sketch Engine* provides links marked with “+” and “-” so concordance lines can be displayed showing positive or negative evidence for the relationship. In *AntConc*, the list of collocates appear like hyperlinks and clicking on them takes the user to a list of concordance lines containing each one. Generating concordance lines for collocates in *The Prime Machine* would entail right-clicking on the desired collocation and then selecting the menu item to use this as a search term. Figure 4.11 shows the context menu which is provided for any text on the screen whether it is an item in a cloud, a cell in a table or a line on a card. Since not all the concordance lines for a query are usually downloaded and stored on the user’s computer, getting concordance lines for collocations would require a further look-up process. Therefore, immediately jumping to the relevant concordance lines is not possible, but the context menu is consistent across the application and presents simple buttons to copy the text to the operating system clipboard, use the text as a main query, use the text as a query for comparison or to use the text in a tag search (explained in Chapter 6). The right-most button copies the text to the compare corpus screen on the “Search Tab”.

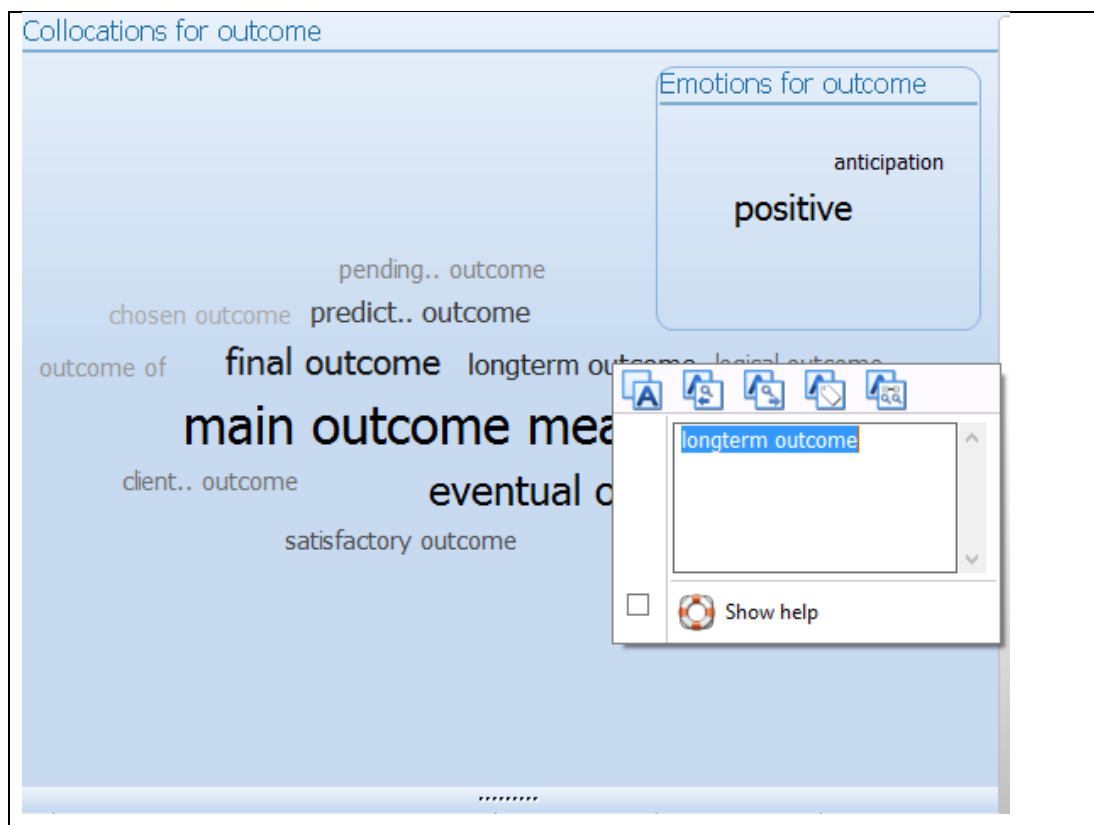


Figure 4.11: Context menu for clouds and collocation table; cloud showing data from the *BNC: Academic* sub-corpus for the node *outcome*.

While it has been recognised that in order to access some information it may be necessary to have longer contexts than the standard KWIC concordance line (Hunston, 2002; Sinclair, 1991), as many researchers have asserted, there are some advantages of viewing vertical lists of truncated sentences four words either side of the search term. Being able to see a large number of results provides a degree of “safety” for conclusions which the user draws (Mair, 2002). They can provide a “snapshot” of how lexis is usually used (Johns, 2002), can be seen as focusing on the “central” and “typical” (Hunston, 2002), and can be organised in such a way as to highlight patterns (Gaskell & Cobb, 2004). It has also been suggested that KWIC concordance lines can free learners from getting caught up in the story or message of a text so as to be able to focus on the language (Cobb, 1999; citing a problem raised by Mondria & Wit-de Boer, 1991). Sinclair (1991) suggests this same freedom is important for researchers, as the KWIC view provides access to patterns which are not meaning-bearing, allowing the distinction between the “physical objects” of text in the corpora and their meanings to be clear.

However, for a corpus engine built on the theory of Lexical Priming, it would seem this is not sufficient. One challenge for this project was to find a way to present a much wider

context than usual in a way which also facilitates visual scanning of patterns, while at the same time enjoying many of the benefits of KWIC. The importance of longer contexts will be considered further in Chapter 5, as it relates to the presentation of data based on other features of Lexical Priming. However, each of the cards on the Cards Tab view of concordance lines which is unique to *The Prime Machine* has a caption which highlights the relationship between the concordance line and collocations. Each caption includes the node as well as any significant items from the top 100 two word collocation log-likelihood lists. When the cards are generated, the 4 word window either side of the node is checked to see whether the items are also present in this list. If they are, they are included in the caption, with “..” added between non-consecutive items. The caption also appears on the card for the currently selected row on the Lines Tab. There are two key benefits of these captions. Firstly, each provides an eye-catching snippet from the concordance line which is essentially a trimmed down KWIC of 4 words either side of the node. Secondly, they help the learner see the main use of the node in each box and should help highlight patterns so users can scan down the list of captions and see which collocations are shown in each box. Since the strongest collocations are labelled in this caption box, it provides a further key to the ordering and relationships between words beyond the limits of the all-in-a-row multi-word collocations which are visible on the Collocations Tab. Because the cards are shown as paragraphs of text with word-wrapping, frequently the node word does not occur in the centre of the line, and words within a 4 word window either side of the node can often be on the preceding or following line. Figure 4.12 shows three cards with their captions for a search on the *BNC Academic* sub-corpus for *outcome*.

<p>eventual outcome</p> <p>... The main role of the therapist at this stage will be to listen and to further his assessment by careful questioning. At the same time, realistic encouragement should be given to ensure that the patient remains hopeful about his eventual outcome. For the patient showing extreme distress, a tranquillizer may be indicated. ...</p>	<p>influence .. outcome of .. process</p> <p>... Instead the trial proceeds on the basis of oral evidence given by witnesses who are called by the parties and examined in much the same fashion as in England.</p> <p>Whether the curbs on police investigation will reduce police influence on the outcome of the criminal process is not easy to determine. It is notable that although the police's formal powers of interrogation during the first 48 hours are limited, their informal opportunities are not. ...</p>
<p>eventual outcome</p> <p>... The view that upsetting issues should be avoided is one that has already been addressed. Providing counsellees can be coaxed into exploring these more sensitive areas, and counsellors feel that they have the time and the ability to cope with any resulting distress, then the eventual outcome can be, more often than not, extremely valuable.</p> <p>Reminiscence can play an important part in ameliorating personal distress, and can be adapted when counselling older people through such matters as retirement, dependence, depression, ill-health, as well as coming to terms with bereavement and death. ...</p>	

Figure 4.12: Collocation-based captions on cards; screenshot showing data from the *BNC: Academic* sub-corpus for the node *outcome*

As can be seen, in the first card the phrase “eventual outcome” appears together on one line, while in the second card “eventual” happens to occur at the end of a line, so the phrase is broken by a line break. The third card shows how the caption may include several words within the 4 word window either side of the node. The caption, therefore, provides an important way of helping learners see nearby words which have a strong relationship with the node, without disrupting the flow of text. Including collocates in a caption goes some way towards overcoming Kenning’s (2000) concern that language learners may need help in seeing how a search term is actually part of a longer unit. It should also support teachers wanting to follow some of the other recommendations in the literature; recommendations such as teaching learners how to note collocations by drawing attention to extra words around a collocation (Michael Lewis, 2000a, p. 134) and directing learners away from separate word analysis (Siyanova & Schmitt, 2008). These pedagogic perspectives also contributed to the formulation of the Auto-Complete processes described in Section 4.8.3 below.

For the creation of clouds or tables, or for highlighting items in a concordance line, it would be possible to calculate multi-word units on the fly. High frequency items would take several seconds to calculate, but presumably there would be some duplication of requests for popular search items, so the database may automatically cache some results making it even faster. On the fly processing would mean that taking the time to pre-calculate multi-word units before making the corpus available would not be necessary. However, the log-likelihood collocations are also used in a number of other ways and for these functions pre-processing is necessary.

4.8.3 Collocations and search query formulation

Computer users are familiar with multi-word units appearing as they enter queries into various search boxes across different applications and websites. It is a familiar experience for online shoppers and several other areas of data retrieval. In *The Prime Machine*, since they are extracted, stored and indexed in advance, short lists of collocations can be retrieved very quickly, allowing text prediction beyond single words to be implemented. While the software will of course allow students to type in complete phrases, many may begin with just one word. Just as Auto-Complete on a word level provides a way of preventing spelling errors, Auto-Complete on the phrase level helps prevent users from making further typos or spelling mistakes and can also provide almost instantaneous feedback on the collocation strength of two or more items. If more than one word has been entered in the search box, when the "Search" button is clicked, the system performs an additional check to ensure that (1) all the words in the box occur together in a 5 word span (i.e. node +/- 4 words) at least once in the corpus, and (2) to check whether the multi-word unit has been stored as being statistically significant using the log-likelihood measure. As will be explained in the next two chapters, information about statistically significant environments for collocations as well as individual items is stored in the database, so the software needs to determine whether or not these data will be available. Within *The Prime Machine*, the following algorithm is used for search strings containing at least one word break³¹.

1. The string is checked to ensure only one occurrence of double period is included.

This has a special meaning in *The Prime Machine*, which follows the display of

³¹ In *The Prime Machine*, word breaks would typically be indicated by spaces, but some other punctuation marks are also interpreted as breaks between words, matching the tokenization rules used in the refactoring process.

multi-word units in that “..” indicates a required gap between two items in a two word collocation pair of at least one word.

2. The string is checked to see whether any of the special symbols have been used to indicate the user is explicitly requesting only concordance lines. These are referred to as *raw window searches*, and the purpose of these will be explained below.
3. The words and symbols contained in the search string are passed to the middle tier, where a check is made for all multi-word units of that length contained in the corpus in any order, with or without gaps. A search is also performed to see whether there is at least one occurrence of the words co-occurring in a 5 word window in any order, whether they occur in order with or without gaps, and whether they occur in order with no gaps.
4. The results are then checked to see whether the original string is included in the list. If so, the search is permitted and concordance lines, collocations (with extensions) and all the other data will be retrieved for the multi-word unit. The item in the phrase with the lowest frequency is used as the node as this provides a performance gain since it is often considerably quicker to go through a short list of occurrences from the index for a low frequency item and then check to see whether matches can be found for the higher-frequency items than it is to use the longer list as the basis for the sub-query on the database. If the original string is not included in the list, the user will be presented with a list of phrases containing the same words but in different orders, and information about whether they occur at least once in each of the three raw window searches. A button also appears which allows them to check to see whether the words they have entered appear together in any of the other corpora. This latter option is the phrase level equivalent of the spell-check routine which tries to reduce the possibility that a user is unable to find something which is available just because they have not chosen the most suitable corpus.

This lookup procedure provides a way to give very quick feedback on whether or not words collocate and whether there are any instances of the phrase in the corpus at all. The results of a study by Römer (2009) into other software for language learning and teaching include a suggestion from a teacher that it would be helpful to have very quick feedback on whether words collocate or not. Lewis (2000) argues that the development of collocation knowledge includes greater awareness of words which should not be used together as well as those which should. While lack of a collocation relationship is not something visible on

any of the results tabs in *The Prime Machine*, the immediate feedback goes some way to meet these needs and should help learners see if a phrase they are considering using may not be appropriate. The look-up phase also has a gate-keeping role, preventing the fruitless waiting period which would occur if users requested a phrase which simply did not occur.

It was hoped that the log-likelihood method of extracting collocations would provide good coverage of useful collocations and the summary information for the environments in which they occur would be of interest to learners. However, the gate-keeping process needed to allow raw window searches for two reasons. Firstly, since the most frequent lexical items are only stored as collocates (rather than nodes), if a user looks up something as frequent in English as “of the”, it would be extremely misleading to report that this does not occur. Secondly, not all users may want to base their searches on the log-likelihood measure or be interested in the other data and for these users concordance lines may be sufficient for their needs. However, as explained in the previous chapter, it did not seem appropriate to develop a highly complex query language, as expecting users to be able to formulate such queries would seem unreasonable. Tips explaining the use of the special raw window search operators will appear, but since the drop-down box contains information as to whether each of the three operators will provide results, users can simply click on these to request concordance lines for a specific string. Effectively, the software does the work itself of formulating queries containing the |, _ or * symbols which correspond to the words occurring in any order, in order with no intervening items, or in order with or without intervening items.

The drop-down boxes for collocations appear as soon as the word-level Auto-Complete routine encounters a string of letters which is listed in the lexicon as a complete word. The top collocations are then retrieved for this word as the node and ordered by log-likelihood. In this way, three, four or five word MWUs will be included in the drop-down list if their statistical significance rather than their raw frequency places them higher in the ranking. As with the lists of individual words, Auto-Complete suggestions include the frequency of the collocations in brackets, providing instant information about the number of instances available. For raw window suggestions calculation of raw frequencies is not practical, and the confirmation that they are attested in the corpus is based on a *SQL* query limited to the first hit, so the brackets simply show “>0”. As mentioned in the previous chapter, the compare feature has also been developed in order to encourage learners to explore and

compare alternative wordings. Collocations which contain words with the same stem and/or the same words in a different order appear on the right-hand side of the Search Tab to encourage users to compare these with the word or phrase they have formulated as a main query. Figure 4.13 shows how these Auto-Complete suggestions appear on screen.

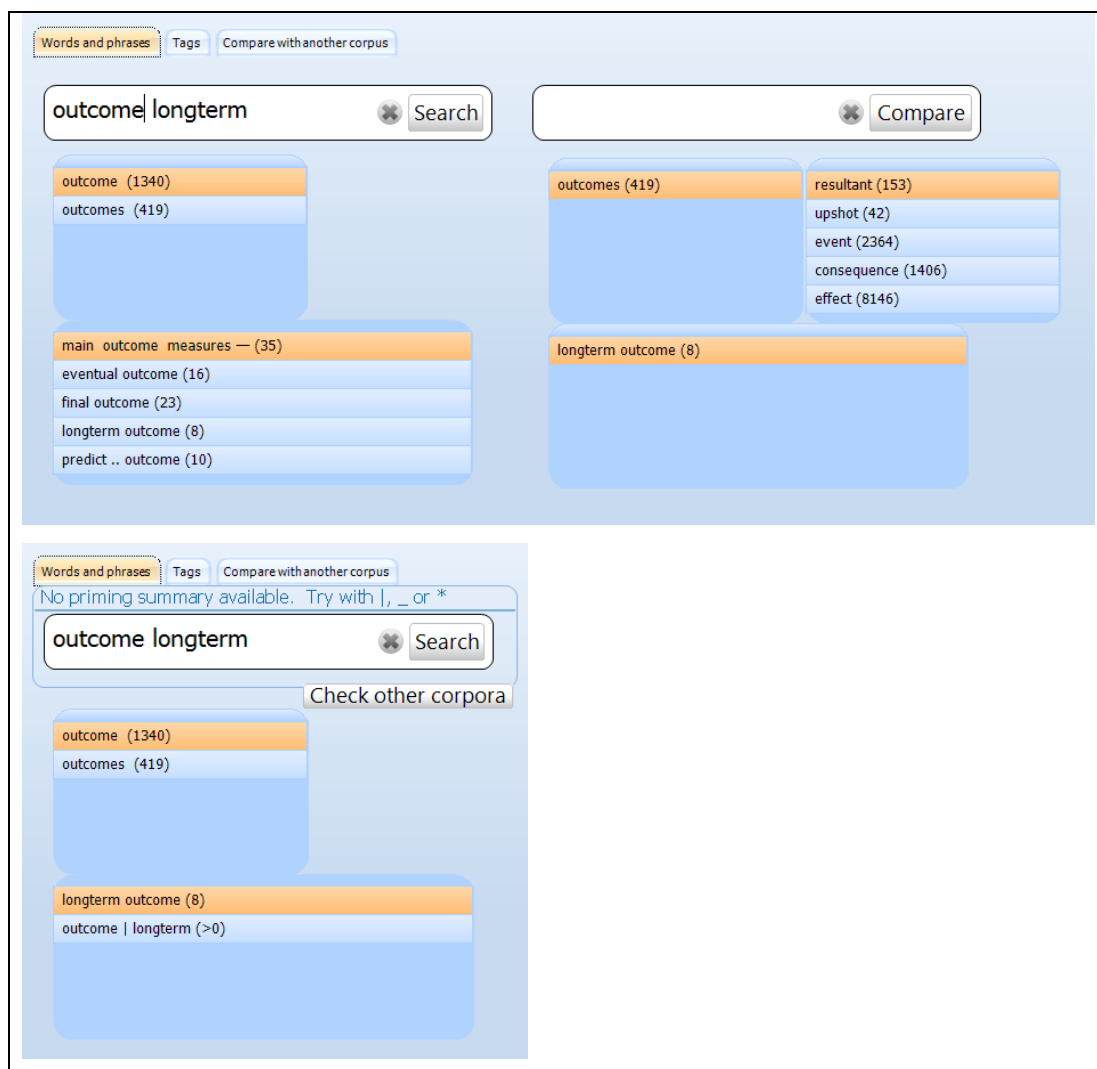


Figure 4.13: Auto-Complete suggestions showing collocations and raw window search queries; showing data from the BNC: Academic sub-corpus for the query *outcome longterm*.

4.8.4 Collocations and indications of semantic association

On the Collocations Tab, a box of “emotions” can also be shown, giving an indication to learners of how a word may be used in the company of highly emotive words. This additional panel of information about the node word is designed to provide at least some minimal coverage of one aspect of Lexical Priming which is both difficult to measure through automatic means and has very limited representation in current lexical resources. Semantic prosody may be described as a hidden connotation-like quality which words may

have resulting from common use with other words. Although there are some differences between the conception and definitions of semantic prosody and semantic association, both include the possibility for emotive charging of words through their frequent use in specific contexts. A fuller exploration of the historical development of theories and understanding of semantic prosody and semantic association is provided by Stewart, where he also calls for “serious improvements” in descriptive works for learning English (Stewart, 2010, p. 263). In a cross-linguistic study of English and Chinese, semantic prosody tendencies have been shown to differ for similar items across different languages, and this evidence suggests that a lack of knowledge in this area could account for some of the struggles non-native speakers face (Xiao & McEnery, 2006). A recent review of dictionaries available in China has highlighted the fact that this kind of information needs to be made a high priority (Ping-Fang & Jing-Chun, 2009).

Through providing automatic prompts encouraging learners to compare words or phrases and view the results side-by-side on screen, some aspects of semantic association should be fairly clear. For example, Figure 4.14 shows the collocation clouds for “result” and “consequence”. The intention is not to specifically teach the terminology of semantic prosody or semantic association, but the collocation clouds should in many cases provide sufficient evidence for how some words tend to be associated with positive or negative contexts, as well as wider categorizations or groupings which a teacher or advanced learner may be able to make.

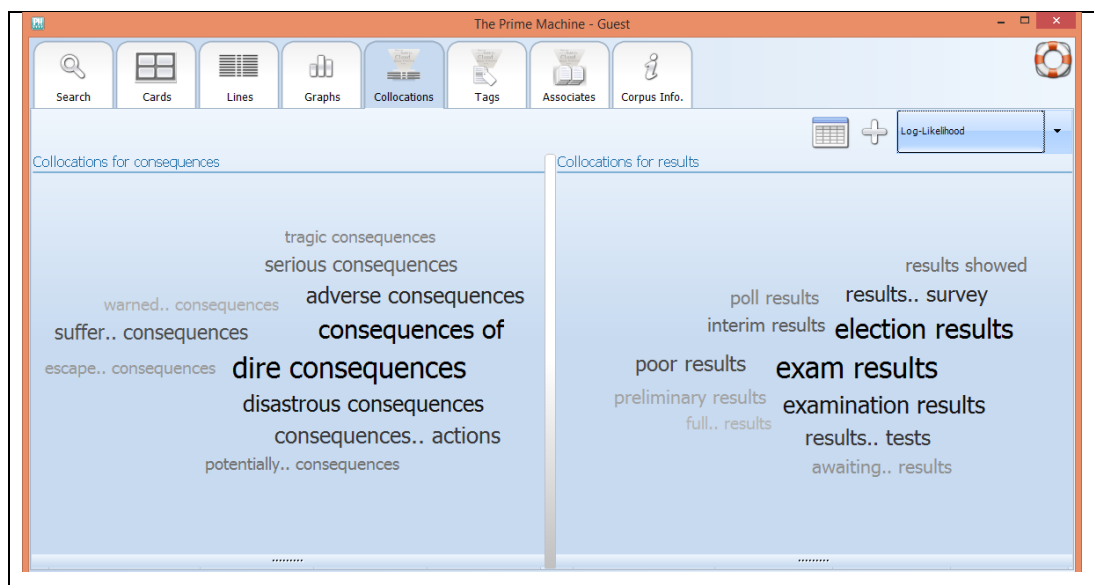


Figure 4.14: Two collocation clouds displayed side by side for the nodes *consequences* and *results* from the *BNC: Newspapers* sub-corpus.

An aspect of semantic prosody which has been discussed is the degree to which it needs to be “hidden”. One way of trying to capture a sense of semantic prosody automatically would be to look at the emotional charges of a node’s main collocates. An approach based on emotions should provide a middle ground between narrower definitions of semantic prosody which limit it to positive or negative charging and the broader possibilities outlined in Hoey’s (2005) exploration of semantic association. Sentiment Analysis is a growing field within Information Retrieval and Natural Language Processing, with an increasing number of electronic resources holding information about associations between words and emotions being developed. Devitt and Ahmad (2013) present a review and analysis of four widely used resources in this field: the General Inquirer lexicon, Dictionary of Affect in Language, SentiWordNet, and WordNet-Affect. Other resources include LIWC (Pennebaker, Chung, Ireland, Gonzales, & Booth, 2007) and the UCREL Semantic Analysis System which has “Emotions” as one of its categories (Rayson, Archer, Piao, & McEnery, 2004). Mohammad and Turney (2012) developed a relatively small lexicon of words which when presented in isolation are associated in people’s minds with specific emotions. Using a method called “crowd sourcing”, participants were recruited through the internet and asked to identify emotions which they felt were associated with the items in the lexicon. In their paper they explain that greater agreement was found between raters if they were asked which emotions were “associated with” the emotions rather than which “evoked” the emotions. The choice of emotions available were ‘anger’, ‘anticipation’, ‘disgust’, ‘fear’, ‘joy’, ‘negative’, ‘positive’, ‘sadness’, ‘surprise’ and ‘trust’. Through combining the results

from the participants, they provide a list of words and emotions as a lexicon and have made it available for research.

A word list of semantic senses or associated emotions could be used in several ways. Rather than presenting a list of emotions for the relatively small number or types in the emotion lexicon, in *The Prime Machine*, a linking procedure was designed to try to indicate some of the common emotions exhibited in the collocations of an item. First, types in the lexicon are linked to the emotion lexicon, so all the words which occur in the corpus which are also included on the list have a link. The emotion lexicon includes separate entries for some items in singular and plural and in different word forms. The use of a stemming procedure was considered, but since there were items in the emotion lexicon which had different values for past and present tenses and seemed to be associated with very different emotions according to word form, it was decided it would be better to base the analysis on specific types rather than stems. In order to provide a baseline for the coverage of emotion-linked lexical items, a table of frequencies for each emotion is then set up for each corpus as part of pre-processing. Since the measure only uses the log-likelihood collocations stored for two word collocations, it is these frequencies which are counted. Next, each lexical item is examined in a loop which generates contingency tables for log-likelihood statistical testing, following the same sort of approach which is used for the extraction of collocations.

Table 4.7: Emotion-Linked Collocation Contingency Table

	Sub-Corpus 1	Sub-Corpus 2
Emotion	A = Sum of the frequencies of collocates listed in the emotion lexicon for a specific emotion	B = Sum of the frequencies of all words listed in the emotion lexicon for the specific emotion – A
Other emotions	C = Sum of the frequencies of all collocates linked to any emotion for the node	D = Sum of the frequencies of all words listed in the emotion lexicon for any emotion - C

As shown in Table 4.7 above, since the emotion lexicon does not cover all lexical items, it seems more appropriate to compare the relative frequency of each emotion as a proportion of the total number of collocates matching any emotion against the proportions for all lexical items. Tendencies are stored in a separate table in the database using the node + emotion combination as a primary key (Figure 4.15).

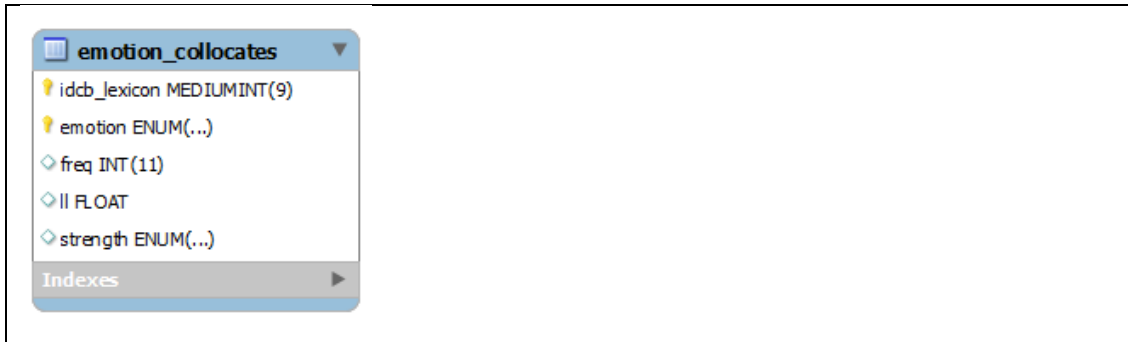


Figure 4.15: Table structure for emotional charging of words derived from their collocates

It is worth noting that the opposite of a positively “charged” emotional item is not necessarily a negatively “charged” one, but rather a lexical item which does not show statistically significant co-occurrence with emotionally charged words at all. This point is conveyed to the user through a number of graded tips (displayed while results are being downloaded) and summarized on the help screen for the Collocations Tab. The emotion clouds cannot be viewed separately, and are intended to be supplementary to the collocation clouds and tables. Figure 4.16 shows the same collocation clouds as Figure 4.14 but this time with the emotion panels visible.

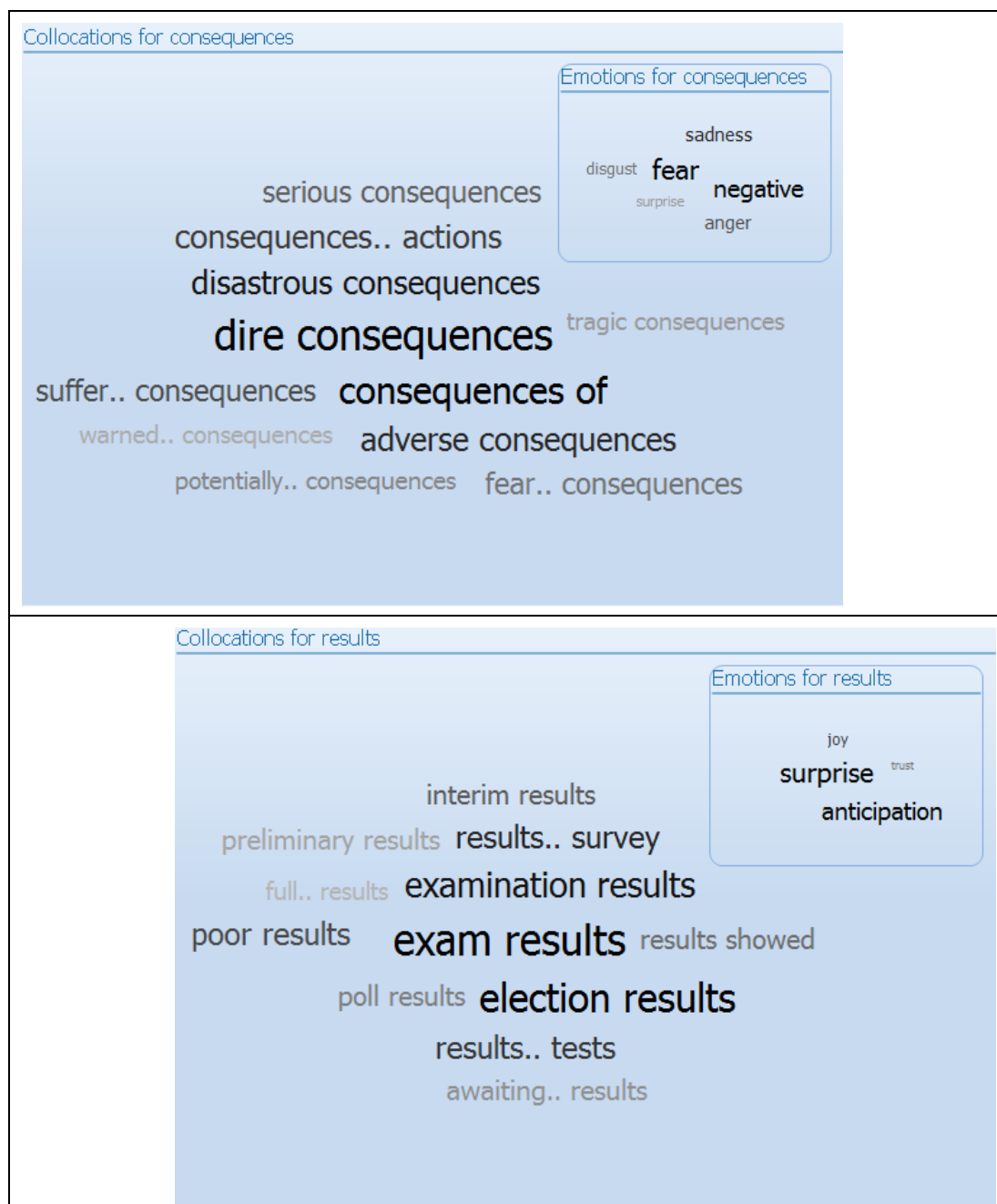


Figure 4.16: Collocation and emotion clouds for the nodes *consequences* (top) and *results* (bottom) from the *BNC: Newspapers* sub-corpus.

It would be interesting in future to explore this kind of technique with other resources. Just as *WMatrix* uses UCREL’s semantic tagger (Rayson, Archer, et al., 2004) to show key semantic tags for two documents or two corpora, *The Prime Machine* could use an externally produced lexicon resource to show which semantic tags are key for a node word. The emotion lexicon currently implemented is a little limited and is based on intuitions of online users paid to respond to decontextualized prompts. Since the development phase of this current project, other resources have been released such as *DepecheMood* (Staiano

& Guerini, 2014) which have larger word lists and use more context in the crowd-annotation task and these may be a fruitful way forward. However, the implementation used in *The Prime Machine* is based on shorter contexts within the actual corpus data. An alternative might be to channel whole sentences or whole texts through this process. One risk of utilizing semantic tagging at an earlier stage by tagging texts before they are imported into the database would be that results would be somewhat circular, as more sophisticated taggers use words in the context as part of the assignment process. Nevertheless, approaching semantic association through looking at common semantic categories in the collocations of a node seems to be one way forward and a platform such as the relational database used in *The Prime Machine* would be one way to operationalize this. In the meantime, despite the lack of a more comprehensive automatic approach it is hoped that the insights they gain through direct access to concordance lines and collocations should help teachers and learners notice other tendencies for semantic association. Some of the ways in which the text and section level analyses may also assist with this are briefly explored in Chapter 6.

4.8.5 Collocations and concordance line ranking and selection

For researchers using concordancers to identify the range of ways in which a word is used, given the frequency of many words in corpora and the time it takes to analyse and categorize each line, it is usually not possible to examine in detail all the concordance lines for a particular search. Therefore, selection and ordering of a sample of concordance lines needs to be carried out in a systematic way (Sinclair, 1991). A very common way of reducing the number of concordance lines for analysis in concordancing software is to provide random sampling. Sinclair (1991) proposes a sample retrieval–analysis cycle ending when no new patterns emerge. In order to determine how many concordance lines are likely to be necessary to provide good coverage, Hoey (personal communication, 25 June 2013) looked at this problem from the other direction, beginning with as large a set of concordance lines as is allowed in *WordSmith Tools*, and working downwards in sample size until a point was reached where patterns of usage considered important began to disappear. For a researcher wanting to spend a significant amount of time exploring a range of uses, concordancers need to provide a means of obtaining samples of different sizes. For a learner concordancer, however, there are a few problems with leading them too forcefully towards random sampling. Firstly, the number of instances required to provide a good overview is likely to be well beyond the patience or skills of language learners, particularly at intermediate levels. Secondly, it is highly likely that when language

learners do concordancing, they will frequently be following directions from a teacher or be working in a more collaborative way. If searches return a different set of randomly selected results each time a search is made, a search by one student or teacher which happened to contain useful patterns will not be matched by a classmate following their advice. In *SQL*, it is possible to put results in a random order efficiently by simply adding “order by RAND()” to the end of the query, but in order to provide the possibility of sorting results in fixed random order rather than an on-the-fly random order, *The Prime Machine* has an additional column in the corpus_words table which holds a random number that is calculated during the compression process and remains fixed after that point.

With small datasets, selection and ordering of concordance lines is not an issue of vital importance, since the user might be able to view all the results on a single page. However, even for an experienced user of corpus tools, a filtering and ranking function can assist in helping to find regularity or notice patterns and to reduce the sheer amount of data to be inspected. In terms of software design, a good ranking or filtering mechanism is also important as a means to cut down the amount of data required to be sorted and retrieved from the server. Sending hundreds of lines from the database server to the client, only to have a very small proportion selected locally for display is not an efficient system. For a language learner, it would also be desirable to sort concordance results in such a way as to move more “useful” patterns to the top. For example, corpora based on newspaper data may include tables of company stock prices or football results. These would probably not be the most useful examples for a learner wanting to use words and phrases in their own writing. Being able to assign a low ranking to instances of a node which occurred in contexts which do not share characteristics of full sentences would not only help in filtering out tables from newspapers, but also aid in the selection of good examples from corpora which have been created automatically from websites through unsupervised harvesting techniques. Corpora can be built through automatic collection of internet pages using tools such as WebBootCat (Baroni, et al., 2006), and documents may contain formatting, advertisements or other “noise”, so one job of a filtering or ranking system often is to push unusual syntax or punctuation to the bottom of the list. Renouf, Kehoe and Mezquiriz (2004) also explain other issues with internet-collected “texts” such as problems with automatic processing (because of misspellings and unclear punctuation) and problems with statistical measurements as it is not clear what would constitute a complete text. Some techniques for cleaning web data have been developed, including ways to cut out duplication and repetition and to identify web documents which are more likely to be

useful according to their size in bytes (Fletcher, 2004). One of the aims of refining the first version of the GDEX algorithm used in *The Sketch Engine* for concordance line selection was to filter out “lists and other web junk” which often seemed to dominate the top ranking examples (Kilgarriff, Husak, McAdam, Rundell, & Rychlý, 2008).

Some researchers have also looked at methods to make the examples easier for language learners to understand. When the intention is to locate examples for learners (as opposed to researchers) to use, the advantages and disadvantages of grading the difficulty of the corpus through text selection when it is created or the use of filtering methods as part of a query need to be weighed. Wible et al. (2002) proposed a method of filtering out examples from concordance data based on vocabulary profiles. By removing lines which have lower frequency items, this approach aimed to make the concordance lines easier to read. However, one of the lessons the Data Driven Learning (DDL) method seeks to instil in learners is that comprehension of everything is not necessary for them to learn something (Johns, 1988). Filtering according to vocabulary frequency may be appropriate in some contexts, but it is likely to hide patterns. It is also questionable whether concordance lines containing company names should be penalized in the same way as low frequency vocabulary items, particularly if capitalization of the initial letters shows learners clearly that they are proper nouns.

Another aspect of any process attempting to improve the usefulness of concordance lines is a careful consideration of what uses learners want to make of them. Frankenberg-Garcia (2012) makes a helpful distinction between examples in dictionaries or concordance lines which aim to aid *comprehension* and examples aiming to aid *production*. Following this contrast, a concordance ranking or filtering system could be used either to provide dictionary like examples which make the meaning of the word clear in very short contexts, or to provide examples which are representative of its collocations and syntactic conventions. As each of the possible ranking methods which have been implemented in *The Prime Machine* are outlined below, these two orientations will be considered.

For all concordance ranking and selection methods, a very necessary step is to decide how to deal with identical sentences. Just as duplicate hits in information retrieval are undesirable, when identical sentences in a corpus appear in a short list of results they are not usually considered useful and could be distracting. As well as the fact that different writers or speakers may use the same short combination of words as a sentence and actually be using a fixed expression in a variety of contexts, the identical sentences could

also be the result of having duplicate texts or sections of a text in a corpus, or be common structural elements. For example, the *BNC XML* corpus has two versions of some of its news stories with minor changes or edits to only a few of the sentences. In newspaper corpora, it is fairly common for weekend editions of “sister” papers to include the same stories as the next Monday morning edition. In these cases, the identical sentences are a result of the selection method used in the creation of the corpus. In web data, the possibility of duplicate texts or duplicate sections of texts seems to be greater, with problems associated with the automatic harvesting of old and updated pages, as well as the common phenomena of cross-posting and forwarding or mirroring content. For *The Prime Machine*, it was felt that it should remain the responsibility of the corpus manager³² to decide whether other techniques should be implemented to prevent these texts being held multiple times in the corpus before the refactoring process is initiated, and for all the other processes (such as collocation, and priming data), all duplicate sentences in the corpus will be considered as instances of language in their own right. The other kind of identical sentence is repeated use in a variety of different contexts. Headings, for example, are treated as separate “sentences” in the system, so for texts containing headings such as “introduction” and “conclusion”, each of these would also be an identical “sentence”. Longer stretches of identical sentences are also possible with examples from the *BNC*, and from the *Financial Times* shown in Figure 4.17 below. However, for concordance line ranking and selection, there seems little point in presenting identical sentences to the user, and for many of the measures described without some marking of these identical sentences, all the duplicates would receive the same equal ranking and therefore be clustered together in the short list of visible concordance lines.

³² As explained in Chapter 3, the corpus manager would set up the server and pre-process all the corpora. Currently, the plan is to have one server to which all users connect, but if a school or university was to set up their own instance of *The Prime Machine* server software these issues would be important.

<p><i>BNC</i>: Undergraduate prospectus for entry 1992. University of Ulster Coleraine 1991 31 exact matches and 38 very similar instances for: <i>The general entry requirements for admission to a first degree course (see page 51).</i></p> <p><i>BNC</i>: [Hansard extracts 1991-1992] HMSO London 1992 19 exact matches and 54 very similar instances for: <i>I refer my hon. Friend to the reply that I gave some moments ago.</i></p> <p><i>The Financial Times</i>: 16/05/1992:VI 66 exact matches and 585 very similar instances for: <i>No legal responsibility can be accepted by the Financial Times for the answers given in these columns.</i></p> <p><i>The Financial Times</i>: 10/05/1994:6 22 exact matches and 31 very similar instances for: <i>A premium of 0.2 per cent is to be added to the credit rates when fixing at bid.</i></p>

Figure 4.17: Examples of longer duplicate sentences in corpora; exact matches are instances picked up automatically for the sample sentence and “very similar instances” are where almost the exact wording and punctuation was used.

In order to mark duplicates without removing them from the corpus, a procedure was set up using *SQL* to try to minimize the number of comparisons which would need to be made. Comparing every sentence with every other sentence would be the easiest approach, but would be highly inefficient and take a considerable amount of time. Just like with many of the other speed issues mentioned in this chapter, comparing sentences to one another increases with degrees of magnitude as the corpus increases in size. A procedure which worked quickly for a small corpus of 1 million words, or a moderate corpus of 10 million words, could bring the refactoring process to a complete standstill on a corpus of 100 million words like the *BNC*. The fine details of the process outlined below could be further developed and made more efficient, and it may even be desirable to have two systems available; one for small to moderate sized corpora and another for large corpora. The method described below is not of any great interest from a linguistic perspective but was a necessary step.

If two sentences are identical, they obviously must be equal in length and since the number of words contained in each sentence is stored in the database, this is the first way in which the number of comparisons is reduced. When experimenting with corpora of 40 million words or more, the calculation for sentences 1, 2 or 3 words in length took much longer than might be expected. As explained above, part of the reason for this is the fact that short section headings or other elements in a text which CLAWS splits into separate sentences will be adding to this number. Since more than a third of the lexicon of any corpus is likely to be made up of words which occur only once (Croft, et al., 2010), and since these *Hapax Legomena* cannot possibly occur in more than one sentence, looking at the primary keys of the two words with the lowest frequency in each sentence is also a good way to minimize the number of possible comparisons. A third measure is the number of letters or symbols contained in a sentence. Sentences may have the same length in words but be a different length in letters. By working through these three measurements, a list of possible matches is created and these can then be extracted and checked using a “join” statement in *SQL*. If the number of matching words between two candidates is equal to the length of the sentence in words, there is a complete match and this item is marked in database. The first occurrence in the corpus is marked with a number indicating the number of other sentences following this in the corpus which are exact matches³³. The sentences in the database following this are marked with a negative number indicating the number of matches occurring before them.

This means that when concordance lines are retrieved, by requiring that the lines must have a value greater than -1, only the first occurrence of a sentence which has duplicates will be retrieved. If this setting is active, the user can hover over the line number on the Lines Tab to see an additional message showing how many other duplicates occur in the corpus.

The most advanced concordance line selection algorithm currently implemented in any mainstream English concordancer seems to be *The Sketch Engine's* GDEX, which is introduced and explained through its application to the extraction of example sentences for collocations by Kilgarriff et. al (2008). Given that the name of the algorithm comes from

³³ The column in the table which holds this number is a *TinyInt* field which can hold a value in the range -128 to 127. This means if there is a large number of exact matches, the value for this column is actually capped at 127. In the client application, if there are more than 100 matches, the message that is displayed states “more than 100 other matches” rather than a specific number.

“Good Dictionary EXamples”, it is not surprising that it was initially developed to provide lexicographers with easy access to corpus examples which contain fewer low frequency words, and use a fairly restricted vocabulary. Table 4.8 below shows a list of the elements contributing to the original GDEX score, as well as additional columns showing whether each element is the same for all words in a specific sentence and whether it was included in my own implementation.

Table 4.8: Features of GDEX and their implementation in *The Prime Machine*

GDEX feature (as presented in Kilgarriff, et al., 2008)	Constant for all words in same sentence?	Implementation in <i>The Prime Machine</i>
Sentences which are too long or too short	Yes	Sentence level penalty of 10 if: <ul style="list-style-type: none"> less than 10 tokens; greater than 25 tokens.
The number of words in the sentence which are not on a list of the most frequent 17,000 words of the language	Yes	Sentence level penalty of 1 for each word not in list of 15,000 most frequent words in the corpus.
Sentences containing pronouns or anaphors	Yes	Sentence level penalty of 5 for each of the following ³⁴ : <i>this, that, one, it</i> .
Sentences where the target collocation is not in the main clause	No	Not implemented.
Sentences which do not begin with a capital letter or end with a full stop, question mark or exclamation mark.	Yes	Sentence penalty of 10 if: <ul style="list-style-type: none"> The first word is not capitalized; The sentence does not end with a full stop, question mark or exclamation mark.
Sentences which do not contain third collocates	Yes	Not implemented
Sentences where the target collocation is not towards the end of the sentence	No	Word penalty of 10 if node is not in Rheme (see Chapter 5 for details of how this is calculated).
Sentences which have features indicative of “web junk” (several capital letters, punctuation marks or other unusual symbols)	Yes	Sentence penalty of 50 if more than 3 tokens in the sentence are capitalized or are marked in the refactoring process as not to be included in a word count (i.e. punctuation or mathematical symbols, etc.) ³⁵ .

As can be seen, the attempt to implement a similar selection process to GDEX in *The Prime Machine* is fairly limited. Actual weightings used in *The Sketch Engine* are not provided, but the paper does explain that the most important measures are the length of sentences and the penalty beyond the frequency cut-off point. Unlike all the other ranking methods used

³⁴ From the explanation given in the paper, even though these words have uses other than as pronouns or anaphors, it seems that the penalty is applied based on any occurrence of specific words (types) rather than grammatical categories.

³⁵ At the time the paper was written, the rules for “web junk” were described as being still under development, but this penalty for having more than three capital letters or more than three punctuation marks will also penalise, for example, sentences containing names and long sentences with several commas.

in *The Prime Machine*, the higher the GDEX score the lower its ranking, so in order to make it easier to query and to keep the ordering (ascending or descending) consistent for the middle tier server commands, the GDEX value is capped at 127 so as to fit in a TINYINT column in the database and then inverted by subtracting this from 127.

The concordance selection system in *The Sketch Engine* seems to have been strongly oriented towards its early target users: lexicographers. With the expansion of corpora from carefully selected and balanced datasets to large automatically harvested collections from the Web, it is obvious that a conflict would arise between displaying the strangeness of internet text and providing the lexicographers with neat examples which can be used and advertised as being “corpus derived”, while still upholding expectations of being well-formed. Since dictionary examples usually appear as single sentences with no further context, the penalties for proper names, long sentences and unusual words are easy to understand. The preference for examples where the node word is in the Rheme of the sentence is likely, as the paper argues, to favour examples which allow the reader to understand the meaning from context, and with a meaning-focused emphasis for dictionary creation this seems very valuable. However, within Hoey’s theory of Lexical Priming (2005), it is argued that some words are actually primed to occur more frequently in Theme rather than Rheme, so in *The Prime Machine* it was not felt appropriate to risk masking this relationship by favouring Rheme-only occurrences. Another point worth making is that with very “dirty” corpora, filtering out “sentences” containing unusual symbols, multiple capitalization, and so forth would seem important, but in a sense this filtering is only hiding from the end user some of the inadequacies of the cleaning process or the suitability of the texts themselves. The cost of increasing corpora to sizes beyond which any human being could investigate every individual text and evaluate it for its suitability has been that web-derived corpora which are filtered in this way could contain many thousands of examples which are not deemed worthy of being displayed in concordance lines, but which nevertheless are contributing to the other results, and could be said to be “hidden behind the scenes”. For these reasons, the attempt to create a GDEX-like measure in *The Prime Machine* proceeded up to this point, but alternative means of sorting and filtering concordance lines were also sought.

Working with *The Bank of English* during the 1990’s, Collier developed a system to rank concordance lines (1994, 1999). His system was based on applying the lexical cohesion measures for text abridgement developed by Hoey (1991) to a set of concordance lines

rather than sentences from a single text. In Hoey's text abridgement system, two levels of relationship between each pair of sentences in a complete text are measured. The first level is called a "link", and is established through finding lexical items which are common to both sentences³⁶. If a threshold number of links between a pair of sentences is reached, both sentences in the pair are marked as forming a "bond". The number of bonds each sentence has is the second level of measurement. While acknowledging important differences between sentences from a single text and concordance lines from a whole corpus, Collier worked through the two level system of links and bonds which Hoey developed as a means of measuring the strength of lexical cohesion between two sentences, and with some modifications successfully applied this to concordance lines. His evaluation showed that different settings will yield different results, but overall the set of lines at the top of the ranking are not very dissimilar to those which would be selected by experts (Collier, 1999).

Collier's approach does not seem to have been implemented in any of the concordancers available today. However, it does provide a way of ranking lines without too many assumptions, and the parameters can allow high ranking for both colligation (through word-order and "function" words) and collocation (based on repeated forms or lemma). It follows that lines which are highly ranked using this system should help guide learners to "notice" patterns in the usage of a node and the method should work more towards provide usage-oriented rather than meaning-oriented examples. The sorting of concordance lines according to the words in the nearby environment on a very basic level is provided in most concordancers, and while sorting alphabetically by the words in specific slots (L1, R1, etc.) has been used to help users notice common patterns, a system like Collier's ought to provide a much more comprehensive ordering. Finding an alternative to alphabetical sorting for *The Prime Machine* was important because the nature of the overall software architecture would mean fairly arbitrary pagination would occur as concordance lines were retrieved. It was thought that implementing Collier's method would provide a good solution and with concordance line ranking for nodes with less than 1,000 hits bonds can be calculated in a second or two.

³⁶ In the text abridgement system, these links may be based not only on repeated use of specific types, but also other word forms from the same grammatical category, other types of "complex repetition", paraphrase and other relationships (see Hoey, 1991, pp. 51-75).

Within Collier's ranking system, there are a number of different parameters which can be altered to generate different results. There are four sets of parameters which he calls stopwords, positional specification, link threshold and span size. In the evaluation of his concordance ranking system, Collier found that two configurations of these appeared to be the most effective. For implementation into *The Prime Machine*, a decision had to be made as to whether to use a stop list and allow free positioning of items within the window, or not to use a stop list but be strict about the position. After developing scripts to calculate both of these, the second option was chosen for the following reasons. Firstly, the zero stop list performed well in Collier's evaluation when looking at usability of the most highly ranked results. He concludes that when humans rate concordance lines for usability over representativeness "the informants are making use of features which are more closely-defined positionally and heavier in grammatical items than those which occur in lines which are chosen as representative" (Collier, 1999, p. 207). As explained in the section below, a ranking score generated through this method is used in *The Prime Machine* in combination with other ranking scores. Since this ranking method counts grammatical items and compares items in fixed positions, it complements the other ranking methods very well. A further consideration was the processing load for the parameters. While including grammatical items increases the overall number of items which are selected, by far the heaviest processing requirement is the sorting and grouping of each item. With no stop list, the *SQL* command does not need to incorporate a join to the table which holds the lexicon and with the fixed position parameter being used, each lexical item can be given a unique value in the temporary table according to its slot. In this way, the values in the temporary tables are more disparate and the database seems to handle them more effectively.

As well as decisions about specific window sizes, positions and stop-lists, another important issue is how to determine a suitable setting for the number of links required. As Collier explains, while the method draws on Hoey's system for text abridgement there are several differences between concordance lines and sentences from a single text. By definition, a node occurs in each and every line of a concordance set, so as Collier argues it is logical to reduce the number of required links by one. Collier tried link thresholds between 1 and 6, and found that for many searches with the highest setting there were no results meeting the threshold at all. He also found that the higher the setting the more likely it was that the only concordance lines meeting threshold were identical sentences or those containing a fixed phrase.

In a sense the point of varying the threshold in both Hoey's text abridgement system and that of Collier's concordance line section is to find a value which balances the need to be fairly strict so as to increase the power of the filtering out (Hoey) or ranking (Collier), while at the same time being loose enough to ensure that a reasonable number of results are left. A low threshold for a text which has very strong lexical cohesion, or for a set of concordance lines containing a strong collocation or grammatical items, will lead to a very flat range of bond values. Similarly, a very high threshold will also lead to a large number of items having zero bonds, thereby also meaning that the range of bonding values is limited. One way to automatically adjust the threshold level would be to calculate the range of different bond values for several different threshold levels and then choose the one which has the widest variety of different values. Within the *SQL* procedure used to calculate bonds in *The Prime Machine*, once links between each of the concordance lines have been calculated, bonding values for 1 to 5 links are calculated and the range of values each of these generates evaluated. The value which is stored in the database is automatically chosen as the one with the widest variety of values. In this way the setting for the number of required links is automatically adjusted within this range to gain optimal discrimination for each set of lines.

For a research-oriented concordancer, Collier's system deserves further attention and an on-the-fly calculation of links and bonds ought to provide a fine-grained and flexible approach to concordance line selection. However, since speed of retrieval and simplicity of presentation were two major areas of importance in the development of *The Prime Machine*, the parameters are fixed and bonding scores are calculated as part of pre-processing. For small to medium sized corpora, results can be generated in less than 24 hours to provide "instant" results. However, the need to compare each concordance line with each of the other concordance lines to generate bonding scores leads to a problem of processing speed which increases in orders of magnitude. If fixed slots are used for 4 words either side of the node, the number of comparisons needed is 8 times the number of pair combinations for the node frequency.

Table 4.9: Magnitude of the growth of operations required for concordance line comparisons

$$\text{Combinations required} = 8 * \frac{n!}{k!((n-k)!)}$$

Node Frequency	Combinations	Comparisons for 8 word window
100	4,950	39,600
1,000	499,500	3,996,000
2,000	1,999,000	15,992,000
5,000	12,497,500	99,980,000
10,000	49,995,000	399,960,000
100,000	4,999,950,000	39,999,600,000

In *SQL*, a join operation with results grouped according to the lexical item is an efficient way to calculate subtotals of matching lexical items for each slot, but even with the high performing sorting and grouping optimizations built into *MySQL*, it is clear from Table 4.9 above that the number of operations increases to a point which would stretch a modern system. From experimentation, grouping the matches for each sentence together when the node frequency gets much beyond several thousand leads to slow disk-based filesorts rather than faster in-memory execution. However, Collier's system is very good for nodes which do not have clear collocation patterns based on statistical measures. *The Prime Machine* uses Collier's method in combination with another measure which draws on collocation data, so high frequency items do not have to be processed. Collier's method adds more fine grained ranking to medium frequency data and provides a means of ranking low frequency items for which little collocation information is available.

The application of Hoey's (1991) automatic abridgement method to concordance line selection is an ingenious way to create a ranking system for researchers or learners. However, as well as applying the method to concordance lines as Collier does, with a corpus stored in an efficient database it is fairly straight-forward to apply some of Hoey's automatic abridgement techniques to each text in the corpus and it seemed worth considering whether this could offer any useful way of filtering and ranking the results for language learners. While many of the goals of text abridgement differ from those of concordance line selection, there are a few interesting points of similarity. Firstly, the need for sentences in an abridgement to still be able to carry a clear message despite being presented in a much reduced context is similar to the need language learners may have for concordance lines to clearly demonstrate the meaning of a node word despite being

presented in the reduced context of concordance lines or cards. Issues surrounding the amount of context displayed in a concordance line have already been discussed in Chapter 2 (Section 2.1.5) and earlier in this chapter (Section 4.8.2), and will be discussed further in Chapter 5. Secondly, the need for those words which are more central to the overall meaning of a text to be retained in an abridgement in preference to those words which are more peripheral to the main topics of the text, is similar to the need language learners may have for concordance lines to clearly demonstrate the range of core meanings and contexts of a node word as opposed to more peripheral meanings and contexts. In the conclusion of his book introducing the text abridgement technique, Hoey (1991) argues that teachers should focus on vocabulary in texts which participate in bonds in order to ensure that they are relevant to the meaning of the texts which are studied and also to ensure that they are met in a variety of contexts. If this text abridgement method is applied to texts in a corpus in order to select a set of clear examples for language learners, Hoey's suggestion could be followed in the opposite direction. It would seem fair to suggest that concordance lines representing sentences which participate in bonds should be ranked more highly. Furthermore, it would seem fair to suggest that concordance lines in which the node word is one of the words creating a link should be selected in preference to sentences which are bonded but the links are not made through repetition of the node word.

The implementation of a concordance ranking system based on these text abridgement principles involved two steps. First, a procedure was needed to measure links and bonds for each sentence within each text. After that, a way of converting the bonding score into a concordance line ranking was needed.

The first of these two steps was accomplished through a stored procedure which is called as part of the corpus refactoring process. The procedure runs through each text in the corpus separately, and measures links based on matches using simple repetition of word type or stem. This process is a simplified text abridgement that can be performed automatically without manual editing of sentences or sophisticated text analysis techniques³⁷. The fact that the results are not to be used for text abridgement, but rather for the ordering of concordance lines means that the setting of a suitable threshold level

³⁷ Development of a database procedure to perform automatic text abridgement including all the complex repetition and expansion of elided items and referents is well beyond the scope of this project, and it is worth noting that for concordance line ranking editing the lines to expand intra-textual references or elided items would not be desirable since the aim would be to leave the corpus texts in the form in which they are found.

for bonds based on these links needs to be carefully considered. With text abridgement, the length of the abridgement can be reduced by increasing the threshold value. However, for concordance line selection, the aim is not to reduce the number of sentences in each text, but rather to prefer concordance lines which demonstrate strong lexical cohesion over others. Therefore, there is no requirement to reduce the number of bonding sentences for a text to a specific point and for *The Prime Machine*, the minimum number of links was set to 3.

The second step is designed to transform the bonding scores so they can be used for the purposes of ranking concordance lines for each node word represented in each sentence. The bonding score for text abridgement purposes is measured at the sentence level, but when it is converted into a concordance ranking score, each word in the sentence is given a bonding score for the number of bonds in which it contributes a link. A further transformation is needed to enable bonding scores for concordance lines from texts of different lengths to be compared against each other. Since the number of sentences in a text varies considerably it is necessary to balance the impact of long texts (which have much more opportunity for bonding to occur) against the fact that extracts from shorter texts might be easier to read in isolation. To achieve this, each token is given a concordance ranking score based on the bonding score as a proportion of the sentences in its text. Through these steps, when this ranking method is selected by the user, each concordance line in the corpus can be instantly ranked according to the bonding score for each specific node word.

In addition to the ranking methods based on penalties for various features, and those based on comparisons between concordance lines or sentences within each text, another way of ranking and selecting concordance lines would be to weight lines according to the strength and/or number of collocations which they contain. As mentioned in Table 4.8, the GDEX algorithm implemented in *The Sketch Engine* includes a score for collocates. Following a longitudinal study of one advanced learner, Li and Schmitt (2009) suggest that competence and confidence with collocations develop not so much as a result of acquiring new phrases, but mastering those previously learned. Durrant and Schmitt (2010) argue adult learners' retention of collocations should be improved through repeated exposure in different contexts. Showing concordance lines which hold examples of strong collocations should help provide input for learners, and several ways of operationalizing this would seem to be possible. Lines could be ranked according to the raw number of collocations

they contain. Alternatively, scores could be to give each line according to the total frequencies of all the collocations represented³⁸. A third alternative could be to make use of a measurement of the statistical strength of the collocations.

Full evaluations of each possible ranking method are beyond the scope of this present project. The aim for concordance line ranking in *The Prime Machine* is to help language learners see typical sentences and typical word ordering. Therefore, it was thought desirable to set up the ranking system in such a way that the Collocation Tab and the concordance lines would be mutually supportive. For the collocation clouds, the square root of the log-likelihood score was used to determine the size of each item. The log-likelihood statistic is already influenced by the frequency of both the node and collocate as well as the total corpus size. For a cloud, the difference in size between items needs to be not too great, otherwise other items in a cloud containing a very strong item would be too small to see. For concordance line ranking, if up to 8 slots are available for each item, using raw frequency or the raw log-likelihood value would rank a very high frequency or very strong item too highly in comparison with the others and the entire page of concordance lines could be filled with just one collocation. By using the square root of the log-likelihood, the differences between values are compressed. In purely statistical terms, a fuller evaluation and exploration of alternative ways of obtaining this kind of measure would be needed. However, as a way of ranking results so that collocations strong on the Collocations Tab can be seen in the top results, this pragmatic approach seems a reasonable way to overcome the speed issues evident with Collier's method for medium to high frequency items in larger corpora.

³⁸ The frequency of each collocation is equal to the number of concordance lines in which the combination occurs, so summing the frequencies of the collocations contained in a concordance line promotes those which contain the most frequent collocations. If a position sensitive collocation measure were used, this would, in some ways, be similar to the scoring method proposed by Collier (1999), with the highest scoring lines containing several words which occur in a similar position in relation to the node in many of the concordance lines. However, there would be two clear differences. Firstly, since some concordance lines would contain more than one collocate for a specific node it would mean that there would be some overlap in the bonds made between concordance lines. Checking each occurrence of each collocation to ensure no overlaps between the bonds established through this scoring method would lead to similar computational challenges to those discussed regarding Table 4.9. Secondly, not all items would be counted since those with very high or very low frequency may not be stored in the database's collocation tables.

Since 2_mwu collocations with gaps are based on three possible slots (L4, L3, L2 or R2, R3, R4), the square-rooted LL scores for collocations with no gaps are first multiplied by 3 and then the overall result is divided by 3. However, the limitations which are imposed on the calculation and storage of collocations explained in Section 4.7, also have implications for the ranking system. At one extreme, very high frequency words are not processed for collocations at all, so the values for these will always be 0. It is hoped that learners will not want to look up these words as they are usually function words, but preventing them from doing this does not seem fair, and it is desirable that a similar ranking method would be used for all items. While collocations are not stored for these high frequency items, they can occur as collocates of other nodes. The collocation ranking for these words, then, follows the same kind of process, but rather than using the lexical item itself to pull out a list of collocations for the node, the search is done on the collocate column in the database. These very high frequency items are ranked by the strength of collocation relationships in which they are the collocate.

Another issue is that some words will have very few collocations or none at all and therefore the rankings will be very flat. This causes problems in terms of what becomes an arbitrary cut-off point since as with any ranking system if there are items with joint rankings selecting what happen to be the top few from the list can be misleading. It can also make the indexes on the database inefficient as the entire set of results might need to be extracted. The more finely grained the ranking system, the more efficient the index can be.

By combining the results of Collier's system and the collocation rankings, however, users can get the benefits of both systems. Collier's system is applied to lexical items with a frequency equal to or lower than 1000. Items with a frequency over 1000 will therefore have a Collier ranking of 0, but they are likely to have many collocations and therefore values for the other measure are more widely dispersed. A similar ranking is obtained for MI collocations, using MI3. This too is offered as a combined ranking method and an index is created sorted by MI3 concordance line ranking then by the Collier-like score. Figure 4.18, Figure 4.19, Figure 4.20, Figure 4.21 and Figure 4.22 show some examples of concordance lines sorted using these ranking methods.

	Text to the left of node	Node	Text to the right of node
1	placed by verbal contact via the telephone and the	outcome	can be satisfactory for the rural resident, although
2	s threshold in effect. The data suggest that optimal	outcome	for the fetus can be achieved only when maternal
3	... That was rejected. But in the	outcome	the jury returned a verdict of not guilty on that cou
4	not happen and, even more interesting, in some the	outcome	is what has been called 'outstanding'; that is to say
5	for him to imagine anything other than a successful	outcome	to his diplomatic and military operations. He did not
6	the concerns about neurological progress and fetal	outcome	, phenylketonuria is a potential candidate for gene
7	ility and community. This change was essentially the	outcome	of concern for prevention in the child care field./ I
8	... (3) The third doubtfully determined	outcome	was from a builder who had been in financial difficu
9	he quality of life of older people. However, such an	outcome	will not be achieved without active pursuit by pract
10	ills whose origins are usually far more complex. This	outcome	of family power politics has to be avoided. The cou
11	ions and did not immediately become aware of their	outcome	. On 22 April 1988 the landlord, through its solicitor:
12	Britain's general practice community — the ultimate	outcome	of the experiment remains to be seen.// General
13	her novels, as in Crime and Punishment , the actual	outcome	of such promptings makes an interesting study./ C
14	il-alveolar oxygen tension ratio predicts respiratory	outcome	better than the minimum ratio. The former may refl
15	the structure of care rather than on its process or	outcome	. Family health services authorities therefore have
16	age, into which the class cleavage was placed. The	outcome	was variations between states in the relative impor
17	agreed upon an organized return to classes./ One	outcome	of this initiative was the formation of the National E
18	s in which lawyer B did not adopt the client's chosen	outcome	as his objective, numbers one to four involve an at

Figure 4.18: Concordance lines for *outcome* in the *BNC: Academic* sub-corpus, ranked by fixed random order

	Text to the left of node	Node	Text to the right of node
1	an Walters. Not until two years had passed was the	outcome	of these disputes at all clear./ Geoffrey Howe's fir
2	:fore the government is not fully responsible for the	outcome	. Control of money supply (M3) has been erratic, an
3	work activities and in rudimentary measures of client	outcome	. The dimensions of further work which would fill out
4	n referred by GPs to avoid compulsory admission./	Outcome	is shown in more detail in Table 4. This shows a not
5	: other reasons, to sum them up and to reflect their	outcome	. For ease of reference I shall call both reasons of t
6	ful neighbours, leaving some 25 or so victors./ The	outcome	of this process was in no way predetermined. No Fi
7	who directed the process had no sense of the final	outcome	. They were driven by domestic challenge and exter
8	ion should never be overlooked when analysing the	outcome	of local elections.// Discussion ...
9	nditional) cue to become associated with a different	outcome	. The different expectancies generated by these cu
10	much wider than anything envisaged by Philips. The	outcome	is a power which transforms the nature of the relat
11	ent causes can add together to produce a particular	outcome	, a process known as multiple causality. Given this, v
12	mpaired, the incidence of flooding is increasing. The	outcome	is thus similar a that of Jamaica. The social factors ;
13	ical politics over recent years is not just the passive	outcome	of national forces working themselves out at the lo
14	any information available concerning the long-term	outcome	of patients who cut themselves. In one study of a :
15	le to make each objective refer to only one learning	outcome	. Example (3) falls down in this respect in that it incl
16	ills whose origins are usually far more complex. This	outcome	of family power politics has to be avoided. The cou
17	ps by clarifying and expanding on what is said. The	outcome	may be to highlight the part played by more power:
18	is they relate specifically to manpower policies. One	outcome	of this process has been the emergence of older pe

Figure 4.19: Concordance lines for *outcome* in the *BNC: Academic* sub-corpus, ranked by the GDEX-like method

	Text to the left of node	Node	Text to the right of node
1	e had seven relatives with Crohn's disease.// Main	outcome	measures —/ Patterns of segregation of either dise
2	tinine concentration below the index value.// Main	outcome	measures and results —/ 125 adults (140 per million
3	icular reference to those aged 0–24 years.// Main	outcome	measures/ Numbers of cases and incidence particul
4	e days, random group 10 days). Differences in final	outcome	measures such as duration of supplemental oxygen
5	quirements. The effect of treatment group on final	outcome	measures appears to be mediated largely through i
6	; in which the values at the last visit of three pivotal	outcome	measures — that is, urinary albumin and sodium ex
7	ition of expected health gain, treatment goals, and	outcome	measures for drug treatment would reduce unnece
8	al hernia repair under general anaesthesia.// Main	outcome	measure—/ Change in anxiety level observed after
9	Research Note: Developing	Outcome	Measures in Child Care HARRIET WARD and SONIA
10	ncern above dictates the inclusion of certain client	outcome	measures across the organizational types, but are
11	ncies? More specifically, what should be the specific	outcome	measures that are selected and to what extent sho
12	ange of interventions, which should look at broader	outcome	measures than just hearing loss. They base their re
13	n the random group was held below the introitus./	Outcome	measures recorded included Apgar scores, initial pa
14	; introduced because this influenced our respiratory	outcome	measures./ Statistical methods — Sample sizes of
15	it group was the most important determinant of the	outcome	measures considered; these included three 'first da
16	outcomes/ It is important to develop	outcome	measures as tools for quality assessment. A signific
17	y similar, using similar psychometric assessments as	outcome	measures./ In the research carried out by one of
18	e believe that a multicentre trial with clearly defined	outcome	measures is necessary to recruit an adequate numt

Figure 4.20: Concordance lines for *outcome* in the *BNC: Academic* sub-corpus, ranked by Log-Likelihood Collocations and then by the Collier-like method

	Text to the left of node	Node	Text to the right of node
1	1936 is best understood in 'internalist' terms, as the	outcome	of a process of intellectual discovery./ How convin
2	n revealed. Quite apart from the importance of the	outcome	of the case to B himself, if it is even arguably inco
3	ities vary across firms and that these influence the	outcome	of the struggle between capital and labour over th
4	this standard must have no personal interest in the	outcome	of the case, but which may also be relevant in the
5	een told.' The parents believed, however, that the	outcome	of the assessment would be a placement in a speci
6	both patients and management for the process and	outcome	of the care they provide. Sixthly, general practitor
7	especially where the witness has no interest in the	outcome	of the case, the lost opportunity to assess demear
8	States have interests that may be affected by the	outcome	of the proceedings, and that the present process c
9	m conveniens and (b) by having regard to the likely	outcome	of the hearing in that court contrary to the principl
10	a local authority, necessarily has an interest in the	outcome	of a decision. In such situations its decision should
11	'an assisted party will, inevitably, depend upon the	outcome	of the case. As has been seen, the assisted party
12	etween professional and users or carers, how is the	outcome	of an assessment to be decided? How can social wr
13	form of society are conceived and explained as the	outcome	of the way a particular structure — the capitalist m
14	: are intended to inform a critical assessment of the	outcome	of policy in the shape of service provision for alcoh
15	a serious disagreement about the nature and likely	outcome	of the process. The disagreement revolves around
16	ndling clients in this field. This can be crucial to the	outcome	of the case and the satisfaction of the client./ Sor
17	: the initial range of options and may also affect the	outcome	of the policy chosen. In this section, we examine h
18	cope with any resulting distress, then the eventual	outcome	can be, more often than not, extremely valuable./

Figure 4.21: Concordance lines for *outcome* in the *BNC: Academic* sub-corpus, ranked by MI Collocations and then by the Collier-like method

	Text to the left of node	Node	Text to the right of node
1	egal discourse was such that it would lead to a legal	outcome	which would translate back directly into the outcom
2	utcome which would translate back directly into the	outcome	chosen by the client as formulated in his or her own
3	s in which lawyer B did not adopt the client's chosen	outcome	as his objective, numbers one to four involve an at
4	an attempted transformation of the client's chosen	outcome	, and number five is a refusal to translate . Contrac
5	lawyer's attempt to achieve another client's chosen	outcome	./ One of the two clients of lawyer A classified as c
6	nvolved an attempt to transform the client's chosen	outcome	by persuading the client to accept a 'reasonable' ob
7	... The client was content. However, the	outcome	had been chosen by the broker land-agent, who re
8	cases in which lawyer C rejected the client's chosen	outcome	are listed below. (1) ...
9	rrent work to the firm. In all other cases the chosen	outcome	of business clients was adopted by the lawyer. This
10	: deviant cases — those in which the client's chosen	outcome	was rejected — and the doubtful cases are examin
11	tcomes./ The six cases in which the client's chosen	outcome	was not adopted were: (1) ...
12	re a translation was necessary if the client's chosen	outcome	was to be achieved. In each case lawyer C refused
13	f the correlated group depends on the fact that the	outcome	of a correct response is reliably different for the tw
14	ome, X, and target stimulus B along with a different	outcome	, Y (Fig. 5.10). Mediation theory requires that X and
15	bjects experience target stimulus A, along with one	outcome	, X, and target stimulus B along with a different out
16	and the third, C, being associated with a different	outcome	. Subsequent tests show discrimination between A and
17	: objective in legal discourse into the client's chosen	outcome	. Clients do know and are entitled to know what the
18	may themselves be useful predictors of subsequent	outcome	— that is, median and minimum arterial-alveolar ox

Figure 4.22: Concordance lines for *outcome* in the *BNC: Academic* sub-corpus, ranked by text abridgement-based method

Summary

This chapter has introduced a new method for extracting collocations and several new ways in which collocation data can be used. Although these multi-word units of 2 to 5 words in length are stored separately from the lexicon table, in support of the aim of showing differences between nested items, they are also used as the basis for various other analyses. The use of log-likelihood contingency tables as a way of finding relevant information to present to a language learner is not limited to collocations in the software. Similar tables are generated to pre-calculate relationships between textual position, grammatical features and repetition. They are also used for key words, the analysis of metadata and key associates. The following two chapters introduce these other features, and data are held for individual lexical items and all of the statistically significant collocations which have been extracted as part of the collocation extraction process.

Chapter 5: Further features of Lexical Priming

In the previous chapter, the phenomenon of collocation was examined and the way collocations are calculated, stored, retrieved and displayed in *The Prime Machine* was introduced. Collocation is an important feature of Hoey's theory of Lexical Priming, and it will now also be clear that in *The Prime Machine* collocation measures are used in a number of ways, both in the display of concordance line data and in support mechanisms. The prominence of collocation in language learning and teaching materials and its potential for helping learners explore patterns beyond the single word led to the decision to devote to it one of the tabs on the main screen of the software. However, a number of measures related to other features of Lexical Priming are gathered together on another tab, and information on this tab is provided for both single words and collocations. The purpose of this chapter is to introduce some of the other ways in which the theory of Lexical Priming has driven the design of the software and to explain how the results provided to users on the Graphs Tab are derived.

5.1 Measuring tendencies for text position

The tendency for words or phrases to occur in different positions in a text is an interesting and under-researched area, and one which is somewhat difficult to explore using standard concordancing software. Hoey (2005) introduces evidence for the existence of such tendencies and calls them "textual colligation" as part of his theory of Lexical Priming. Tendencies of lexical items and the nested combinations of these items to occupy different positions are reported in the form of proportions and percentages. Some work has been done looking at some of the possible different text units and the tendency for words and longer phrases to occupy positions at the beginning of these. For example, at the text and paragraph level, Hoey and O'Donnell (2008) and O'Donnell et al. (2012) compared the first sentences of texts and paragraphs against the sentences from the remainder of these texts in order to establish which words had a tendency to be used in text initial and paragraph initial position. Their procedure was complicated, especially for the generation of concgrams, and involved splitting the corpus into sub-corpora according to each of the required set of positions, using concgrams in *Wordsmith Tools* and then a *Python* script before running the wordlist function in *Wordsmith Tools* again.






The use of the key word method to identify words which occur with statistical significance in text initial or paragraph initial position seems very promising. However, concordancing

software provides little integration of functions to explore such features and few language learners would be skilled or motivated enough to go through the process of splitting a corpus themselves and then performing key word analysis and interpreting the results. Garretson's *CenDiPede* software (2010) includes three features under the heading "Pseudo-Colligation", two of which are relevant to textual position. The first uses the results from clausal analysis to report the raw frequency of occurrences of the node occurring before the verb within its clause. This is designed to be a rough mapping to Theme-Rheme. The second is described as a "nod to Hoey's notion of *textual colligation*" (p149), and is the percentage of instances of the node where it is sentence initial. Although measurements of an item's occurrence in specific positions such as the beginning or end of a paragraph do not seem to be well supported in concordancing software, other features of concordancing software which are related to the distribution of words in texts are briefly considered in the next chapter where the use of structural information from corpus documents is introduced.

The results from the study by O'Donnell et al. (2012) which found that one in forty individual words showed a tendency to be used in specific positions provides good evidence that this is something worth researching further, but it does also suggest that if the starting point is a word or phrase and the aim to is to discover whether or not this word or phrase has such a tendency, the overwhelming majority of cases are likely to be disappointingly negative. Writing about concordancing software in more general terms, Cobb (1999) argues that language learners need software which does not assume detailed linguistic knowledge and which also does not assume that the users will be curious enough to explore. It would seem obvious that for phenomena like textual colligation which are less well-understood by both teachers and students, these two aspects of software design are even more important.

Therefore, in order to provide more information about the tendencies of words and nested combinations to occur in various environments, procedures were developed for *The Prime Machine* to calculate and store these tendencies and display them. The key word approach is applied to a range of different kinds of feature in *The Prime Machine* and these have been put into 5 groups. Table 5.1 below shows the complete list and groupings of features which are currently implemented, as well as the dependencies each one has on pre-processing and other mark-up processes.

Table 5.1: Features of Lexical Priming on the Graphs Tab

Group	Feature	Values	Level	XML / other encoding	CLAWS tags
Headings 	Title	Title; Not a title	Sentence	✓	
	Heading	Heading; Not a heading	Sentence	✓	
Position in text ³⁹ 	Sentence position in text	Text Initial; Text Ending; Not text initial or text ending	Sentence	✓	
	Paragraph position in text	First Paragraph; Last Paragraph; Not first or last paragraph	Sentence	✓	
	Sentence position in paragraph	First Sentence; Last Sentence; Not first or last sentence	Sentence	✓	
	Word position in sentence	First Fifth; First Third; Last Third; Last Fifth; Not first or last third	Word	✓	✓
	Word position in sentence	Theme; Rheme; (unknown)	Word	✓	
CMVYN group 	Complexity	Simple Sentence Complex Sentence	Sentence		✓
	Modality	Volition/prediction; Permission/possibility/ability; Obligation/necessity; No modals	Sentence & Word		✓
	Voice	Active Voice/Other; Passive Voice	Sentence & Word		✓
	Polarity	Positive; Negative	Sentence & Word		✓
Det. & Prep. group 	Determiners	Definite articles / Possessives; Indefinite articles; No articles	Word		✓
	Prepositions	Near Prepositions; Not Near Prepositions	Word		✓
Repetition 	Repetition	Same form Same stem Not repeated	(Not stored)		

³⁹ Not all the values for features in this group are mutually exclusive. For example, words which are in the first fifth of a sentence will also be in the first third. However, paragraphs of one sentence in length and texts of one paragraph or one sentence in length are not included in the calculations for certain tendencies. This is explained in more detail in Section 5.1.3 below.

Although, *CLAWS* tags are identified in the table for only some of the features, it should be noted that for most corpora all of the processes rely on *CLAWS* for sentence segmentation as measurements work within sentence boundaries.

Since the features related to headings and position in text draw on the full range of these processing dependencies and also have a direct influence on the presentation of concordance lines, they will be introduced first. After explaining how these features are shown in concordance lines, the means by which tendencies are measured and stored will be introduced. The way in which these are introduced to the user on the Graphs Tab will then be explained, and some specific examples from corpora will be given. For the remainder of the chapter, each of the other elements will be introduced in the same way, and then the ways in which the user can interact with the data will finally be described.

5.1.1 Headings and Position in Text

Hoey's claim regarding textual colligation is:

Words (or nested combinations) may be primed to occur (or to avoid occurring) in the beginning or end of independently recognised discourse units, e.g. the sentence, the paragraph, the speech turn.

(Hoey, 2005, p. 115)

For a corpus text such as a book, this could mean phrase level, sentence level, paragraph level, chapter level or text level, but it is generally the case that corpora do not contain whole books (mainly due to copyright restrictions). With shorter text types such as essays, academic journal articles, pamphlets, and newspaper articles, however, corpora often contain complete texts. Following the method introduced here, it would be possible to adapt the software to extend to measuring the tendency to occur in other units, but *The Prime Machine* currently implements measures at the sentence level (raw position in words and Theme-Rheme), at the paragraph level and at text level. It is worth noting that due to the limitations imposed by the structure and design of corpora, the meaning of "text initial" and "text ending" could vary considerably within and across the corpora used in the system. For example, the *British National Corpus* uses a sampling technique which includes many extracts rather than full texts, and for more than 40% of the written texts, the sampling type in terms of this is not provided (Burnard, 2007a). This has a direct implication for the generation of text-initial or text-ending statistics as the beginning and end of the file or sub-document frequently is not actually the beginning or end of a complete text.

Garretson (2010) avoided doing text-level collocations because of speed considerations and

because he used the *BNC* and so many of the texts were extracts. However, for *The Prime Machine*, since position in text information would be simple to extract from corpora made up of complete academic texts or newspaper articles, it was decided it would be worthwhile to explore the *BNC* sampling details further and to consider what impact this could have on processes and results for these. On the one hand, the *BNC* manual states that where possible, full chapters were used and where a cut-off point was needed due to the word limit restrictions this was made at the end of a recognizable unit such as the end of a section or chapter. Also, it would seem that adding the written text figures for sentence units together for complete text samples and for those which were from the beginning of a complete text, the proportion of sentences covered would be more than 30%. On the other hand, if the beginning of some of the speech transcriptions are inspected with the human eye it is clear that many of them begin part-way through a meeting, or that the speaker had already said something which was not transcribed. There are, of course, particular difficulties dividing stretches of transcribed speech into separate texts (Biber & Conrad, 2009). One way to approach the counting of instances of different positions in text for items in the *BNC* could have been to try to extract the sample method from the header in the *BNC* and then only set text initial or text ending if the sampling method corresponded. However, information encoded in the XML files seemed to be quite limited. In the end, it was decided that the same procedure would be followed as for the other corpora, and the beginning of and end of texts would be marked as if the extracts really were the beginning and end of complete texts. While this has some important limitations for research on the *BNC* corpus, none of the other corpora used in the system so far have this limitation. Since the positions being measured account for a small minority of all the positions available in a text, it should also be clear that treating a text which has been sampled from the end of a text as if it were text initial will in actual fact be much more likely to dilute the results rather than raise the significance of an item which should not really be counted as being text initial. In addition, as will be evident from the introduction below, the user is encouraged to explore the concordance line data and to evaluate the reported tendencies, rather than to only base his or her conclusions on the raw figures presented.

5.1.2 Making Position in Text more prominent in concordance lines

For all the advantages of KWIC (some of which were mentioned in Section 4.8.2 of Chapter 4), by showing the node word in the centre of the screen with one horizontal line for each concordance line, not only are paragraph breaks usually masked, but the position of the

node in the sentence is not very prominently displayed either. Even if the KWIC window is limited to words occurring in the same sentence, white space to the left of a sentence initial instance gives some indication that the word occurs towards the beginning of a sentence, but then masks whether or not this is a paragraph break. Concordance lines in which the node word is more than 4 or 5 words away from the start of a sentence appear much the same whether or not they are towards the beginning of a long sentence, part of a singleton paragraph, or towards the middle of an average length sentence.

Both *WordSmith Tools* and *AntConc* provide access to the original corpus texts, so it would be possible to click through to the source document in each instance and look at where the concordance line occurs. In *The Sketch Engine*, two ways of viewing concordance lines are provided: either KWIC or complete sentences. There is also provision to extend the window so that more context for a concordance line can be seen. XML tags indicating paragraph breaks are visible, but these do not change the way the text is formatted on screen.

If users are familiar with the mark-up and tags used in the texts in the corpus, it would be possible to formulate procedures whereby aspects of each occurrence of a search term could be limited according to paragraph breaks, heading markers and so forth. *WordSmith Tools* provides a number of ways to facilitate this functionality. Corpora can be split into two or more sub-corpora through the text processing tools, allowing users to use rules to determine the beginning and end of texts as they are processed before being included in a sub-corpus. Alternatively, more complicated queries can be entered for the search, making use of tags and other information. In *AntConc*, regular expressions (regex) are supported, meaning that users could formulate or copy and paste regex expressions from elsewhere to search for combinations of one item followed by another, or for instances within specific environments.

One of the main differences in the presentation of concordance lines in *The Prime Machine* is the Cards Tab and the card for the currently selected KWIC line. For several years, the possibility of presenting more context to learners in a concordancer had been part of a vision I had had for helping learners become more confident and more familiar with corpus data. In the literature, there have been many reports of students finding the KWIC display difficult, at least at first. While some writers have played down the importance of this, and others have suggested it could be a benefit, since my concordancer was being built from scratch, it seemed sensible to try to find an alternative way to display the information. At

the time when KWIC was developed, computer screens (and printers) had a limited number of characters which could be produced on each line and using computers was essentially an experience of looking at text on screen in a form which was very different from other reading materials like magazines, etc. However, in recent years, the development of the internet and mobile technologies has meant that younger generations are regularly interacting with lists of data in different ways. Rather than scanning down a screenful of cramped data, modern apps and applications present the user with a list box where it is understood that most of the data is off screen “somewhere” and the user will scroll through the pages.

At the time the initial proposal for this project was developed, a prototype of a concordancer using on-the-fly text processing was developed to show how extending concordance line displays to show full paragraphs might be achieved. As can be seen in Figure 5.1, the design at that time was very basic, using visual components from the *Delphi* programming language which had been available since the mid 1990’s. Although the first concordance in this screen-shot includes some formatting problems, it shows that the idea was to try to present concordance data in the form of concordance boxes, and to allow two different nodes to be compared side by side.

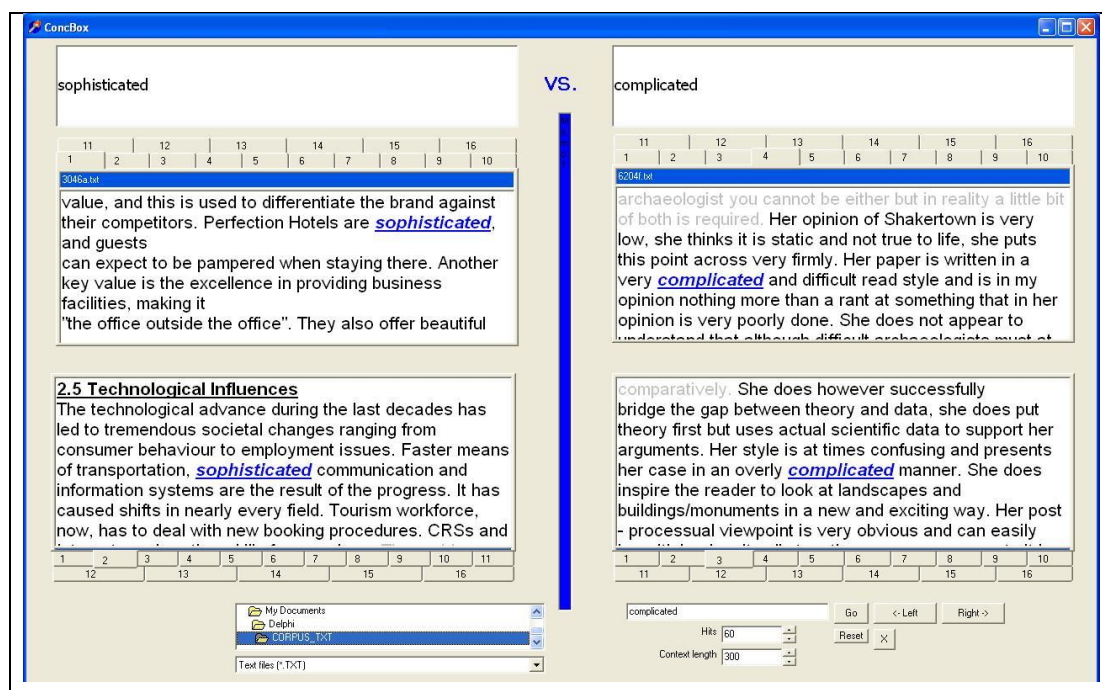


Figure 5.1: Early design for the project, showing simple visual components with incidental data from BAWE⁴⁰.

⁴⁰ The incidental data in this visual come from the *British Academic Written English corpus* (BAWE), which was developed at the Universities of Warwick, Reading and Oxford Brookes under the directorship of Hilary Nesi and Sheena Gardner (formerly of the Centre for Applied Linguistics

As described in Chapter 3, after beginning the project, I upgraded to *Delphi 2010*, and this led to both an appreciation of the developments in visual elements available and also the way in which data storage had developed in partnership with visual components. At first, attempts were made to draw each concordance box and provide graphical elements around these. The design was influenced by some of the other graphical patterns popular at the time, particularly the headline and summary combination used on the *Guardian* website. The *Guardian's* own report on the new layout of the front page which was launched in 2007 and still influenced the design in 2010, noted:

And there is an opportunity for more serendipitous browsing in the vertical stack of picture-led links, which showcases a miscellany of our favourite things on the site.

(Porter, 2007)

While some success was had with these when looking at small static datasets, the processing speed and manual calculation for positioning of boxes on screen using component library routines rather than lower level direct painting instructions seemed to be far from ideal, and alternatives were explored. The *TMS Component Studio for Delphi* (TMS_Software, 2011) contained many different visual components, including a modern looking *TAdvSmoothTileList*. Although this tiles component provided some support for HTML mark-up, meaning that if this was used for concordance boxes, node words could be made bold for example, there was limited support for other features such as highlighting the background of the text and an important limitation was that the spacing between words seemed to be inconsistent if not all of them were highlighted. A future upgrade to this component may address some of these short-comings. However, at the time of development, from a performance point of view, multiple processing of tiles for HTML encoding each time they were displayed also seemed to be quite slow, particularly if the number of concordance lines was increased to more than 40.

Another component from the same company which supports text-wrapping and styles and shading is *TAdvCardList*. This seems to be modelled around presenting data as a list of fields which might be displayed as they would have been on a card filing system. Each field can be set to have a different font and alignment and can be drawn with a minimum,

[previously called CELTE], Warwick), Paul Thompson (Department of Applied Linguistics, Reading) and Paul Wickens (Westminster Institute of Education, Oxford Brookes), with funding from the ESRC (RES-000-23-0800).

maximum or automatically determined height. It also supports invisible fields, which became useful as a means of storing some of the additional contextual information for each concordance line, as well as for filtering results. This was the component which was adopted in *The Prime Machine*, and customized *OnPaint* procedures were used to add panels of bullets and buttons for navigation to the top and bottom of these, aiming to match some of the modern look and feel of the *TAdvSmoothTileList* component. Figure 5.2 shows screen-shots for these two.

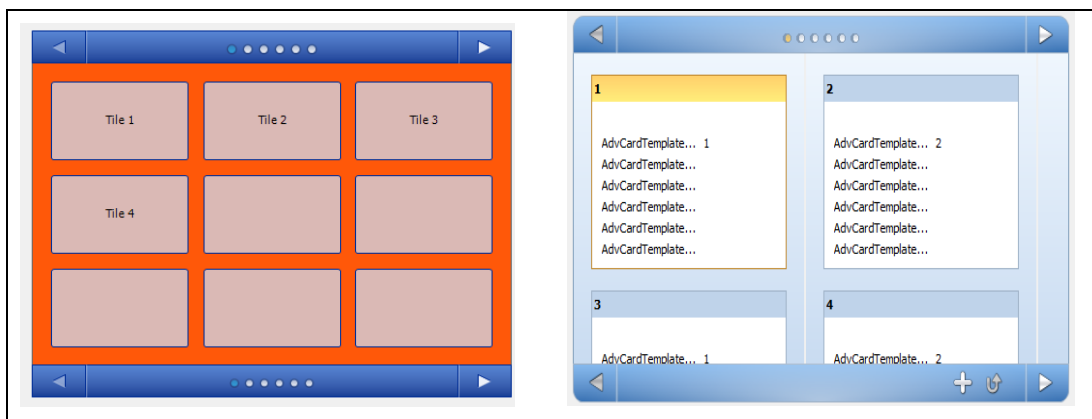


Figure 5.2: Designs for the Cards Tab (right), based on the navigation bars which appear as default on the *TAdvSmoothTileList* component (left).

When the card is prepared following the retrieval of concordance lines from the server or the local cache, different fields on the card are populated, according to whether or not each sentence was part of a heading. Figure 5.3 shows the card template as shown in the preview of the component in the software development environment, and the full list of data fields which are stored for each card. This template will accommodate a fairly wide range of configurations including cards where all three sentences appear as one field, and others where paragraph breaks and headings can be seen before or after the node sentence. The beginning or end of a text is indicated by a blank line at the top or bottom of the card.

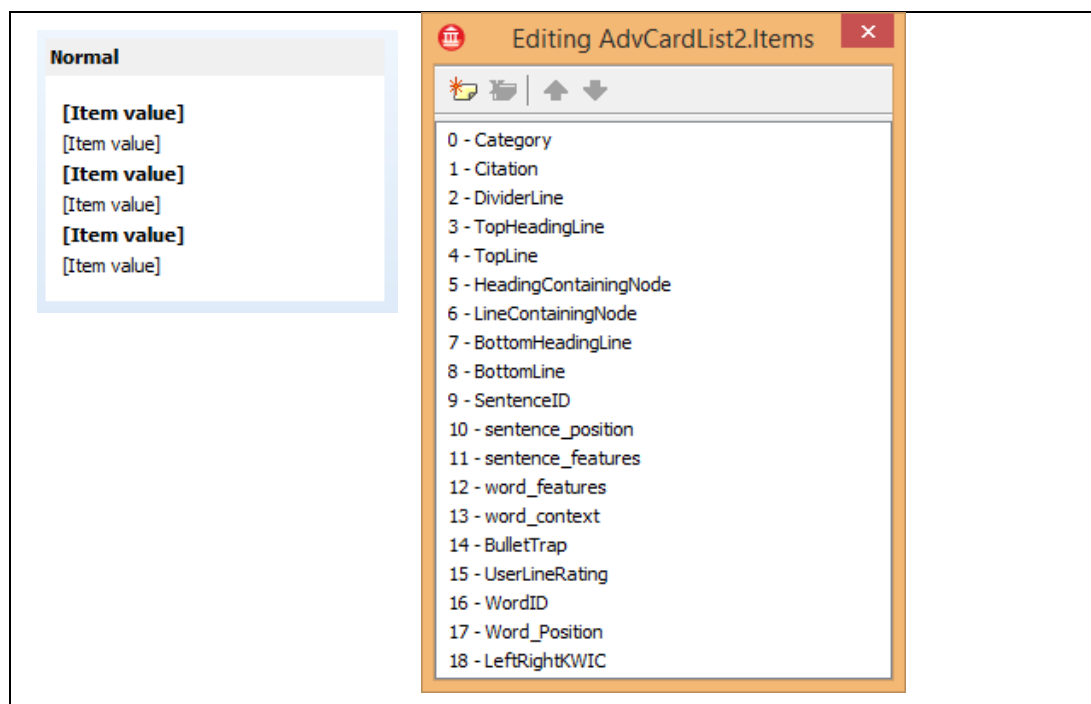


Figure 5.3 The card template for the Cards Tab (left) and the full set of data fields used (right).

It is worth noting that the card view is intended to be a compromise between the desire to provide additional information about headings and paragraphing and trying to reduce the complexity of both displaying text as it would be shown in the original sources and of reproducing it from the rows in the database. It was not thought desirable to try to implement full HTML support and be able to show text in exactly the same way as it would appear on a website or on the page of a book. Rather, the card view is a simplification bringing some order and uniformity to aspects like font size, colour and highlighting, while providing some visual information about the position of words in sentences and sentences in paragraphs. For example, the way in which headings are highlighted in text files varies considerably, yet during the refactoring process tags indicating underlining, bold, italics and other typeface information is discarded, and only the information about whether it was a heading or standard paragraph is retained. Therefore, if a corpus such as BAWE (BAWE, 2007) was imported where the text formatting of headings includes simple bold or simple underlining as well as almost every combination of bold, italics and underlining possible, all the headings would actually appear in the same way when viewed in the concordancer.

One issue regarding cards is that because sentences vary considerably in length, the number of lines of text inside a card needed to hold one sentence before, the sentence containing the node and one sentence after also varies considerably. Rather than having white space before and after any shorter blocks, the height of each card is not fixed, so

Chapter 5: Further features of Lexical Priming

cards with fewer lines occupy less space. As well as maximizing the number of cards visible at one time within these constraints, another reason for doing this was to make it clear at the top and bottom of each card whether or not it was part of a continuous paragraph. A screenshot of the Cards Tab showing the paragraph layout and different heights of cards can be seen in Figure 5.4.

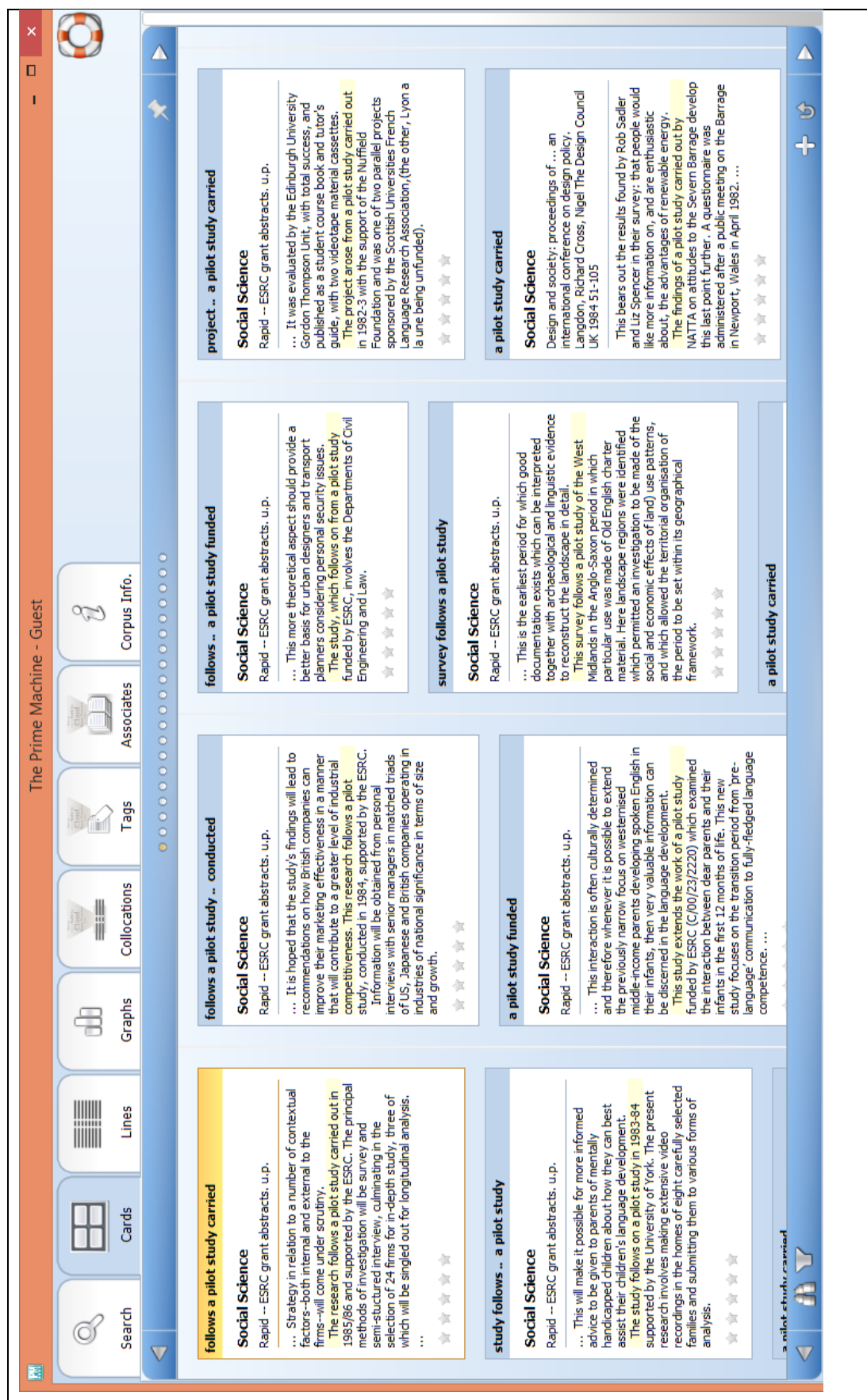


Figure 5.4: Cards of different heights on the Cards Tab with captions at the top and gentle yellow highlighting of the line containing the node; with incidental data from the *BNC: Academic* sub-corpus for a search on the node *pilot*. The currently selected card is shown with a yellow caption and border.

Chapter 5: Further features of Lexical Priming

As can be seen, complete sentences are shown on the card above and below the sentence containing the node, and the length of these can also vary considerably, meaning that on some cards the node may appear towards the top, while on others it could be towards the bottom. One further disadvantage of the card design is that since in free-flowing text the node can appear at any position horizontally (that is to say the word-wrapping system is not influenced by the position of the node), scanning to find the node on a set of cards can be quite difficult. To overcome this, as well as the caption at the top of each card which was explained in the previous chapter, two options are available for gentle highlighting of the line of text on the card which contains the node. The default option is to highlight the line of text with gentle yellow colour, but it is also possible to select dotted lines or no highlighting. The effect of three options can be seen in Figure 5.5.

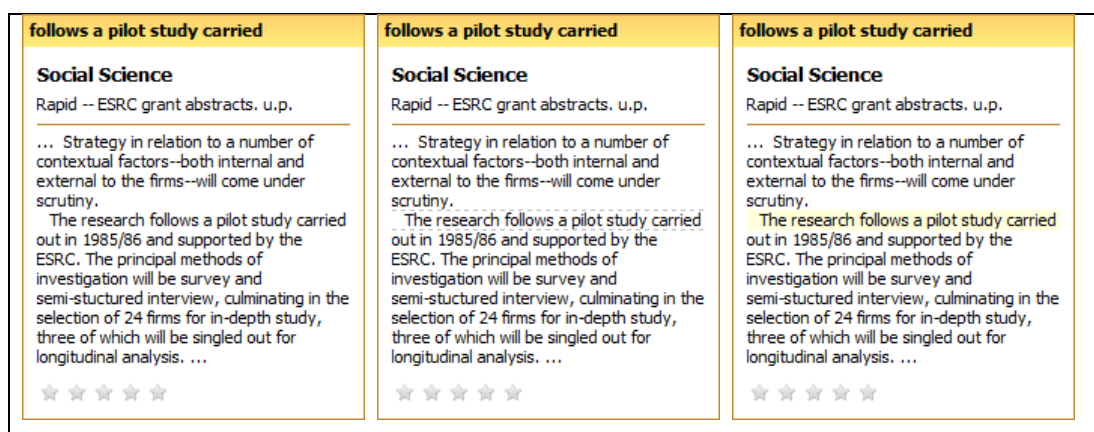


Figure 5.5: The same card from the Cards Tab, with no highlighting (left), dotted line highlighting (centre) and gentle yellow highlighting (right) of the node word, showing one concordance box from the *BNC: Academic* sub-corpus for *pilot*.

In order to accommodate this, a small enhancement was made to the drawing procedure for this component. The highlighting is also a way of tempting users towards looking in more detail at just the words near the node, in some ways helping to familiarize them with some of the benefits of a KWIC view.

The concordance “card” or “box” design provides several benefits. Firstly, more context is provided as default. This means that the learner can see not just up to 40 characters either side of the node, but the full sentence of the current concordance line plus the one before and the one afterwards. Since the theory of Lexical Priming shows that position in text is important, if the top sentence is not text initial, three dots ... indicate more is above.

Similarly at the bottom of the concordance box, if the last sentence displayed is not the last sentence of the text, ... is displayed too. Paragraphing is also an important feature of the

primings of words. While concordancers like *The Sketch Engine* and *WordSmith Tools* can show paragraph breaks as <p> tags, in *The Prime Machine* paragraphs are show with line spaces and indenting.

When concordance lines are retrieved, the rows from the `cb_corpus_texts` table for each node and the rows from the `cb_corpus_sentences` table for each sentence before, containing and after the node are retrieved as well as the rows from the `cb_corpus_words` table itself. Paragraphing and heading information are therefore available for each sentence and these values are used to determine how each concordance line appears on the card. The information about paragraphing, headings, and many other aspects of the sentence and word level environment are put into the database through the refactoring process and later compressed. For text position, some of the information is extracted from XML or other explicit codes present in the original text files, some is calculated by analysing words or sentences and their position relative to other words and sentences in the same text, and some is determined by analysis of *CLAWS* tags. Since each word in the database is attached to one sentence, all of the features are essentially limited by the accuracy of the sentence segmentation process.

As was noted, in Chapter 3, some corpora may already have *CLAWS* tags encoded in the corpus files. For the XML edition of the *BNC* which was used in this project, the *CLAWS* tags provided were based on the C5 tagset. When texts from corpora without *CLAWS* tags were processed, however, the newer C7 tagset was used. Therefore, in the tables showing the interpretation of tags for the identification of various features of Lexical Priming which are described in this chapter, columns will be shown for both C5 and C7 tags.

The Lines Tab in the application provides a KWIC list of concordance lines, and although this is much more similar to other concordancers, the design also incorporated some consideration of the position in paragraph and sentence. One important difference is that the component used is more like a grid from a spreadsheet than a table or text box. By using the *TAdvStringGrid* component (TMS_Software, 2011), it was possible to fill the grid with the wider context window, but have different horizontal alignment settings for the parts of the concordance line data before and after the node word. The column containing text to the left of the node is right-aligned, while the column containing text to the right of the node is left-aligned. Figure 5.6 shows how these alignment settings keep the node in its central position, while permitting more of the wider context window to be visible when the width of the grid is increased.

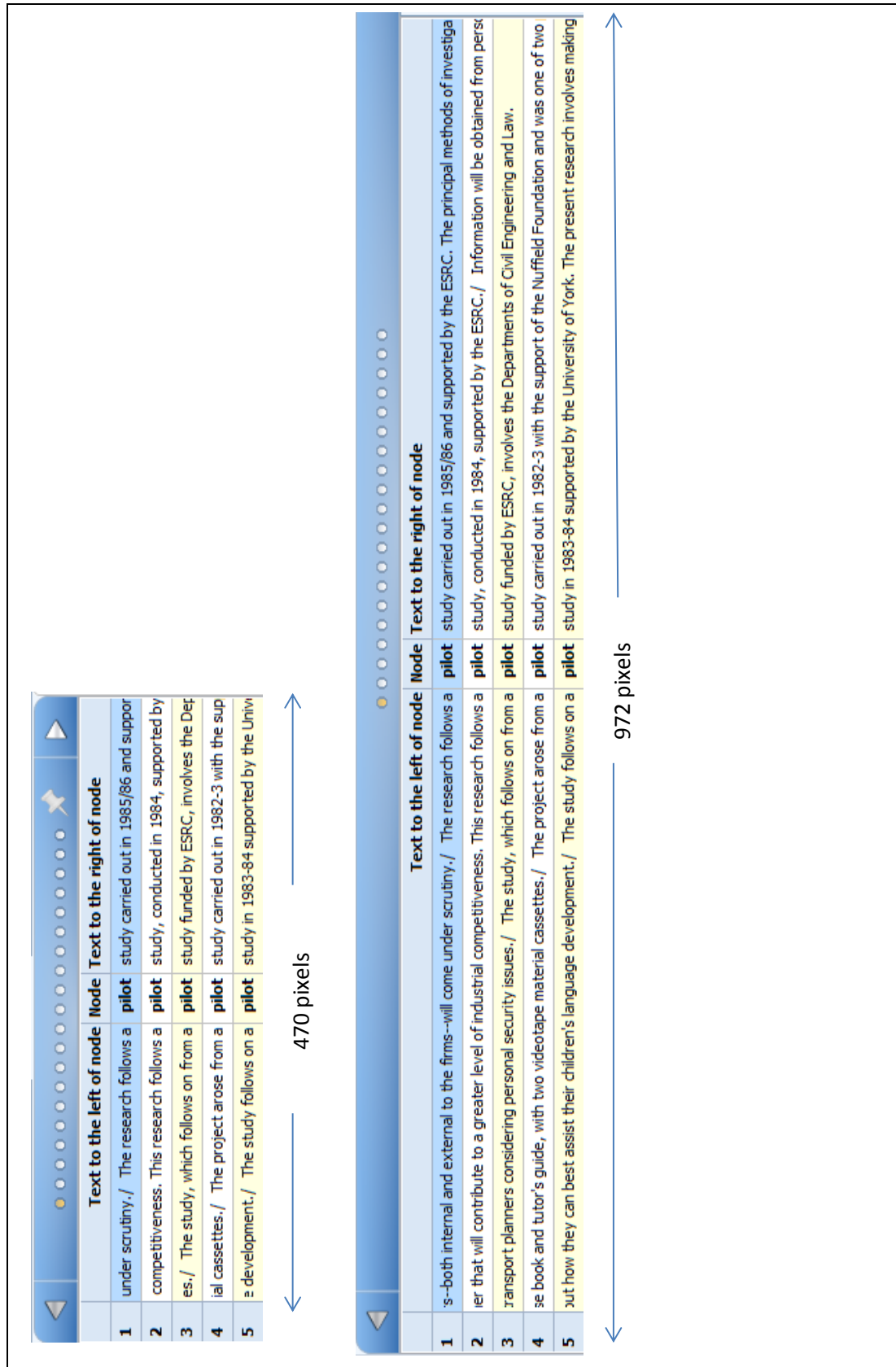


Figure 5.6: Text alignment in different columns of the grid on the Lines Tab for a relatively narrow width (top) and a relatively wide width (bottom) with incidental data from the *BNC: Academic* sub-corpus for a search on the node *pilot*. The currently selected line is shown with a blue background.

If data are exported from the application as a sheet, the *Microsoft Excel* compatible workbook also retains these extended left and right contexts. This means that if the window of the application or the width of the columns in *Excel* is increased, more context will become visible. In order to make it clear where the breaks between paragraphs occur, the / symbol is used. An additional feature is that the visible context of the concordance lines on the Lines Tab can be reduced to only include the sentence containing the node through the use of a button on the bottom panel. Figure 5.7 and Figure 5.8 show the display of concordance lines in these two modes, and the icons used on the tool bar to represent these.

	Text to the left of node	Node	Text to the right of node
1	ng personal security issues./ The study, which follows on from a	pilot	study funded by ESRC, involves the Departments of Civil Engine
2	reater level of industrial competitiveness. This research follows a	pilot	study, conducted in 1984, supported by the ESRC./ Informatio
3	al to the firms--will come under scrutiny./ The research follows a	pilot	study carried out in 1985/86 and supported by the ESRC. The pr
4	about, the advantages of renewable energy./ The findings of a	pilot	study carried out by NATTA on attitudes to the Severn Barrage
5	ice to reconstruct the landscape in detail./ This survey follows a	pilot	study of the West Midlands in the Anglo-Saxon period in which p
6	ith two videotape material cassettes./ The project arose from a	pilot	study carried out in 1982-3 with the support of the Nuffield Four
7	: their children's language development./ The study follows on a	pilot	study in 1983-84 supported by the University of York. The prese
8	n the language development./ This study extends the work of a	pilot	study funded by ESRC (C/00/23/2220) which examined the inter
9	illy managed is to be explored./ The research, which builds on a	pilot	study carried out in 1984, involves a depth case study of a large
10	ttish education system./ The current project was preceded by a	pilot	study carried out in 1980-81, supported by the former SSRC.
11	Subjects, methods, and results/ A	pilot	study was carried out to test the hypothesis that British trained
12	Methodology/ Initially a small database would be created as a	pilot	study. This database would be created by three S3 pupils from a
13	n graft thrombi; moreover, it seems to be less immunogenic. In a	pilot	study with 10 mg recombinant staphylokinase given over 30 min
14	triple therapy regimen which had previously been evaluated in a	pilot	study./ Methods ...
15	of laser palliation for advanced rectal and rectosigmoid cancer: a	pilot	study Abstract ...
16	rol medium was prepared with a similar dilution of alcohol and in a	pilot	study had no obvious effect on proliferation./ ORGAN CULTUR
17	Usually interferon is given in a single continuous course./ From a	pilot	study comparing the antiviral effect of interferon, acyclovir and
18	d to understand the matters the respondent is reporting on. In a	pilot	study it allows researchers to test out various lines of questionin
19	Award Title: Housing inheritance and wealth: a	pilot	study Award Type: ...
20	Title: The market availability of land for industrial development: a	pilot	study Award Type: ...
21	primary and four secondary) in two Local Education Authorities. A	pilot	study has already traced the progress of these new bodies since

Figure 5.7: The Lines Tab showing left and right contexts for the node extending beyond sentence boundaries, with concordance lines for the node *pilot* in the *BNC: Academic* sub-corpus.

	Text to the left of node	Node	Text to the right of node
1		pilot	study funded by ESRC, involves the Departments of Civil Engine
2	/ The study, which follows on from a	pilot	study, conducted in 1984, supported by the ESRC.
3	This research follows a	pilot	study carried out in 1985/86 and supported by the ESRC.
4	/ The research follows a	pilot	study carried out by NATTA on attitudes to the Severn Barrage
5	/ The findings of a	pilot	study of the West Midlands in the Anglo-Saxon period in which p
6	/ This survey follows a	pilot	study carried out in 1982-3 with the support of the Nuffield Four
7	/ The project arose from a	pilot	study in 1983-84 supported by the University of York.
8	/ The study follows on a	pilot	study funded by ESRC (C/00/23/2220) which examined the inter
9	/ This study extends the work of a	pilot	study carried out in 1984, involves a depth case study of a large
10	/ The research, which builds on a	pilot	study carried out in 1980-81, supported by the former SSRC.
11	/ The current project was preceded by a	pilot	study was carried out to test the hypothesis that British trained
12	/ Initially a small database would be created as a	pilot	study.
13	In a	pilot	study with 10 mg recombinant staphylokinase given over 30 min
14	triple therapy regimen which had previously been evaluated in a	pilot	study.
15	of laser palliation for advanced rectal and rectosigmoid cancer: a	pilot	study
16	rol medium was prepared with a similar dilution of alcohol and in a	pilot	study had no obvious effect on proliferation.
17	/ From a	pilot	study comparing the antiviral effect of interferon, acyclovir and
18	In a	pilot	study it allows researchers to test out various lines of questionin
19	Housing inheritance and wealth: a	pilot	study
20	The market availability of land for industrial development: a	pilot	study
21	A	pilot	study has already traced the progress of these new bodies since

Figure 5.8: The same display when contexts are reduced to only include the sentence containing the node.

Limitations on the display of text in the grid component used for KWIC display mean that it is not possible at this time to align text to the right and use HTML mark-up to provide bold features, etc. For KWIC display, text to the left of the node would need to be right-aligned so that the word nearest the node is always visible, rather than left aligned. However, since screen sizes vary considerably and users may want to change the size of the application window so they can see other applications behind, it is not really practical to use a method to programmatically determine how many characters are visible in the column. One way of overcoming this would be to use a fixed-width font, but this would make the display look very old-fashioned and it is felt that more natural looking fonts

without bolding for headings is preferable to fixed-width “console” fonts. However, the Cards Tab provides easy access to a more fully formatted view of the concordance lines, and since the card for the currently selected concordance line on the Lines Tab is also visible (except in compare mode), it was felt that using the right-alignment features of *TAdvSmoothGrid* but with limited formatting inside each cell would be the best compromise.

The short summary of the exploration of different components used for the Cards Tab which is given above demonstrates how various factors influenced the design process. As well as trying to make the application look reasonably modern, the aim was to make many of the buttons and other visual elements “familiar”. Within the field of computer science, “interface idioms” are used to refer to “recognizable types or styles of interfaces”, and “patterns” refer to familiar components of a visual design which can be used in different contexts (Tidwell, 2010, pp. xvi, xvii). At the same time as the software was being developed, changes in the way data are shown natively in operating systems were also taking place. From contact lists and instant messaging on *iPhone/iOS* and *Android* to online news and *Windows 8* tiles, lists of boxes containing content to be scrolled through, zoomed in on and studied in more detail were becoming more common in other applications too. These changes had an influence on the design of many aspects of *The Prime Machine*.

5.1.3 Position in text

The special role of words and combinations of words in text initial and paragraph initial sentences is clearly presented by Hoey (2005), but the processing requirements for software for calculating if something is text or paragraph initial also mean that the system will be able to mark sentences as paragraph ending or text ending. The kind of formulaic teaching of topic sentences which I had been familiar with earlier in my career and which Hoey’s theory challenges, also often provides students with formulaic exercises for writing summary sentences at the end of paragraphs and essays (e.g. Hogue, 1996; Oshima & Hogue, 1997). It was decided that an indication of tendencies for words and collocations to be used at the end of paragraphs and texts would also be included.

The main procedures for adding Lexical Priming environment information take place in the post-*CLAWS* phase in the refactoring application. During the pre-*CLAWS* process, an additional procedure is completed for corpora such as the *Financial Times* corpus, which use carriage return codes to mark paragraph boundaries, whereby <p> tags are added to the texts and a carriage return following a full stop, question mark or exclamation mark is

also interpreted as the end of a paragraph. During the post-*CLAWS* process for all corpora, each word is read from the *CLAWS* output file (or *CLAWS* encoded XML), and information about its position, capitalization, spacing and other information is stored in a table. Once a new sentence marker is reached, the sentence is then processed. It is at this point that any *CLAWS* tags which are required for identification of priming environments are also processed. Information about the sentence is also added to a separate table so it can be exported to the `cb_corpus_sentences` table in the database. This sentence level information includes the position in text and position in the paragraph as well as a code representing whether or not it is a title or heading. Since the texts are read linearly from beginning to end, after the first sentence in a text which is not part of a heading has been marked as text initial, each sentence picked up from the post-processed files assumes that it is the final sentence in the text, until another sentence is started and then it is set back to being unmarked. In a similar way, as new paragraph tags are read, the paragraph counter is increased and the next sentence is marked as paragraph initial. All further sentences are set to be paragraph ending as default, but each subsequent new sentence which is part of the same paragraph resets the previous sentence's position to be unmarked. The elements in a raw corpus file which are interpreted as the beginning and end of titles and headings are flexible, but typically, these would be `<H1>`, `<H2>` or `<H3>` tags in XML. The output files from the Refactoring application are MySQL dump files which contain details about each word and each sentence and, as outlined in Chapter 3, these are loaded into MySQL databases for further compression and processing.

In terms of the application of the log-likelihood statistic for the measurement of priming information, the SQL script to calculate the sentence level priming information uses the same contingency table and follows the same process for each kind of priming. In fact, the script is automatically created using a small *Delphi* program which reads information about priming environments and labels from an external table and then puts these values into a SQL script template. First, a list of sentences containing each feature is created in a temporary table using a "where" clause. Then, the total number of words which are contained in these sentences is determined by summing the sentence length values stored in the table. After that, another temporary table is used to store the number of times each item in the lexicon occurs within the particular kind of environment. Finally, a contingency table is formed as shown in Table 5.2. Summary data is stored for all log-likelihood values reaching a BIC value of 2. As explained in Chapter 4, following Wilson (2013), Bayes Factors

are used as a way of standardizing the cut-off point for the key word method, and the level of significance is stored using the BIC interpretation given there.

Table 5.2: Contingency table for sentence level features

	Corpus One	Corpus Two
Freq. of word	A = inside sentences with the specific feature	B = <i>Outside the sentences with the specific feature</i>
TOTAL	C = <i>Count of all words inside sentences with the specific feature</i>	D = Whole corpus – C

For features related to position in text, it was necessary to decide how to treat sentences which were marked as being both the beginning and the end of the paragraph or text. These would include sentences occurring in single sentence paragraphs, but also sentences occurring in texts comprised of single paragraphs, or sentences from texts comprised of single sentences. It was decided that these cases would not be counted in the measures for beginnings or endings⁴¹.

During the extraction and storage process for collocations which is summarized in Chapter 4, information about the priming environments for each occurrence of the node of each collocation is also extracted and held in a temporary table. This means that the same process can be followed to calculate tendencies for collocations to be primed in certain environments for every collocation stored in the database. The same *SQL* script generating application uses the priming feature table to generate loops within the collocation extraction script too, and tendencies are stored in summary tables for each length of collocation (see Section 4.7 of Chapter 4 for an explanation of the lengths of collocations).

Within the tables of summary data that are stored in the database, there will obviously be degrees of strength, with some items self-evidently marked out as having strong tendencies and others where the statistics are trivial. While I am aware that my software could be used by linguists to explore the wider territory from the very strong textual colligations to the weaker ones and into the areas where no textual colligation is visible, my focus is rather different. There are results where the figures are weaker and while these may still be of interest to a linguist, they may not be very useful from a language learning perspective. However, some of the results reveal tendencies which may be of benefit to an

⁴¹ Single sentence paragraphs are excluded from the measure for position in paragraph, but they may be counted in the measures for occurrence in the first or last paragraph of the text.

advanced language learner. In this thesis, therefore, the reader’s attention is only being drawn to places where the evidence is very clear. First, as each group of features is introduced, the reader will be given evidence of the statistics that make it quite obvious that there is something going on, and that the software is able to measure these tendencies automatically. After that, examples will be presented as they appear in the life-ring help screens, stripped of all the technical evidence, where they are provided in order to help explain to an advanced learner what each feature was designed to measure. The reader is not being asked to dwell too heavily on whether there is anything remarkable or surprising about the tendencies of the example words to be used in these specific contexts, but rather to consider whether given a learner’s interest in the use of one or more of these words it would not be to his or her advantage to have attention drawn to the existence of such tendencies. The way in which their attention is to be drawn is explained towards the end of this chapter. By taking each set of features and introducing them one by one, this chapter and the help screens for these features take the feature and the measure as the primary focus, with the examples being provided as a means of explaining what it is that is being measured or claimed. However, within the software itself, the focus is on the words which users have entered into the software: words which learners want to explore as they compose or edit their own writing, or as a result of teacher or peer feedback on a particular instance in their own writing that they may need to consider adapting. The data on the Graphs Tab are there as a means of prompting the user to consider exploring these areas in the concordance lines.

The first sub-menu on the Graphs Tab is based on the occurrences of words as part of titles. Table 5.3 shows the overall proportion of tokens in the *BNC: Newspapers* sub-corpus which are located within titles and the figures for the word *sport*. In the tables of examples figures reaching the BIC level of “Strong evidence” are shown in bold type.

Table 5.3: Tendencies to occur in text titles in the *BNC: Newspapers* sub-corpus.

	Frequency	Title
Tokens in the Newspaper sub-corpus	10,809,050	2.3%
<i>sport</i>	1,268	30.5%

As can be seen, more than 30% of the occurrences of the word *sport* in this corpus occur as part of a title. One of the reasons for this high figure is that *The Independent*, which is one of the newspapers in the corpus, had a regular column called “Sport in Short”. Figure 5.9

shows how some of this information is provided on the Life Ring help screen for the *Text Title* submenu on the Graphs Tab.

Graphs Tab: Title

This shows the proportion of concordance lines which are taken from the titles of texts.

Examples from the *BNC: Newspapers* sub-corpus

Only 2.3% of all words in this corpus are part of a title.

Yet three out of ten of the occurrences of the word *sport* are titles. This is partly because one of the newspapers included in this corpus had a regular column called “Sport in Short”.

Click [here](#) for more details.

Notes:

- Titles appear in bold text on the Cards.
- Not all the texts in all the corpora have titles included, so some data may be missing.
- It is **always** a good idea to look at the concordance lines to see what patterns of priming seem to occur. To filter the concordance lines, click to clear the tick mark against one or more of the features. Then click on the filter or compare buttons.

Figure 5.9: Information provided on the Life Ring help screen for the *Text Title* submenu on the Graphs Tab.

The information on this help screen includes some simple proportions, highlighting how a specific word in a specific corpus has an unusually high tendency to occur in the title of a text. Where space permits, each help screen also includes information about the extent to which the feature is likely to be accurately represented, means of filtering results, encouragement to look at concordance lines and any other information about how the feature is displayed. Although all of the main help screens are primarily focussed on language and do not include detailed percentage values, some of these values can be revealed within the software by clicking on links marked “Click here for more details”. The help screens draw on data from a number of different corpora. For the second measure, which is related to headings, data are taken from the Academic sub-corpus of the *BNC*. Table 5.4 shows the overall proportion of tokens in this sub-corpus which occur as part of headings and gives the figures for the words *conclusion* and *ending*.

Table 5.4: Tendencies to be used (or not used) in paragraph headings in the *BNC: Academic* sub-corpus.

	Frequency	Heading
Tokens in the Academic sub-corpus	18,085,284	0.6%
<i>conclusion</i>	2,154	13.0%
<i>ending</i>	299	-

It will be no surprise to see that the word *conclusion* occurs fairly frequently in the sub-corpus as part of a heading, while there are no instances of the word *ending* being used in headings. Information about paragraph heading tendencies which is provided on the help screen for headings is shown in Figure 5.10.

Examples from the *BNC: Academic* sub-corpus

Only 0.6% of words in this corpus are part of a heading.

Yet 13% of the occurrences of the word *conclusion* are paragraph headings and none of the occurrences of the word *ending* are paragraph headings. Obviously, the heading used for the last section of an academic article is usually *Conclusion*, but it also occurs very frequently within sentences.

Figure 5.10: Information provided on the Life Ring help screen for the *Heading* submenu on the Graphs Tab.

By using very simple examples such as “conclusion”, the help screens try to show how the measurements work in the way which they would be expected to. Since the software was primarily designed for language learners to use within university contexts, the words which were selected as examples are often connected with academic topics. Conceptually, the idea of counting words in titles and headings should be fairly easy for learners to grasp. Therefore, the examples only need to explain some obvious uses which can be readily understood. It is worth noting that headings are also stored as part of the metadata for texts in the corpora, so, as will be explained in Chapter 6, tendencies for use as part of a heading may also be visible on the Tags Tab.

The measures for “Position in text” are probably a little less easy for learners to understand conceptually, since comparing frequencies across different sections of a text is a little less intuitive than the idea of looking for which words are used as part of headings. For these measures, examples are provided using data from some of the *Hindawi* corpora. Table 5.5 shows the overall proportion of tokens situated in the first and last sentence of texts, using data from the *Hindawi Computer Science* corpus.

Table 5.5: Tendencies to be used in the first or last sentence of texts in the *Hindawi Computer Science* corpus.

		Frequency	First sentence	Last sentence
Tokens in the entire corpus		9,847,424	0.7%	0.5%
Sense of change / progress / growth	<i>witnessed</i>	31	32.3%	-
	<i>advances</i>	238	16.8%	1.7%
	<i>tremendous</i>	60	16.7%	1.7%
	<i>worldwide</i>	94	16.0%	1.1%
	<i>increasingly</i>	280	14.6%	0.7%
	<i>ubiquitous</i>	241	12.4%	-
Sense of looking forward to the future	<i>intend</i>	112	-	14.3%
	<i>future</i>	2,429	0.8%	11.5%
	<i>promising</i>	453	2.4%	4.2%

As can be seen, the overall proportions of tokens are very small, yet a number of words show quite strong tendencies. The first group of words are related to some sort of sense of change or progress or growth. These occur in the first sentence of a text many times more often than would be expected by chance. The second group of words are related to the future and occur in the last sentence of a text many times more often than expected by chance. Some of the words which occur frequently in the first or last sentence of a text also show tendencies at the paragraph level. Table 5.6 shows the overall proportions and figures for text initial and text ending paragraphs, including results for some of the same words as the previous table. Some synonyms for each group of example words are also given for contrast. The synonyms for these examples were selected using the thesaurus which is built into *Microsoft Word*.

Table 5.6: Tendencies to be used in the first or last paragraph of texts in the *Hindawi Computer Science* corpus.

		Frequency	First paragraph	Last paragraph
Tokens in the entire corpus		9,847,424	3.2%	0.5%
Sense of change / progress / growth	<i>advances</i>	238	26.5%	1.7%
	<i>increasingly</i>	280	24.6%	0.7%
	<i>emerging</i>	232	22.4%	1.3%
	<i>novel</i>	1,304	18.3%	1.0%
	<i>growing</i>	368	16.0%	0.3%
	<i>enhancements</i>	111	4.5%	0.9%
	<i>expansion</i>	323	1.2%	1.2%
	<i>budding</i>	46	4.3%	-
	<i>evolving</i>	188	6.9%	0.5%
	<i>fresh</i>	76	2.6%	-
<i>unique</i>	963	5.3%	0.3%	
Sense of looking forward to the future	<i>hope</i>	140	1.4%	22.1%
	<i>future</i>	2,429	2.5%	11.6%
	<i>anticipate</i>	40	2.5%	2.5%
	<i>expect</i>	303	1.0%	2.3%
	<i>trust</i>	257	1.9%	1.2%

The help screens for these two sets of measure can be seen in Figure 5.11 below. The help screens do not include the synonyms, but focus on the examples for which there is positive evidence.

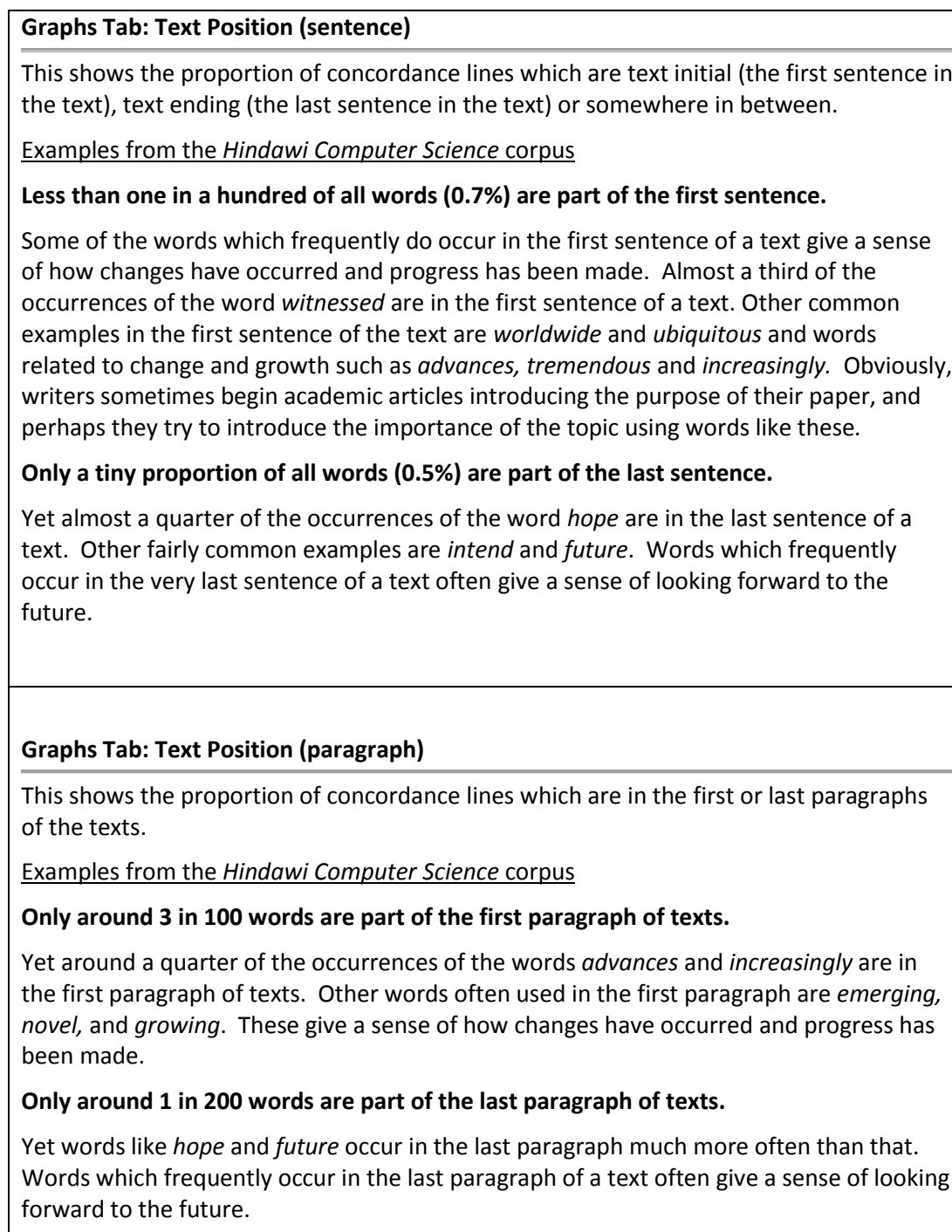


Figure 5.11: Information provided on the Life Ring help screen for the *Text Position* submenus on the Graphs Tab.

The last set of measures within the “Position in text” group is related to the position of sentences within paragraphs. Table 5.7 shows the overall proportions of tokens in the first and last sentence of the paragraph, with data from the *Hindawi Biological Science* corpus. As before, some synonyms are provided for contrast.

Table 5.7: Tendencies to be used in the first or last sentence of paragraphs in the *Hindawi Biological Science* corpus.

		Frequency	First sentence	Last sentence
Tokens in the entire corpus		23,107,819	18.2%	17.9%
Introducing or linking to a topic.	<i>discuss</i>	1,048	57.0%	19.0%
	<i>discusses</i>	141	56.7%	12.8%
	<i>summarize</i>	345	50.4%	17.1%
	<i>summarizes</i>	281	44.1%	14.2%
	<i>focuses</i>	342	38.9%	15.8%
	<i>focus</i>	1,900	31.0%	18.9%
	<i>consider</i>	881	24.0%	18.6%
	<i>considers</i>	108	18.5%	17.6%
	<i>argue</i>	215	19.1%	24.2%
	<i>argues</i>	84	26.2%	16.7%
Signposting	<i>center</i>	1,365	16.1%	15.9%
	<i>centers</i>	360	18.6%	18.9%
Signposting	<i>furthermore</i>	5,368	21.9%	24.2%
	<i>besides</i>	1,304	19.1%	21.0%

It is clear that there seems to be a group of words which occur in sentences at the beginning of paragraphs more frequently than would be expected by chance. The results for words like *discuss*, *summarize* and *focus* show strong tendencies and they fit well with intuitions that they may be used in academic texts at the beginning of paragraphs to link to a previous topic or to introduce a new one. Words with a similar meaning do not always share the same tendencies and this is evident in the lower percentage values for words like *consider*, *argue* and *centre*. In the results provided here, the part-of-speech tags have not been used, so no attempt to distinguish use of *focus* or *centre* as a noun or a verb has been made. With regard to signposting words, it can be seen that *furthermore* occurs more often than would be expected by chance in sentences at the beginning of paragraphs, but also occurs even more frequently in the last sentence of paragraphs. The other signposting word which is given as an example does not show such strong tendencies. Some of the positive evidence from this table is given on the help screen, and this is shown in Figure 5.12.

Graphs Tab: Paragraph Position

This shows the proportion of concordance lines which are from the first or last sentences of paragraphs.

Examples from the Hindawi Biological Science corpus

Overall, around 1 in 6 of all words are in the first sentence of paragraphs.

Yet more than half of the occurrences of the words *discuss* and *discusses* occur in the first sentence. Other words which occur quite frequently in the first sentence of paragraphs are *summarize*, *summarizes*, *focus* and *focuses*.

The word *furthermore* tends to occur quite frequently in the first or last sentence of a paragraph.

Obviously, writers sometimes introduce a paragraph by linking to details which have already been discussed, or by summarizing the focus of the new paragraph. Towards the end of a paragraph, they may use signals like *furthermore* to take their points further.

Note:

- Some paragraphs are only one sentence long, so these results have not been counted.

Figure 5.12: Information provided on the Life Ring help screen for the *Text Position* and *Paragraph Position* submenus on the Graphs Tab.

The examples which have been presented so far use percentages and tendencies for words which are easy to understand and they are designed to help users of the software understand ways in which the actual results for each corpus query they execute could be interpreted. In this thesis, in order to provide a brief summary of the proportion of different words (types) which show different kinds of tendency across different corpora, some additional tables are provided for each group of features in this chapter. These tables use six corpora to show the proportion of types which have at least one priming tendency stored in the database for different frequency ranges. Token counts for each of the corpora across these different frequency ranges are shown in Table 5.8 below. The frequency ranges were chosen to include very low frequency items and very high frequency items. The two middle values of 20 or more and 100 or more were chosen as they represent items where the restrictions on the number of concordance lines which can be retrieved in the application without the user specifically requesting additional results could start to mean that summary information for all occurrences of the item may start to be more useful.

Table 5.8: Token counts for different frequency ranges for corpora used in the summary tables provided in this chapter.

Corpus	Tokens	≥ 3	≥ 20	≥ 100	≥ 1,000	≥ 10,000
<i>Hindawi Biological Sciences</i>	23,106,017	90,299	30,516	11,687	2,310	205
<i>Hindawi Computer Science</i>	9,833,111	35,754	13,614	5,666	1,143	92
<i>Hindawi Chemistry</i>	6,233,170	35,676	11,672	4,276	710	47
<i>Hindawi Maths</i>	12,465,839	33,872	12,789	5,268	1,143	123
<i>BNC Academic</i>	18,085,284	70,120	25,312	9,972	1,895	156
<i>BNC Newspapers</i>	10,809,050	60,138	21,262	7,623	1,145	98

In terms of sentence position, these corpora show slightly different proportions of types as having priming tendencies across the different frequency ranges, and this can be seen in Table 5.9.

Table 5.9: Proportions of types at different frequency thresholds showing at least one tendency for use in sentences in particular positions in text.

Sentence Position	≥ 3	≥ 20	≥ 100	≥ 1,000	≥ 10,000
<i>Hindawi Biological Sciences</i>	2.9 %	7.7 %	16.8 %	43.7 %	70.7 %
<i>Hindawi Computer Science</i>	4.0 %	8.9 %	17.9 %	38.4 %	63.0 %
<i>Hindawi Chemistry</i>	4.0 %	9.4 %	19.1 %	41.5 %	72.3 %
<i>Hindawi Maths</i>	5.6 %	13.1 %	26.3 %	55.9 %	82.9 %
<i>BNC Academic</i>	3.7 %	7.9 %	16.1 %	38.9 %	64.1 %
<i>BNC Newspapers</i>	5.7 %	13.6 %	26.7 %	55.5 %	82.7 %

These tables are not provided to the user of the software, but they demonstrate how a fair proportion of the mid-frequency items in all of the corpora listed can be considered, on the basis of the procedures described above, to have a tendency to be used in particular positions. As expected, the figures for the *BNC: Academic* sub-corpus are noticeably lower than those for the *BNC: Newspapers* sub-corpus. For these particular features, the figures for the *BNC: Academic* sub-corpus are the least reliable since when these are calculated the sampling technique is not taken into account and many of the texts are not complete. However, these results provide an interesting comparison with other research which has shown 1 in 40 types having a text or paragraph initial preference in a corpus of *Home News* articles (O'Donnell, et al., 2012).

5.1.4 Position in sentence

As each sentence is added to the MySQL dump file in the refactoring application, one of the additional pieces of information added to each word in the table is its position in the sentence in terms of *Theme* or *Rheme*. Within the field of Systemic Functional Linguistics (SFL), there has been considerable debate over what the extent of *Theme* should be, and a full discussion of this and the extent to which Theme-Rheme might map onto elements in other languages is beyond the scope of this thesis. The *unmarked Theme* typically maps to *Subject*, but there are a wide range of ways in which marked structures may be analysed (Fontaine, 2013; Halliday & Matthiessen, 2004; Thompson, 2004). The definition of *Theme* for the Graphs Tab statistics in *The Prime Machine* takes everything from the start of each sentence up to but not including the first main verb. This definition is certainly an oversimplification, and students of SFL and researchers would need to be mindful of the severe limitations this places on the results available within the software. However, working with this definition as a starting point for analysis can also be quite fruitful as a way of exploring how understanding of the role of constituents develops (Ravelli, 1995). Berry (1996) reports on this “preverb” approach from a teaching perspective, but notes there are limitations. Nevertheless, this simple definition was adopted in order to operationalize an automatic labelling of words as being *Theme* or *Rheme* which would provide results arrived at by a similar means to that used by Hoey (2005). In *CLAWS* the tags for main verbs which are used in the refactoring application can be seen in Table 5.10 below.

Table 5.10: CLAWS tags used to identify the beginning of “Rheme”

	C7 Tag Set	C5 Tag Set
Start of	VB0 VBDR VBDZ VBM VBR VBZ	VBB VBD VBZ VDB VDD VDZ
Rheme	VD0 VDD VDZ VH0 VHD VHN VHZ VM VMK VV0 VVD VVN VVNK VVZ	VHB VHD VHN VHZ VM0 VVB VVD VVN VVZ

Working from the top of the table which contains the first word for the sentence, the application moves down the column of *CLAWS* tags until one of the tags listed in Table 5.10 above is encountered. The word in this row and all the rows below it are marked as being “Rheme”. There are some sentences in the database which do not contain any of these tags, and so rather than have the whole sentence marked as “Theme”, these are marked as “unknown”. Thus the Theme-Rheme setting has three states: Theme, Rheme and unknown.

The contingency table for word level features is very similar to that used for sentence level features (Table 5.2) but differs slightly, and it can be seen in Table 5.11 below. For collocations, the contingency table is based on the number of occurrences of the node of the multi-word unit in each environment.

Table 5.11: Contingency table for word level features

	Corpus One	Corpus Two
Freq. of word	A = where the specific feature has been marked	B = where the specific feature is absent
TOTAL	C = <i>Count of all words with the specific feature</i>	D = Whole corpus – C

The marking of Rheme in the database also meant that it was possible to include a “penalty” for Theme or unknown Theme-Rheme as one of the measures in the implementation of the concordance ranking score based on *GDEX* (Kilgarriff, et al., 2008) which was introduced in Chapter 4.

However, this approach to analysing Theme-Rheme provides only a very limited mapping to the Theme-Rheme analysis of SFL, and it also leads to quite a heavy reliance on *CLAWS* and its ability to assign the correct POS tag.

Table 5.12 shows the overall proportions of tokens in Theme and Rheme for the *BNC: Academic* sub-corpus. As can be seen, several words related to the concept of academic research have been chosen as examples, and there are also some examples of adjectives.

Table 5.12: Tendencies to be used in Theme or Rheme in the BNC: Academic sub-corpus.

		Frequency	Theme	Rheme
Tokens in the sub-corpus		18,085,284	17.2%	78.8%
Matters of academic research	<i>aim</i>	1,616	51.0%	47.6%
	<i>experiment</i>	1,033	34.6%	60.4%
	<i>research</i>	10,338	31.8%	45.5%
	<i>questionnaire</i>	488	31.1%	63.1%
	<i>data</i>	7,469	30.6%	65.4%
Related to certainty	<i>likely</i>	6,680	2.9%	96.7%
	<i>explanation</i>	1,870	27.4%	70.6%
	<i>likely explanation</i>	25	60.0%	40.0%
Related to doubt	<i>unlikely</i>	1,349	1.6%	98.1%
	<i>dubious</i>	139	15.8%	83.5%
Related to difficulty	<i>difficult</i>	4,686	3.5%	95.8%
	<i>challenging</i>	258	12.8%	82.2%
	<i>tough</i>	126	13.5%	83.3%
Related to practicality	<i>feasible</i>	287	5.6%	92.7%
	<i>viable</i>	204	11.8%	87.7%

Although *aim* is the only word in the group of matters of academic research which is used more frequently in Theme than in Rheme, it is clear that all of the words are often used as the subject of sentences. Indeed, these proportions are quite high considering that the top three words in the list could be used as verbs and so the Rheme figures for these would include instances when they are the main verb. The other sets of examples show tendencies for words which are adjectives to be used in Rheme. The word *likely* seems to occur more frequently in Rheme, but interestingly when used in combination with *explanation*, it is more likely to occur in Theme. As mentioned earlier, figures for occurrences of collocations in different environments are also provided in the software, meaning that different tendencies between base words and their collocations can be explored. The third group of words shows how the word *unlikely* is almost always used in Rheme, while the synonym *dubious* shares an overall tendency for use in Rheme but without meeting the threshold for statistical significance. The last two groups in the table have been selected to demonstrate similar patterns, with *difficult* and *feasible* occurring in Rheme more often than would be expected by chance, while *challenging*, *tough* and *viable* have slightly weaker tendencies. Figure 5.13 shows the examples provided on the help screen for this feature along with the explanation and cautionary notes.

Graphs Tab: Theme/Rheme

This shows the proportion of concordance lines where the word is in the Theme or Rheme of the sentence. The Theme is defined as all the words leading up to the first main verb but not including it. The Rheme is the rest of the sentence.

Examples from the *BNC: Academic* sub-corpus

Less than one in six of all words are in Theme.

Yet more than half of the occurrences of the word *aim* are in the Theme. Also more than 30% of the occurrences of the words *experiment*, *research*, *questionnaire* and *data* are in the Theme. Obviously, *aim*, *experiment*, *research*, *questionnaire* and *data* are often the subject of sentences in academic texts.

The vast majority of the occurrences of the words *difficult* and *likely* are in the Rheme. *Likely* and *difficult* could be used in the subject, but more frequently occur later in the sentence.

Note:

- This measure relies on automatic part-of-speech tagging and so the results may not be 100% accurate.

Figure 5.13: Information provided on the Life Ring help screen for the *Theme/Rheme* submenu on the Graphs Tab.

While *CLAWS* was considered a good choice even during initial development, there was a period of about 18 months from getting the data refactored and processed and starting to see the results on screen in the client application (although I did, of course, work with some preliminary data using the *MySQL* console along the way). One of the corpora which it was decided would form part of the testing of the system was *SpringerOpen's* collection of academic journal articles from the field of biomedicine (SpringerOpen, 2011). It was not clear how well *CLAWS* would be able to deal with articles of such a highly specialized nature without customizing the rules or *CLAWS* lexicon. The question arose of how well academic journal articles could be reliably tagged for the main verb, and it was decided that an alternative measure should also be made available for the analysis of textual colligation within sentences.

Some work on textual colligation has been carried out by dividing sentences into a fixed number of segments. When counting instances of a word occurring in sentence initial position, Scott (2006) included those occurrences within the first 20% of the sentence. He also conducted some analysis looking at frequencies across sentences which had been divided into 8 segments. In *The Prime Machine*, a measure based on word position as a proportion of the length of its sentence is also used. Each word in the database has a TinyINT value which gives its position in the sentence, using the proportion in 120ths. This

means that rows in the database can be selected according to whether this number corresponds to the first fifth (24), the first third (40), the last third (80) or the last fifth (96) of the sentence. Unlike the Theme-Rheme measure, this does not rely on POS tags, but it should be noted that the values are entirely dependent on correct sentence segmentation through *CLAWS*.

Table 5.13 shows the overall proportions of tokens in the first and last 20% of a sentence, with some examples of words from the *BNC: Academic* sub-corpus.

Table 5.13: Tendencies to be used in the first or last 20% of a sentence in the *BNC: Academic* sub-corpus.

		Frequency	First 20%	Last 20%
	Tokens in the sub-corpus	18,085,284	17.7%	22.3%
Attitude	<i>interestingly</i>	275	80.7%	1.5%
	<i>unfortunately</i>	618	78.6%	2.6%
	<i>fortunately</i>	148	75.0%	3.4%
Signposting	<i>furthermore</i>	1,442	91.9%	0.6%
	<i>moreover</i>	1,913	88.3%	0.9%
	<i>lastly</i>	132	84.8%	6.8%
	<i>firstly</i>	543	79.2%	0.2%
Other adverbs	<i>respectively</i>	1,303	2.5%	56.2%
	<i>properly</i>	1,008	10.7%	29.4%

The first group of words in the table may be classified as attitude stance adverbials. These kinds of adverbial are used more frequently in academic registers (Biber, et al., 1999). It is very clear that the three examples in the table tend to occur most frequently in the first 20% of the sentence, and almost never occur in the last 20%. The second group of signposting words show even higher proportions for all but one of the words, and match intuitions that they would be used most frequently at the beginning of sentences. The last group of words have the opposite tendency. The word *respectively* is used more than half of the time in the last portion of the sentence, and occurs very rarely in the first 20%. The word *properly* shows a slightly weaker tendency to be used at the end of the sentence, but it still reaches statistical significance. Figure 5.14 shows the examples used on the help screen to explain this measure.

Graphs Tab: Sentence Position

This shows the proportion of concordance lines where the word is in the first or last portion of the sentence. Sentences are divided into the first fifth, the first third, the last third and the last fifth.

Examples from the *BNC: Academic* sub-corpus

Obviously, around 20% of all words are in the first fifth of the sentence.

Yet more than three quarters of the occurrences of the words *interestingly*, *unfortunately* and *fortunately* are in the first 20% of the sentence. Also, as you might predict, many signposting words like *furthermore*, *moreover*, *firstly*, *lastly*, etc. also occur more than three quarters of the time in the first fifth.

Similarly, around 20% of all words are in the last fifth of the sentence.

Yet more than half of the occurrences of the word *respectively* and three out of ten of the occurrences of *properly* are in the last 20% of the sentence.

Notes:

- Sentences cannot always be divided into five equal chunks, but one might expect the figures to be close to 20% on average.
- The graphs also show figures for the proportion of occurrences in the first third and last third of the sentence.
- Results may be particularly interesting for collocations, so look out for the icons at the bottom of the screen.

Figure 5.14: Information provided on the Life Ring help screen for the *Sentence Position* submenu on the Graphs Tab.

The prevalence of tendencies for words to be used in particular positions in a sentence can be seen in Table 5.14 below.

Table 5.14: Proportions of types at different frequency thresholds showing at least one tendency for use in particular positions in a sentence.

Word Position	≥ 3	≥ 20	≥ 100	≥ 1,000	≥ 10,000
<i>Hindawi Biological Sciences</i>	6.9 %	19.9 %	42.2 %	77.0 %	93.7 %
<i>Hindawi Computer Science</i>	9.0 %	22.9 %	44.1 %	76.3 %	95.7 %
<i>Hindawi Chemistry</i>	8.0 %	23.5 %	49.0 %	83.8 %	91.5 %
<i>Hindawi Maths</i>	9.6 %	24.5 %	46.1 %	77.8 %	94.3 %
<i>BNC Academic</i>	7.8 %	20.9 %	43.1 %	80.8 %	98.1 %
<i>BNC Newspapers</i>	9.3 %	24.9 %	51.3 %	86.7 %	98.0 %

It is noticeable that these figures are rather higher than those for sentence position in text (Table 5.9). It is also noticeable that the figures for the *BNC: Academic sub-corpus* seem to be more similar to those of the other corpora, and this is probably because the results for the position in sentence measures are not marred by incomplete data as in the case of

sentence positioning. However, it is very important to remember that the tendencies for particular words may be quite different across different corpora.

5.1.5 Icons to represent different positions

Rather than just relying on names for each priming feature and its set, icons were created to try to convey some information about the priming environment visually. Designing icons showing tendencies for the position in a text or paragraph seemed fairly straightforward because positions could be indicated by highlighting the top or bottom of a small collection of lines which could represent complete texts or paragraphs. However, graphic design is not one of my strong points, and as I started trying to design icons for some of the other features, I realized that some sort of element in each icon should be used to help indicate the level at which each feature would operate. The taskbar which appears at the bottom of a modern *Windows* computer shows icons for frequently accessed programs, and I noticed that the most obvious feature of *Microsoft Office* icons is a letter representing the name of the program: W for *Microsoft Word*, P for *PowerPoint*, and X for *Excel*. Similarly several other applications simply have a capital letter in a specific font with specific colouring. In *The Prime Machine*, icons were designed using *Axialis IconWorkshop 6* (Axialis_Team, 2011) which provides a large range of elements and symbols which can be incorporated into icons, as well as other useful icon editing features. A colour scheme was selected based on blues with green and orange to show positive or negative priming. After that, a capital letter or combination of letters was allocated to each kind of priming feature. For features representing positional information, “T” refers to text level, “P” to paragraph level and “S” to sentence level. For these features, highlighted lines or arrows give an indication of the location, and for Theme-Rheme the icon is based on the idea that *Theme* provides background information and is represented by a landscape, while *Rheme* relates to foregrounded information or detail and so is represented by a leaf. Users are not expected to fully understand the composition or meaning of the elements of the icons, but through repeated exposure it is expected that as with other icons computer users take for granted, the meaning will be associated with the feature over time. Examples of the icons related to position in text are shown in Table 5.15 below. It is possible that a word may be primed to occur in more than one position, for example both text initial and text ending, so icons are also available showing both these positions in green.

Table 5.15: Icons representing features related to position

 <p>Text level title;</p>	 <p>Paragraph level heading;</p>	 <p>Tends not to occur in headings at the paragraph level.</p>	
 <p>Text initial sentences;</p>	 <p>Text ending sentences;</p>	 <p>Paragraph initial sentences;</p>	 <p>Paragraph ending sentences.</p>
 <p>Text initial paragraphs;</p>	 <p>Text ending paragraphs.</p>		
 <p>Word in Theme;</p>	 <p>Word in Rheme;</p>	 <p>Word in first portion of sentence;</p>	 <p>Word in last portion of sentence.</p>

5.2 Colligation

The next two groups of priming features which will be introduced provide information about the typical colligations of words. The term *colligation* has been used by different researchers in the Firthian tradition in different ways (Hanks, 2013). In the theory of Lexical Priming, Hoey follows an example of Halliday's use of the term, and describes colligation as "a midterm relation between grammar and collocation" (Hoey, 2005, p. 43). The definition provided for the purposes of his presentation covers tendencies of an item to be used (or not to be used) with three aspects: grammatical patterns, grammatical functions and position in a sequence. The purpose of this section is to introduce how some of these

aspects are currently implemented in *The Prime Machine*; the measures are currently limited to complexity, modality, voice, polarity, articles and prepositions.

A few important limitations need to be mentioned. Firstly, as was explained in Chapter 3, all of these measures are highly dependent on the tagging process. The list of features is in no way intended to be exhaustive, and if the software is extended to work with other languages, these will almost certainly need different kinds of analyses. The development of appropriate taggers and testing of these for other languages is beyond the scope of this project. In order to accommodate some future changes, it would be possible to adjust the list of tags used in the refactoring application, but for others more fundamental changes may be needed at the refactoring and compression stages as well as in the client application itself. Secondly, in some cases, the process for the identification of certain features of colligation does not make a clear distinction between collocation and colligation or between colligation and identification of word class, but it should be remembered that the aim of the Graphs Tab is to provide indications to the user of how specific word forms are typically used, and from a language learner's point of view an absolute separation of these relations is unlikely to be necessary. Beyond making aspects of text position more accessible to language learners, the other features which appear on the Graphs Tab were selected to provide a range of measures with the level of sophistication of intermediate language learners of English in mind.

5.2.1 Sentence Complexity

The first feature in this section is sentence complexity, and this is included in the software as a way of indicating to users whether or not a word or collocation has a tendency to be used in simple or complex sentence structures. It would have been possible to look for much more specific grammatical features such as different kinds of subordination and to use a parser, but this would mean a heavy reliance on *CLAWS* for tagging and a third party parser. Garretson (2010) made use of Dependency Grammar in the colligation measurement processes in *CenDiPEde*, and *The Sketch Engine* has a very sophisticated way of representing the grammatical relations between collocates in its *Word Sketches*.

However, the main purpose for including information on complexity in *The Prime Machine* was to provide language learners with a very simple indication of whether or not it might be useful to look at the role of the word or collocation in short simple sentences or in longer more complicated structures. The process is very simple: the refactoring application checks all of the POS tags for the words in each sentence against a small range of POS tags

(shown in Table 5.16), and if any of these occur anywhere in the sentence it is marked as “complex”.

Table 5.16: CLAWS tags used to identify *complex* sentences.

	C7 Tag Set	C5 Tag Set
Complex Sentence	CS CSA CSN CST CSW	CJS CJT

Table 5.17 and Table 5.18 show the overall proportions of tokens occurring in complex and simple sentences for two of the sub-corpora from the *BNC*. Examples and figures from the *BNC: Newspapers* sub-corpus (shown in Table 5.17) chiefly focus on some strong tendencies related to the word *that* and several words which are often used in clauses containing *that*.

Table 5.17: Tendencies to be used in Complex or Simple sentences in the *BNC: Newspapers* sub-corpus.

	Frequency	Complex	Simple
Tokens in the Newspaper sub-corpus	10,809,050	33.7%	66.3%
<i>that</i>	71,485	80.0%	20.0%
<i>indications</i>	73	79.5%	20.5%
<i>stating</i>	74	75.7%	24.3%
<i>stated</i>	273	73.3%	26.7%
<i>argues</i>	175	72.6%	27.4%
<i>convince</i>	118	72.0%	28.0%
<i>survey</i>	1,096	41.1%	58.9%

Table 5.18: Tendencies to be used in Complex or Simple sentences in the BNC: Academic sub-corpus.

		Frequency	Complex	Simple
Tokens in the Academic sub-corpus		18,085,284	49.7%	50.3%
<i>survey</i>		2,226	38.9%	61.1%
Other matters of academic research	<i>project</i>	3,646	24.5%	75.5%
	<i>projects</i>	1,006	37.4%	62.6%
	<i>questionnaire</i>	488	28.3%	71.7%
	<i>research</i>	10,338	30.2%	69.8%
	<i>samples</i>	1,387	32.4%	67.6%
	<i>studies</i>	7,155	32.8%	67.2%
Related to evidence	<i>fact</i>	8,255	77.6%	22.4%
	<i>facts</i>	1,953	59.3%	40.7%
	<i>detail</i>	1,763	39.2%	60.8%
	<i>details</i>	1,349	37.6%	62.4%
	<i>statistic</i>	40	47.5%	52.5%
	<i>statistics</i>	1,052	34.7%	65.3%
Adjectives	<i>arguable</i>	115	91.3%	8.7%
	<i>practicable</i>	192	85.4%	14.6%
	<i>doubtful</i>	311	81.0%	19.0%
	<i>chargeable</i>	136	77.9%	22.1%
	<i>unreasonable</i>	386	76.7%	23.3%
	<i>achievable</i>	54	51.9%	48.1%
	<i>noticeable</i>	247	50.6%	49.4%
	<i>detectable</i>	170	47.1%	52.9%
	<i>observable</i>	168	46.4%	53.6%
	<i>considerable</i>	3,031	45.3%	54.7%
Adverbs	<i>reasonably</i>	921	72.7%	27.3%
	<i>unreasonably</i>	98	83.7%	16.3%
	<i>conclusively</i>	78	79.5%	20.5%
	<i>surprisingly</i>	519	40.8%	59.2%
	<i>remarkably</i>	285	43.9%	56.1%
	<i>moderately</i>	140	37.1%	62.9%

The figures for the word *survey* are given for these two sub-corpora, and it can be seen that while the actual proportions for this specific word are fairly similar, the difference in balance between complex and simple sentences in the sub-corpora overall mean that in the *Newspapers* sub-corpus the word *survey* would be marked as occurring in complex sentences more often than expected by chance, while in the *Academic* sub-corpus, it would be marked as occurring in simple sentences. One of the reasons for this difference is likely to be the way in which newspapers might use a *survey* to introduce a topic, while academic articles may tend to describe a survey in more simple terms as part of a methodology.

Other examples of words which are used to describe matters of academic research also have a tendency to be used in simple sentences.

Moving on to some of the other examples in Table 5.18, the group of words related to evidence show that the words *fact* and *facts* are usually used in complex sentences, while *detail*, *details*, *statistic* and *statistics* are more often used in simple sentences. The frequency of *statistic* is much lower than the frequencies of other words in this group, and the tendency for this word to be used in simple sentences does not reach a level of statistical significance. Looking directly at the collocation data for *fact* reveals that 44.0% of the occurrences of this word are accounted for by the collocation *fact that*. While this collocation represents a fair proportion of the instances of *fact* within complex sentences, it is also the case that quite a few cases have other complex structures.

The next two groups of words in the table are adjectives. The first 5 words show tendencies to be used in complex sentences, with particularly high figures for *arguable*, *practicable* and *doubtful*. Looking at the collocation data for these reveals that *arguable* collates with *that* and occurs as *arguable .. that* 57.4% of the time. Similarly, *doubtful if* and *doubtful whether* and *doubtful .. whether* account for a combined total of 41.4%, with *doubtful whether* taking the largest share. The second group of adjectives do not show any strong tendencies, except *considerable* which occurs in simple sentences to a statistically significant level.

The last two group of words are adverbs. The first three words show strong tendencies to be used in complex sentences, while the second three words do not show statistically significant tendencies.

Examples and additional information about this measurement are available on the Life Ring help screen. Figure 5.15 shows the text from this help screen, which includes information about different tendencies across the *BNC: Academic* and *BNC: Newspapers* sub-corpora. As before, users are also encouraged to analyse the concordance lines directly.

Graphs Tab: Complexity

This shows the proportion of concordance lines where the sentence is grammatically complex. A complex sentence includes at least one of the following.

- a subordinating conjunction (e.g. *if, because*)
- *as, than, that* or *whether* as a conjunction

Examples from the *BNC: Newspapers* sub-corpus

Only about one third of all words in the newspaper sub-corpus are in complex sentences.

4 out of 5 of the occurrences of the word *that* are marked as being complex. This is not at all surprising given that *that* is a conjunction.

However, several other words including *indications, stating, argues* and *convince* also occur more than 70% of the time in complex sentences. These words are probably used in complex sentences containing *that*.

The word *survey* occurs 41.1% of the time in complex sentences in newspapers.

Notes:

- In the *BNC: Academic* sub-corpus, the overall balance between complex and simple sentences is roughly equal (50%).

Figure 5.15: Information provided on the Life Ring help screen for the *Complexity* submenu on the Graphs Tab.

5.2.2 Modality

The second feature which is introduced in this section is information about modality. The measurements for modal verbs rely on two processes, with the first taking place in the refactoring application and the second taking place in the SQL scripts used to compress the data. In the refactoring application, words in each sentence are checked against the list of *CLAWS* tags for modals, and when a modal is found, the software adds information about it to the next four words of the same sentence. The information includes the distance from the modal and the specific word form of the modal itself. The sentence containing the modal is also marked to be flagged in the database as containing at least one modal. Table 5.19 shows the *CLAWS* tags used.

Table 5.19: *CLAWS* tags used to identify modal verbs

	C7 Tag Set	C5 Tag Set
Modal Verbs	VM VMK	VM0

Further categorization is performed on the MySQL server as the corpus data are compressed, with labels added according to the modal groups given in Biber et al. (1999, p.

489). Table 5.20 shows the slight simplification of these groupings which was needed as the interpretation of *CLAWS* tags in *The Prime Machine* is limited to single word units⁴².

Table 5.20: Limitations on the modal groupings included in the software

Groupings from Biber et al. (1999, p. 489)	Groupings in <i>The Prime Machine</i>	
	Included	Excluded
Permission/Possibility/Ability: <i>can</i> <i>could</i> <i>may</i> <i>might</i>	<i>can</i> <i>could</i> <i>may</i> <i>might</i>	
Obligation/Necessity: <i>must</i> <i>should</i> <i>have to</i> <i>(had) better</i> <i>(have) got to</i> <i>need to</i> <i>(be) supposed to</i> <i>ought to</i>	<i>must</i> <i>should</i> <i>need to</i> <i>ought to</i>	 <i>have to</i> <i>(had) better</i> <i>(have) got to</i> <i>(be) supposed to</i>
Volition/Prediction <i>will</i> <i>would</i> <i>shall</i> <i>be going to</i>	<i>will</i> <i>would</i> <i>shall</i>	 <i>be going to</i>
Past time: <i>used to</i>		 <i>used to</i>

During the processing and compression stages, priming tendencies for words and collocations to occur with modals are calculated on both the sentence level and word level. The sentence level calculation uses the information stored within each row of the table of sentences, simply measuring those containing at least one modal against those which contain none. The word level calculations use the information which has been added to tokens occurring in a four word window to the right of each modal and which has been stored in the rows of the table of words. The calculations and priming summaries for both levels are stored in the database, but only the word level is actually displayed and used in the client application.

Table 5.21 shows the overall proportions of tokens near the three groups of modal verbs in the *BNC: Academic* sub-corpus. Although there are some cases where the examples in this

⁴² Also, distinguishing non-modal uses of *have* and *got* could have been problematic.

table co-occur with more than one group of modals to a statistically significant level, clear examples of words are provided for the three groups.

Table 5.21: Tendencies to be used with three groups of modal verbs in the BNC: Academic sub-corpus.

	Frequency	<i>can, could, may, might</i>	<i>must, should, need, ought</i>	<i>will, would, shall</i>
Tokens in the sub-corpus	18,085,284	2.3%	0.8%	1.5%
<i>legitimately</i>	90	81.1%	-	-
<i>usefully</i>	155	70.3%	-	1.3%
<i>conceivably</i>	108	59.3%	-	1.9%
<i>easily</i>	1783	39.5%	0.7%	3.2%
<i>remembered</i>	324	3.1%	40.7%	6.8%
<i>noted</i>	2001	4.2%	17.3%	1.9%
<i>emphasised</i>	363	1.1%	16.8%	-
<i>stressed</i>	681	1.2%	10.6%	1.0%
<i>carefully</i>	754	2.0%	11.1%	2.9%
<i>surely</i>	700	3.9%	9.6%	8.6%
<i>suffice</i>	186	10.2%	9.1%	50.0%
<i>cease</i>	252	4.4%	8.3%	37.7%
<i>depend</i>	1184	8.4%	4.1%	35.5%
<i>disappear</i>	184	8.2%	1.6%	29.3%
<i>examine</i>	1487	3.6%	2.8%	22.5%
<i>argue</i>	1464	13.0%	0.3%	19.4%
<i>discuss</i>	971	5.5%	1.8%	17.7%

The examples which have been selected for the first group of modals include *legitimately* and *usefully* which occur with permission/possibility/ability modals very frequently indeed. Even though the figures for the other two examples are lower, they still reach levels of statistical significance. The strongest example for the second group is *remembered* which occurs with obligation/necessity modals more than 40% of the time. The other three examples for this group have lower figures, but well over one in ten of the instances co-occur with modals from this group, as opposed to less than one in a hundred for the tokens in the corpus overall. Two further examples of adverbs are provided for this group of modals, with around one in ten of the instances of *carefully* and *surely* co-occurring with obligation/necessity modals. Finally, examples of volition/prediction modals include verbs which may be used to describe the existence of something, and a second set of verbs which can be used to introduce or link to a topic. Since figures are calculated for specific types, the forms of these verbs preclude singular third person use without an accompanying modal. Nevertheless, if the learner has chosen to explore one of these types, the figures for the different groups of modals could still be of interest.

Figure 5.16 shows the examples provided on the help screen for modal verbs.

Graphs Tab: Modality

This shows the proportion of concordance lines which contain modal verbs within 4 words to the left of the main search word.

Modals are counted in three groups.

- *can, could, may* and *might*
- *must, should, need* and *ought*
- *will, would* and *shall*

Examples from the *BNC: Academic* sub-corpus

Less than 5% of words in the corpus are near modal verbs.

Yet words like *legitimately, usefully, conceivably* and *easily* are often used with the words *can, could, may* or *might*.

Words like *remembered, noted, emphasised* and *stressed* are often used with the words *must, should, need to* or *ought to*. Other words often used with these modals are *carefully* and *surely*.

Words like *suffice, cease, depend* and *disappear* are often used with the words *will, would* or *shall*. Other words often used with these modals are *examine, argue* and *discuss*.

Notes:

- It is a good idea to look at the concordance lines to see which modal verbs within each group are used most often.
- None of the words given as examples here are always used with modal verbs, but the proportions are higher than those of most other words.

Figure 5.16: Information provided on the Life Ring help screen for the *Modals* submenu on the Graphs Tab.

This help screen also has an additional note after the encouragement to look directly at concordance lines, stating: “It is a good idea to look at the concordance lines to see which modal verbs within each group are used most often.” At present, the word form of each modal is removed from the rows of nearby words in the database during the compression phase. However, it would be possible to add new functionality either in the SQL scripts or through a checking process of the concordances after they have been downloaded to provide information about the frequencies of specific modals. These could then be displayed in graphs like those of the Macmillan Dictionary (*Macmillan English Dictionary for Advanced Learners*, 2007) or the Longman Grammar of Spoken and Written English (Biber, et al., 1999), but with live data derived from the concordance lines. However, at present, the data are only provided in groups rather than for individual modals, and in order to

obtain this kind of information, users would need to search manually, use a specific modal in the query, or use the search inside feature (outlined at the end of this chapter).

5.2.3 Voice

In the explanation of the distribution of words and longer nested items, Hoey (2005) describes how these may be primed to occur in or avoid passive voice. The way passive forms are marked in the refactoring application is slightly more complicated than the process for marking modal verbs. First the application goes through all the words in the current sentence and puts a temporary marker next to any potential candidates for the passive form of the verb. Then, it works backwards through the sentence again, this time looking for a passive auxiliary or a form of the word “got”. Table 5.22 shows the *CLAWS* tags used for each of these.

Table 5.22: *CLAWS* tags used to identify passive voice

	C7 Tag Set	C5 Tag Set
Passive form of verb	VVN	VVN
Passive Auxiliary	VBDZ VBN VBZ VBI VBDR VBM VBR	VBD VBN VBZ VBI VBB
Got forms	got get gotten	got get gotten
Got POS tags	VV0 VVD VVI	VVB VVD VVI

If one of these is found within 4 words of the passive verb candidate, it is marked as a passive auxiliary. The application then goes through the sentence one more time and confirms the status of the passive form of the verb for those instances where a passive auxiliary exists and marks a 4 word window around these. In this way, each sentence can be marked as being passive voice and each word can also be marked as being in the proximity of a passive form.

Table 5.23 shows the overall proportions of tokens occurring in passive voice sentences in the *BNC: Newspapers* sub-corpus.

Table 5.23: Tendencies to be used (or not used) in passive voice sentences in the BNC: Newspapers sub-corpus.

		Frequency	Passive voice	Active voice / other
Tokens in the sub-corpus		10,809,050	22.7%	77.3%
associated	<i>prosecuted</i>	76	88.2%	11.8%
with police	<i>remanded</i>	244	85.2%	14.8%
actions	<i>discharged</i>	119	82.4%	17.6%
	<i>rewarded</i>	166	79.5%	20.5%
	<i>punished</i>	94	78.7%	21.3%
	<i>forgiven</i>	84	83.3%	16.7%
	<i>tempted</i>	125	82.4%	17.6%
	<i>debated</i>	67	76.1%	23.9%
	<i>understood</i>	460	75.7%	24.3%
actions	<i>clinched</i>	146	6.8%	93.2%
	<i>jumped</i>	415	9.2%	90.8%
states	<i>tired</i>	237	10.1%	89.9%
	<i>worried</i>	584	14.6%	85.4%
	<i>failed</i>	1,708	17.5%	82.5%

As can be seen, the first group of words, which intuitively can be strongly associated with police actions, demonstrate strong tendencies to be used in passive voice structures. This use of passive voice is a familiar example in language teaching: use where the agent is typically not directly mentioned. The second group of words includes *forgiven* and *tempted* which readily lend themselves to examples where the event is outside the power of the subject. This group also includes *debated* and *understood* which provide examples of how a topic can be foregrounded. The last two groups of words have strong tendencies not to be used in passive voice sentences with very expressive verbs like *clinched* and *jumped* as well as examples of a physical state (*tired*), a mental state (*worried*) and an outcome (*failed*).

Figure 5.17 shows the examples and information made available to users through the Life Ring help screen for this feature.

Graphs Tab: Voice

This shows the proportion of concordance lines which are passive voice.

To be counted, passive voice verbs must have a passive auxiliary verb (e.g. *is, was, got*).

Passive voice is usually associated with formal writing like academic articles, but we can see some differences in the kinds of verbs used in passive voice in newspapers.

Examples from the *BNC: Newspapers* sub-corpus

Less than a quarter of all words are in sentences which are passive voice.

Yet words like *prosecuted, remanded, discharged, rewarded* and *punished* occur more than three quarters of the time in passive voice sentences. These verbs are often associated with police actions and frequently occur in passive voice sentences.

Words like *forgiven, tempted, understood* and *debated* also occur very frequently in passive voice sentences.

Words describing actions like *clinched* and *jumped* typically do not occur in passive voice sentences. This is also true of words which describe states like *tired, worried* and *failed*.

Note:

- Interestingly, the overall proportions in the *BNC: Academic* sub-corpus are much higher with more than one third of sentences in passive voice.

Figure 5.17 Information provided on the Life Ring help screen for the *Voice* submenu on the *Graphs* Tab.

As Thompson points out, there would usually be a “complex web of reasons for choosing passive rather than active”, but these might include cohesive considerations for the choice of Theme (Thompson, 2004, p. 154). Alexander, Argent and Spencer (2008) argue that it is important for EAP students to know how passive voice constructions may be used in academic writing to make it possible for the Theme to carry information about the paragraph topic. Having one sentence before and one sentence after for each concordance line visible on the Cards Tab means that it is possible to see more context than the standard KWIC view, and this should facilitate the kind of exploration of how passivisation might be influenced by factors such as cohesion and the topical focus of the paragraph.

An area for future expansion of the software measures for English could be information about tense, aspect and mood, perhaps following the marking of time, *finite vs. non-finite* and *declarative vs. imperative vs. interrogative* from SFL (e.g. Thompson, 2004). There will be an inevitable trade-off between accuracy of mark-up and level of detail, and further consideration would be needed as to whether data on these features would be beneficial and useful for language learners.

5.2.4 Sentence Charge

Another feature introduced in Lexical Priming is that of affirmation or denial as being characteristic in the context of some words. In English there are several ways to deny something, and not all of these have a distinct POS tag in *CLAWS*. Table 5.24 shows the tags used in the approach implemented in *The Prime Machine*.

Table 5.24: CLAWS tags used to identify polarity

	C7 Tag Set	C5 Tag Set
Negative charge	XX	XX0

Words in a 4 word window around this tag are also marked as being in the proximity of a negative marker, so sentence or word counts for each node are stored, but only results derived from sentence level information are currently implemented in the client application.

Table 5.25 shows the overall proportions of tokens occurring in negative sentences in the *BNC: Academic* sub-corpus.

Table 5.25: Tendencies to be used in negative sentences in the *BNC: Academic* sub-corpus.

	Frequency	Negative	Positive
Tokens in the sub-corpus	18,085,284	16.9%	83.1%
<i>watertight</i>	15	73.3%	26.7%
<i>invalidate</i>	52	69.2%	30.8%
<i>necessarily</i>	2,014	69.1%	30.9%
<i>preclude</i>	128	65.6%	34.4%
<i>dissimilar</i>	114	64.9%	35.1%
<i>always</i>	4,879	34.5%	65.5%

These examples were chosen partly because of the very strong tendencies, and partly because each of them has a very strong meaning. In cases where the word *not* or some other form of negativity marker is not used, it is obvious that all of the words in this table have extreme meanings. Yet when they are used in academic writing, it is quite interesting to note how frequently they occur in sentences where it is likely that they are actually conveying a sense of openness, possibility or hedging. The figure for *always* is rather lower than the figures for the other words given in the table, but still more than one third of its occurrences are within sentences containing *not*.

Figure 5.18 shows the examples and information made available on the Life Ring for this feature.

Graphs Tab: Polarity

This shows the proportion of concordance lines where the sentence is negative.
 Negative sentences contain the word *not*.

Examples from the *BNC: Academic* sub-corpus

Less than 1 in 5 words occur in sentences containing the word *not*.

Yet words like *watertight*, *invalidate*, *necessarily*, *preclude* and *dissimilar* seem to occur quite frequently in sentences with the word *not*. The word *always* occurs in sentences containing the word *not* more than one third of the time. These frequencies are high given the overall low proportion of negative sentences and given that these words all seem to have strong meanings.

Figure 5.18: Information provided on the Life Ring help screen for the *Polarity* submenu on the Graphs Tab.

The results provided by this simple measure of sentence level co-occurrence with *not* may be helpful when giving feedback to Chinese learners on academic writing, as in the past I often felt the need to comment on their overuse of extreme adverbials such as *always*.

Looking at the first group of features of colligation, Table 5.26 shows how the high frequency and very high frequency items are extremely likely to be shown in the software as having at least one tendency. This is not surprising given the variety of relations which are measured, but even so the results give an indication that a fair proportion of items in a corpus with a frequency of more than 20 are likely to have at least one of these aspects worth exploring in more detail.

Table 5.26: Proportions of types at different frequency thresholds showing at least one tendency for use with the first set of features of colligation.

Colligation	≥ 3	≥ 20	≥ 100	≥ 1,000	≥ 10,000
<i>Hindawi Biological Sciences</i>	7.5 %	21.3 %	43.3 %	82.8 %	97.6 %
<i>Hindawi Computer Science</i>	9.3 %	23.6 %	44.6 %	78.4 %	100.0 %
<i>Hindawi Chemistry</i>	8.5 %	23.9 %	46.0 %	79.3 %	100.0 %
<i>Hindawi Maths</i>	10.6 %	26.5 %	49.3 %	82.1 %	99.2 %
<i>BNC Academic</i>	8.1 %	21.9 %	45.9 %	84.3 %	98.1 %
<i>BNC Newspapers</i>	6.3 %	17.3 %	38.9 %	83.8 %	100.0 %

5.2.5 Definiteness/indefiniteness

Determiners are on a separate menu from the other colligation features which have been introduced so far. They form a separate group with prepositions on the Graphs Tab for a number of reasons. Firstly, these are areas which are a constant struggle for intermediate and advanced language learners, so having a more prominent position on a separate menu could be helpful. Tsui (2004) argues that exploration of the use of definite versus indefinite articles is an area where teachers struggle to provide concrete and useful rules, and where access to concordance line data can be rewarding. Secondly, the co-occurrence of an item with a determiner or a preposition can be regarded in several different ways. On the one hand, the relationship can be considered to be part of collocation, with no division necessary between the way grammatical items and lexical items are treated in the collocation process (Hoey, 2003). On the other hand, by grouping certain kinds of determiner together, tendencies for items can be revealed, and both determiners and prepositions provide clues as to the grammatical class of an item. Finally, from a very practical point of view, having too many types of priming summary on a single menu would mean that not all submenus would be visible on a standard screen, so some divisions were needed.

The process in the refactoring application which marks articles and possessives is similar to that used for marking proximity to modal verbs at the word level. The actual word-form is also stored in the database row, but currently this is discarded during the database compression process. As with modal verbs, markers of definiteness and indefiniteness only affect data held for words on the right-hand side within a window up to 4 words. Unlike modal verbs, no sentence level information is stored for these. Table 5.27 shows the *CLAWS* tags used.

Table 5.27: *CLAWS* tags used to identify articles and possessives

	C7 Tag Set	C5 Tag Set
Articles and Possessives	AT AT1 APPGE GE	AT0 DPS POS

Further processing to distinguish between definite and indefinite determiners takes place in the compression phase. The forms *a*, *an*, *every* and *no* are marked as indefinite and all others are grouped as definite articles and possessives. This is an oversimplification, and does not include all the non-specific deictics listed by Halliday and Matthiessen (2004, p.

315) but should cover many of the common patterns for English and the main articles which language teachers tend to focus on in teaching and feedback.

Table 5.28 shows the overall proportions of tokens near articles and possessives in the *BNC: Academic* sub-corpus.

Table 5.28: Tendencies to be used with two groups of articles and possessives in the *BNC: Academic* sub-corpus.

		Frequency	Definite article or possessive	Indefinite article
Tokens in the sub-corpus		18,085,284	25.5%	8.2%
superlative	<i>biggest</i>	121	100.0%	-
adjectives	<i>widest</i>	63	100.0%	-
	<i>slightest</i>	53	100.0%	-
	<i>broadest</i>	62	98.4%	-
	<i>earliest</i>	473	97.7%	0.2%
	<i>finest</i>	70	97.1%	-
	<i>richest</i>	75	97.3%	-
	<i>largest</i>	793	97.2%	-
	<i>poorest</i>	142	97.2%	-
	related to quantities	<i>handful</i>	123	10.6%
<i>lot</i>		719	11.3%	84.8%
<i>variety</i>		2,337	16.3%	74.2%
<i>dozen</i>		123	16.3%	72.4%
objects of academic research	<i>survey</i>	2,226	48.4%	30.7%
	<i>questionnaire</i>	488	40.0%	43.0%

The first group in the table are superlative adjectives, and as would be expected, these have some of the strongest tendencies to occur with definite articles or possessives. The second group are words which are related to quantities and show strong general tendencies to be used with indefinite articles. These figures will be in no way surprising, but it is worth remembering that like all the measures described in this chapter the aim is to highlight to the user any tendencies which exist for the specific words that they have used for a search. The last group of words in Table 5.28 are *survey* and *questionnaire* and they are included to provide points of contrast with the results from the *BNC: Newspapers* sub-corpus which are provided in Table 5.29.

Table 5.29: Tendencies to be used with two groups of articles and possessives in the *BNC: Newspapers* sub-corpus.

	Frequency	Definite article or possessive	Indefinite article
Tokens in the sub-corpus	10,809,050	24.7%	9.0%
<i>survey</i>	1096	41.7%	46.2%
<i>questionnaire</i>	25	28.0%	60.0%

While in the academic texts, *survey* occurs with definite articles or possessives more frequently than with *indefinite* articles, in newspapers the balance shifts the other way. However, while indefinite articles are used more often than definite articles or possessives with the word *questionnaire* in both sub-corpora, the figure for indefinite articles in newspapers is much higher. It seems reasonable to suggest that news stories may use a survey or questionnaire as a way of introducing some points from recent research, but may be less interested in describing the same survey again later in the same story.

Figure 5.19 shows the examples and information provided to users on the Life Ring help screen for definite and indefinite articles and possessives.

Graphs Tab: Definite/Indefinite

This shows the proportion of concordance lines where there is an article or possessive within 4 words to the left of the main search word.

They are grouped in this way:

- Definite articles (*the*) or possessives (e.g. *my, your, 's*)
- Indefinite articles (*a, an, every, no*)

Examples from the *BNC: Academic* sub-corpus

Around a quarter of all words are near definite articles or possessives.

Words ending in “-est”, usually follow the, with 97% or more of the occurrences of the words *biggest, widest, slightest, broadest, earliest, finest, richest, largest* and *poorest* near definite articles or possessives.

Less than 1 in 10 of all words are near indefinite articles.

However, words like *lot, handful, variety* and *dozen* usually follow a, with 72% or more of the occurrences near indefinite articles.

Note:

- Obviously, you can get a sense of how often a word is used as a noun by looking at these figures.

Figure 5.19: Information provided on the Life Ring help screen for the *Articles and Possessives* submenu on the Graphs Tab.

5.2.6 Prepositions

Finding the right preposition for a verb and remembering to use prepositions is a particular challenge for Chinese learners of English. However, at the time of developing the import and mark-up procedure for priming features, the means of storing and highlighting collocations in *The Prime Machine* had not been finalized. Since *CLAWS* was marking prepositions and since they form a (relatively) closed set, it seemed reasonable to use up two more columns in the database to store this information. Table 5.30 shows the relevant tags from *CLAWS*.

Table 5.30: CLAWS tags used to identify prepositions

	C7 Tag Set	C5 Tag Set
Prepositions	IF II IO IW	PRP PRF

As will be explained in Section 5.5.1, the priming tendencies are used to provide visual prompts to language learners using the software, so although co-occurrence with prepositions may be considered part of collocation, having these tendencies stored in the system means that an additional prompt to explore nearby prepositions can be given when this could be fruitful. Having the proximity to prepositions available as a filter also means it is quick and easy to filter the results to show only those with prepositions or only those without. The filtering function is explained in Section 5.6 below.

Table 5.31 and Table 5.32 show the overall proportions of tokens near prepositions in the *BNC: Academic* sub-corpus and the *BNC: Newspapers* sub-corpus. The tables show figures for two words which are often muddled up by Chinese learners and highlights differences in the tendencies they have to be used with prepositions.

Table 5.31: Examples for tendencies to be used with (or without) prepositions in the *BNC: Academic* sub-corpus.

	Frequency	Prepositions	No prepositions
Tokens in the sub-corpus	18,085,284	58.0%	42.0%
<i>spite</i>	476	98.7%	1.3%
<i>despite</i>	2,750	-	100.0%

Table 5.32: Examples for tendencies to be used with (or without) prepositions in the *BNC: Newspapers* sub-corpus.

	Frequency	Prepositions	No prepositions
Tokens in the sub-corpus	10,809,050	52.3%	47.7%
<i>spite</i>	279	98.9%	1.1%
<i>despite</i>	2,261	-	100.0%

As can be seen in both these tables, the preposition measure clearly distinguishes the way in which *spite* and *despite* are used in sentences. The help screen for this feature is shown in Figure 5.20.

Graphs Tab: Prepositions

This shows the proportion of concordance lines where there is a preposition within 4 words either side of the main search word.

Prepositions include:

- General prepositions (e.g. *at, on, by*)
- *for, of, with* or *without* as prepositions

Examples from the *BNC: Academic* and *BNC: Newspapers* sub-corpora

A little more than half of all words in these corpora are near prepositions.

Yet 99% of the occurrences of the word *spite* are near prepositions while none of the occurrences of the word *despite* are near prepositions.

Sometimes similar words can be quite tricky to use correctly when writing in a foreign language, but a quick search for *despite* vs. *spite* in either of these corpora can show preposition patterns very clearly. We would expect the concordance lines to show us *despite* near verbs and in the phrase “despite the fact”. We would also expect to see *spite* used in sentences in the phrase “in spite of”.

Figure 5.20: Information provided on the Life Ring help screen for the *Prepositions* submenu on the *Graphs Tab*.

The measure for prepositions is very similar to a measure of collocation. Indeed, strong collocations containing prepositions will also show up in the summary tables as having high proportions of occurrence in the proximity of prepositions.

Table 5.33 provides a list of some node words which have *of* stored as one of their collocates and which have very high proportions of occurrences near prepositions. Some other high frequency prepositions are shown in the right-most column if these are also stored as collocations.

Table 5.33: Examples for tendencies of collocates of *of* to be used with prepositions in the *BNC: Academic* sub-corpus, along with the proportions accounted for by collocations containing *of*.

		Frequency	Prepositions	<i>of</i>	Others
Tokens in the Academic sub-corpus		18,085,284	49.7%		
beginning or presence	<i>advent</i>	188	100.0%	97.3%	<i>with</i>
	<i>outbreak</i>	165	96.4%	86.1%	
	<i>commencement</i>	124	96.0%	83.9%	<i>at</i>
	<i>onset</i>	413	95.2%	71.7%	
	<i>presence</i>	2,485	94.3%	67.9%	
	<i>aftermath</i>	139	94.2%	80.6%	<i>in</i>
	<i>dissemination</i>	159	90.6%	61.0%	
	<i>conception</i>	1,129	90.3%	71.7%	
	<i>beginnings</i>	160	90.0%	59.4%	<i>from</i>
	<i>adoption</i>	581	89.5%	43.5%	
absence or destruction	<i>irrespective</i>	361	100.0%	100.0%	
	<i>regardless</i>	372	98.7%	97.8%	
	<i>absence</i>	2,232	96.7%	79.7%	<i>in</i>
	<i>abandonment</i>	152	94.7%	77.0%	
	<i>breaches</i>	161	93.8%	71.4%	
	<i>breach</i>	1,844	93.5%	72.9%	<i>for</i>
	<i>removal</i>	549	93.1%	66.5%	<i>from</i>
	<i>demise</i>	181	91.2%	58.0%	
	<i>destruction</i>	461	89.6%	52.7%	
range or number	<i>sorts</i>	425	98.8%	86.8%	
	<i>kinds</i>	1,569	98.6%	80.4%	
	<i>lots</i>	110	98.2%	90.0%	
	<i>kind</i> ⁴³	4,267	97.1%	≥61.5%	
	<i>amounts</i>	762	96.5%	48.4%	
	<i>sort</i>	1,858	95.7%	69.6%	
	<i>variety</i>	2,337	95.5%	85.0%	
	<i>handful</i>	123	95.1%	91.9%	
	<i>number</i>	10,648	94.5%	84.0%	
	<i>plenty</i>	191	94.2%	82.7%	
	<i>proportion</i>	2,484	93.9%	77.4%	
	<i>aspects</i>	2,776	93.6%	82.9%	
	<i>combination</i>	1,201	93.3%	68.7%	<i>with, by</i>
	<i>series</i>	2,814	92.3%	64.7%	
	<i>subset</i>	139	92.1%	71.2%	
	<i>mixture</i>	474	91.6%	66.5%	<i>with</i>
	<i>parts</i>	2,765	90.4%	59.9%	
<i>aspect</i>	1,391	90.2%	75.3%		
<i>stages</i>	1,290	89.7%	42.2%	<i>at, in</i>	

⁴³ The proportion of cases collocating with *of* exceeds the figure shown because 39.0% of occurrences are for the collocation *of.. kind*, while 61.5% are for the collocation *kind of*. However, some of these may coincide as in the phrase *of the kind of*.

It is clear that in these cases, the measure of proximity to prepositions would be drawing attention to a relationship which is also demonstrated through the collocation measure. Nevertheless, for some words the difference between the proportion with any preposition and the proportion for its collocation with *of* is quite large, yet most of these words do not have any other common prepositions listed as collocates.

Table 5.34 shows proportions of items displaying at least one tendency, and as can be seen many of these figures are very high.

Table 5.34: Proportions of types at different frequency thresholds showing at least one tendency for use with the second set of features of colligation

Determiners and Prepositions	≥ 3	≥ 20	≥ 100	≥ 1,000	≥ 10,000
<i>Hindawi Biological Sciences</i>	14.6 %	40.2 %	72.0 %	92.8 %	98.0 %
<i>Hindawi Computer Science</i>	18.3 %	45.3 %	76.7 %	95.5 %	100.0 %
<i>Hindawi Chemistry</i>	16.9 %	46.4 %	78.4 %	94.1 %	100.0 %
<i>Hindawi Maths</i>	22.5 %	55.1 %	84.6 %	95.7 %	98.4 %
<i>BNC Academic</i>	16.8 %	42.6 %	74.0 %	93.3 %	98.7 %
<i>BNC Newspapers</i>	16.2 %	41.7 %	71.3 %	92.7 %	98.0 %














The tendency for items to co-occur with determiners and prepositions will be strongly influenced by word class and this is probably one of the main reasons for these figures being higher than the figures for most of the other features. As determiners and prepositions are two important areas of difficulty in language teaching, these high figures suggest that the results for a very large range of items when looked at individually ought to demonstrate some interesting differences.

5.2.8 Icons to represent colligations

The icons used to represent colligation patterns are a little less intuitive than those used for positions. As can be seen in Table 5.35, capital letters are used as before to indicate the category, and green or orange elements represent different features. The choice of elements was to some extent constrained by the range of elements available in the icon design software library. Cogs were used to represent simple or complex sentences. For modals, a heart represents “Volition/Prediction”, a key represents “Permission/Possibility/Ability” and a crossed through circle represents “Obligation/Necessity”. Passive voice is slightly more oblique and is represented by a hand

and a person standing still, while sentences which are not passive are indicated by a worker with a shovel. Polarity or sentence charge is represented by “Y/N” and a tick or a cross. Definite articles and possessives are indicated by a target symbol, while indefinite articles are shown with a question mark. Since “P” is already used to indicate tendencies associated with position in paragraph, “Prep” is used for prepositions, with a tick mark shown in green or orange.

Table 5.35: Icons representing features related to colligation

 <p>Simple sentences;</p>	 <p>Complex sentences.</p>	
 <p>Volition/Prediction modals;</p>	 <p>Permission/Possibility/Ability modals;</p>	 <p>Obligation/Necessity modals.</p>
 <p>Active voice/other;</p>	 <p>Passive voice.</p>	
 <p>Positive sentences;</p>	 <p>Negative sentences.</p>	
 <p>Near definite articles or possessives;</p>	 <p>Near indefinite articles.</p>	
 <p>Near prepositions;</p>	 <p>Avoids prepositions.</p>	

5.3 Self-Repetition

The tendency of a word to collocate with itself is dealt with during the calculation for collocations on the server. Since all instances of each possible node are retrieved during this process, it is efficient to perform a count of the number of occurrences of the word in each text at this point, rather than in the SQL script which is used to calculate and store other priming tendencies for individual words. The stem information for each word in the lexicon is pre-calculated and this is described in Chapter 3. The default minimum number of occurrences of a word or stem in each text for it to be counted as demonstrating a tendency for repetition is 3. Contingency tables for repetition are based on measuring the number of instances and the text lengths in which the word occurs repeatedly against the number of instances and the text lengths in which the word is not repeated or occurs less than 3 times. It is essentially a dispersion measure, but follows the same log-likelihood and Bayes factors process as the other features for consistency. Figure 5.21 shows the information provided on the help screen for this feature.

Graphs Tab: Repetition

This shows the proportion of concordance lines where the main search word occurs more than twice in the same text.

Many words occur just once or twice in the whole corpus, but if just the words which occur three or more times are counted we can see the following results:

In the *BNC: Newspapers* sub-corpus, more than 3 out of 10 of all the different kinds of words seem to occur in the same text more often than might be expected by chance.

In the *BNC: Academic* sub-corpus, more than half of all the different kinds of words seem to occur in the same text more often than might be expected by chance.

These figures suggest that there are some words which can be repeated several times in a text quite naturally, while other words may stand out more obviously if you use them too often.

Notes:

- This measure is different from the other features on the Graphs Tab. The percentages shown here show the proportion of words in the lexicon (types).
- It is always a good idea to look at the concordance lines to see what patterns of priming seem to occur, but since some texts are very long, you may not be able to see the repetition very clearly in the results.
- Since repetition has been shown to be one form of cohesion, you could get a sense of how a word contributes to this by using the [Lexical Cohesion Bonding](#) ranking method which is available on the Options page. You may also notice that many of the top lines for this ranking method come from the same text.

Figure 5.21: Information provided on the Life Ring help screen for the *Repetition* submenu on the Graphs Tab.

When the tendency of types to be repeated is explored across different frequency thresholds, as can be seen in Table 5.36, the pattern which emerges is noticeably different from the tendencies which were reported for the other features.

Table 5.36: Proportions of types at different frequency thresholds showing a tendency for repetition of form or stem.

Repetition	≥ 3	≥ 20	≥ 100	≥ 1,000	≥ 10,000
Hindawi Biological Sciences	58.6 %	91.2 %	94.0 %	89.2 %	61.0 %
Hindawi Computer Science	60.5 %	89.0 %	90.1 %	83.5 %	41.3 %
Hindawi Chemistry	51.1 %	81.5 %	84.4 %	78.5 %	12.8 %
Hindawi Maths	62.2 %	90.9 %	92.1 %	84.5 %	49.6 %
BNC Academic	52.0 %	80.3 %	85.4 %	82.3 %	60.3 %
BNC Newspapers	31.9 %	56.4 %	65.4 %	56.3 %	15.3 %

Whereas for the other sets of results (Table 5.9, Table 5.14, Table 5.26 and Table 5.34) the higher the frequency threshold the greater the proportion of types, for repetition the figures tend to peak somewhere between the 20-100 range, with lower proportions for very high frequency items. Given the differences in the average lengths of the texts, the marked differences between the proportions for the *BNC: Newspapers sub-corpus* and the other corpora are not surprising. Nevertheless, these results provide an interesting contrast with previous studies which have looked at “burstiness” (Kenneth W. Church & Gale, 1995; Katz, 1996; both cited in Madsen, Kauchak, & Elkan, 2005) and the use of dispersion plots (Scott & Tribble, 2006). These results also suggest that further research into tendencies of words of various frequency levels to be repeated or not to be repeated could be worthwhile.

As can be seen in Table 5.37, the icons representing repetition are fairly simple. The “R” indicates repetition, with a chain link representing simple repetition and a spider’s web representing repetition by stem for individual words or by node word for collocations.

Table 5.37: Icons representing a tendency for repetition

 <p>Repetition of the same form</p>	 <p>Repetition of the same stem</p>
--	--

5.4 Storage and retrieval of data

Refactoring the data and storing them in a database is a costly operation in terms of customization and time. An alternative would have been to design the server or client application to work with XML files directly, and use XML search tools with complicated regular expressions to count specific features and add these up. If the goal had been to develop a robust and highly flexible system for researchers, and if users would have access to a super-computer cluster, this might have been something which could have been developed. The goal for this project was rather different, and some of the reasons for working with relational databases in the multi-tier architecture have been explained in Chapter 3. The costs of this are that the flexibility is lost, but the gains are that the computer specifications are less demanding and calculating each of the features across the whole corpus is relatively fast. Table 5.38 provides an overview of the automatic processes involved in taking raw text files and preparing them for the scripts which calculate the tendencies of words and collocations to occur in the range of environments that have been outlined in this chapter.

Table 5.38: Overview of automatic processing

	Process
1.	Tags are converted from SGML to XML tags/milestones
2.	Apostrophe characters are “disambiguated” to represent a true apostrophe or the opening or closing of single quotation marks ⁴⁴ .
3.	Titles, headings and sections are changed into milestone XML tags which the post- <i>CLAWS</i> processing system can read.
4.	Raw text is “cleaned” and stretches of text are marked with <text> tags for <i>CLAWS</i> to process.
5.	The <i>CLAWS</i> output file (vertical format) is opened in the post- <i>CLAWS</i> process and the marked up text is lined up with the XML tags in the <i>Supp</i> file.
6.	The following information is added at the word level: <ul style="list-style-type: none"> • Most likely POS tag with likelihood as marked by <i>CLAWS</i> • Case (upper/lower/title/other) • Position as proportion of words in sentence • Position in Theme or Rheme • Quote/bracket level • Proximity to: <ul style="list-style-type: none"> ◦ Articles or possessives (left) ◦ Modals (left) ◦ “Not” (left or right) ◦ Passive (left or right) ◦ Prepositions (left or right)
7.	The following information is added at the sentence level: <ol style="list-style-type: none"> 4. Sentence position in text 5. Sentence charge (if “not” appears anywhere in sentence) 6. Sentence modality (if any modals appear) 7. Sentence voice (if any part in passive voice) 8. Sentence complexity
8.	Metadata is added into the database file on several different levels ⁴⁵ : <ul style="list-style-type: none"> • Information about the corpus • Information about the text • Information about the author(s) • Information about the sections
9.	All these data are converted into SQL insert statements so they can be imported into the database quickly.
10.	SQL scripts compress the information through creation of one-to-many tables, summary tables and columns to hold important information about the environment of each sentence and each word.

The database schema before and after compression is shown in Figure 5.22 below.

⁴⁴ Further information about this is provided in Chapter 3. Currently, the quotation level information is not used in later priming feature processes.

⁴⁵ This process is discussed further in Chapter 6.

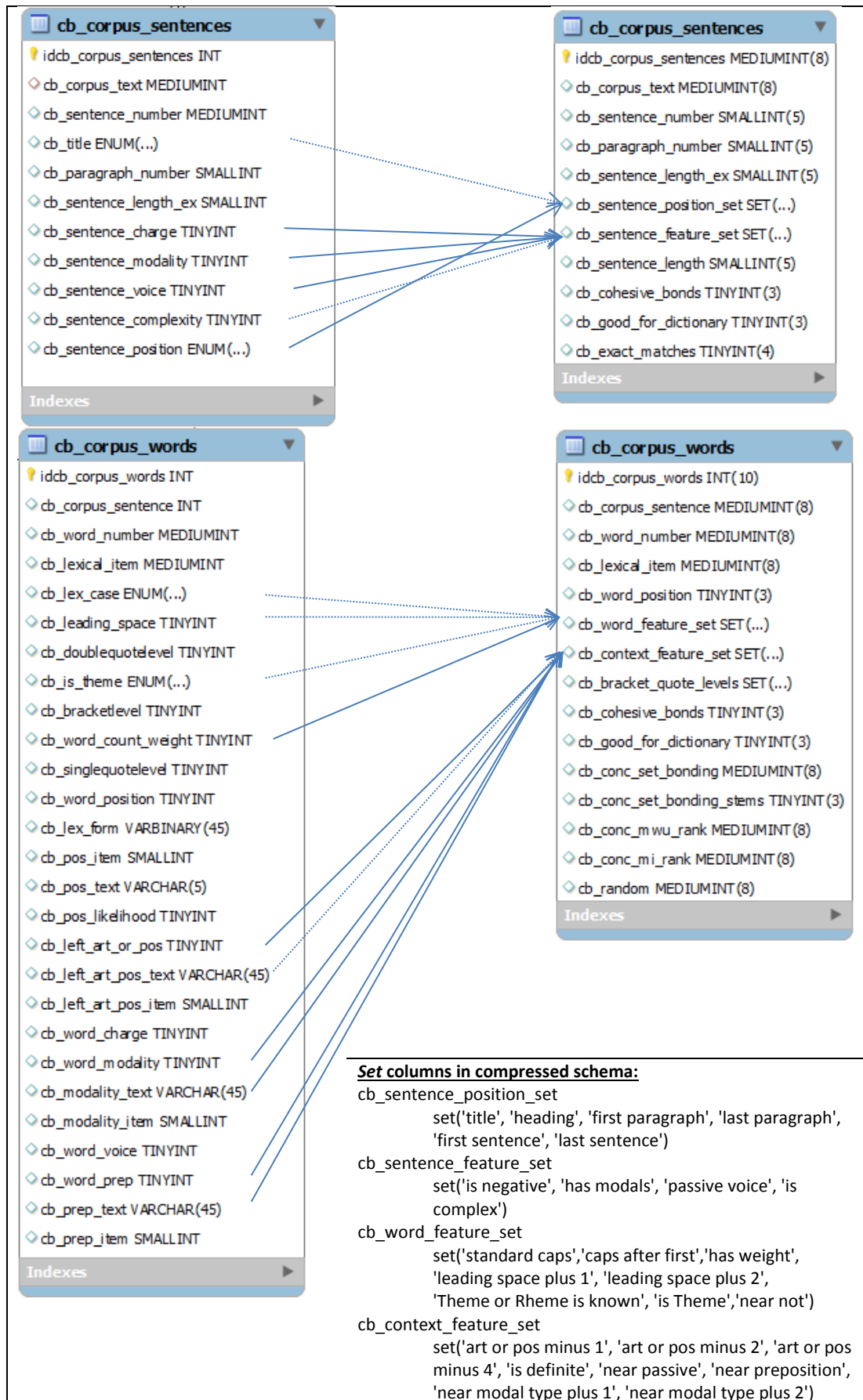


Figure 5.22: The database schema for the uncompressed tables for sentences and words (left), and for the compressed tables (right).

As can be seen, information about position and colligation is transformed into columns in the database which are of the *set* variable type. This is a column type in *MySQL* which allows the storage of 8 bits of information in one byte, with each element in the *set* having an on or off state. In the SQL scripts and in the client application these sets can conveniently be accessed using either bit-wise logical operators or a string composed of the “on” elements. During the compression stage, tables are created to hold the primary key values from the lexicon for the word forms for nearby modal verbs, articles or possessives, and prepositions, but since these are not used in the final application this information is discarded after the type of modal verb and non-definite articles have been detected and stored in the *set* columns. Two of the *set* columns in the database hold information about the environment for each sentence, and two about the environment for each word. It would have been possible to hold the information about less frequent priming features in a separate table, and a storage size (and therefore speed) improvement could be obtained if the size of the primary key for the sentence or word table were smaller than the number of bytes saved per row. In this way, each aspect of the environment could have been held in separate columns which would make updating and accessing the data a little more straightforward. Since, however, the client application uses information about the environment for the downloaded concordance lines in both the graph displays and in the filtering options (which are described in Sections 5.5.2 and 5.6 below), this would have meant that further complicated (and potentially slow) join statements would have been needed in order to retrieve all the information.

Tendencies for words and collocations to occur in each of the environments are held in summary tables in the database for all items reaching the Approximate Bayes Factor threshold of “positive evidence” (see Chapter 4). Figure 5.23 shows the table schema which links to the lexicon using its primary key.

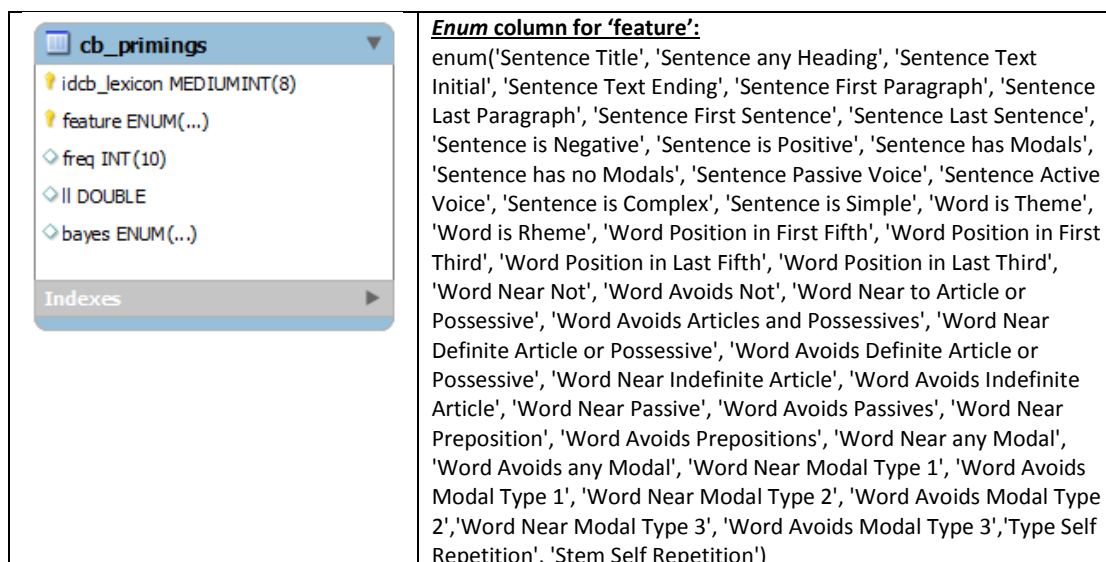


Figure 5.23: Schema for the `cb_primings` table⁴⁶

Tables for collocations are similar, with primary keys composed of columns for each lexical item in the collocation and the collocation type.

5.5 Ways in which the results are displayed

Storing information about the Lexical Priming features for each concordance line in the database table for each word (token) means that this information can be displayed to the learner as they use the software. The calculations explained above will, of course, provide summaries and other indications, but one of the important goals was to find a way to make more prominent the tendencies of words to occur in contexts related to features of Lexical Priming. The way in which the Cards Tab and Lines Tab provide formatting information indicating the position in paragraph has already been explained. In this section, some other design features related to the Graphs Tab will be introduced.

5.5.1 Hot Icons

The reason for pre-calculating results for the features introduced in this chapter and storing them as summaries in the database is not just to improve the experience of the learner in terms of speed of delivery; one of the key principles of the concordancer is that it should be able to *guide* the learner to find interesting and useful patterns. A researcher who is highly motivated to explore exhaustively the evidence for primings of a particular word or phrase

⁴⁶ The *Enum* labels are not visible in the software itself and “Active Voice” corresponds to “Active voice/other”.

based on tendencies revealed through corpus analysis may well be motivated enough to spend time trying different features, not losing too much interest if no relationship is found. However, if a vast array of options is made available to learners without any guidance, they could either waste time filtering the data or become frustrated. Therefore, a means was needed of helping direct the user's attention to priming information which might be explored more fruitfully, and this is the purpose of the "hot" icons. The same icons for each priming environment used for the submenus are also stored in higher resolution, and placed on a dock at the bottom of the screen. The *TAdvSmoothDock* component (TMS_Software, 2011) is designed to work like the icon dock on *Macintosh* operating systems, providing a horizontal list of icons which grow in size in a wave-like manner when the mouse cursor is moved over each one. When each list of concordance lines and other summary data are retrieved, the application goes through the table of statistically significant priming environments and changes the icons to match the features. Icons representing priming environments which do not reach the "Positive evidence" BIC Factor Score for the current search term are set to be invisible. In this way, the dock typically shows 3 to 5 icons for each search, and hovering over an icon makes it grow to a size where the details become more obvious and an additional heading appears. Clicking on the icon takes the user directly to the sub-section on the Graphs Tab menu corresponding to this feature. Figure 5.24 shows the Lines Tab, with the currently selected line visible to the right as a card, and the dock at the bottom showing statistically significant tendencies for position, complexity, indefinite articles and repetition. Figure 5.25 shows how the icon grows in size when the mouse is hovered over it.

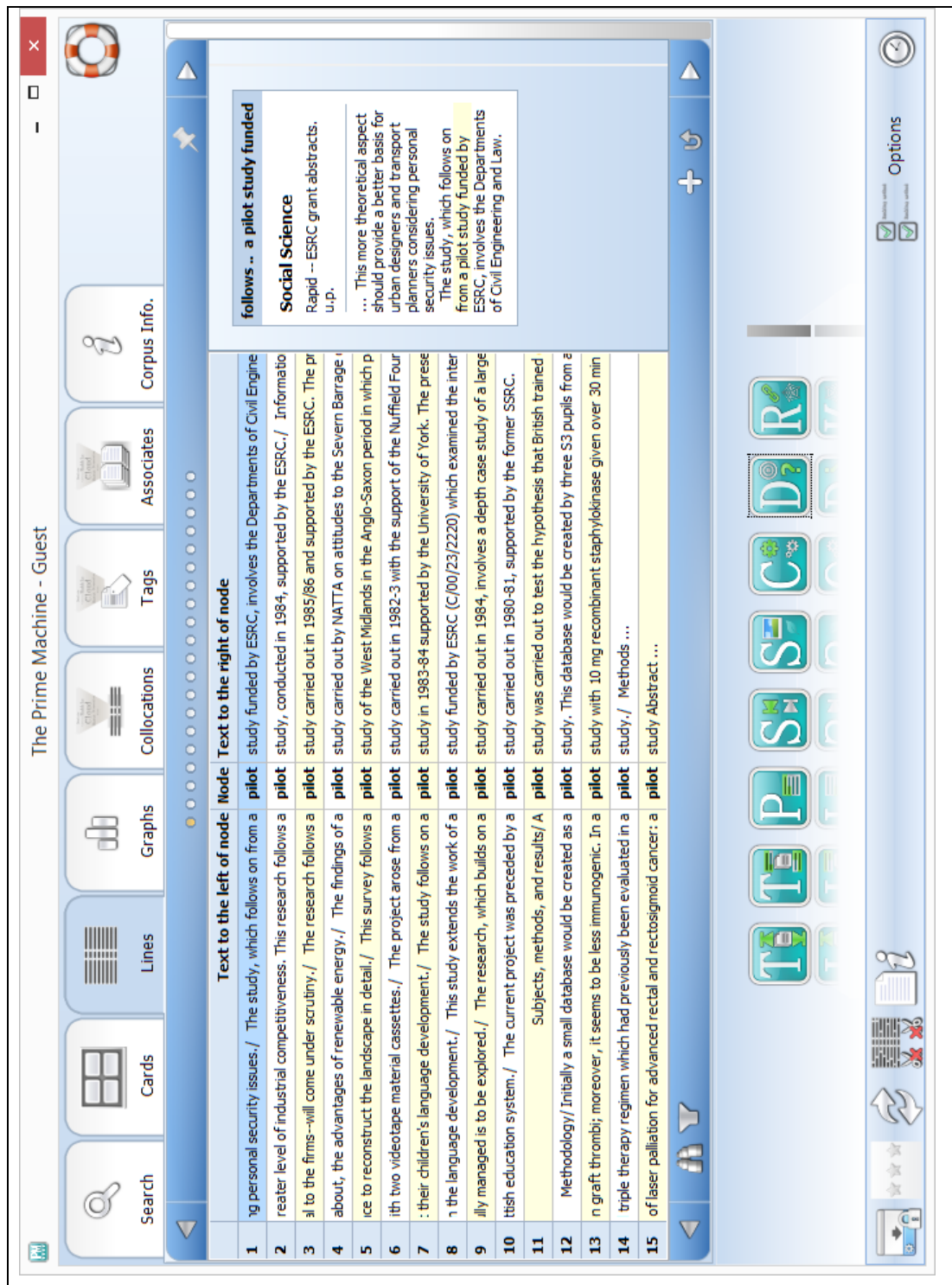


Figure 5.24: Lines Tab showing the card for the currently selected concordance line and the dock of icons for the node *pilot* in the *BNC: Academic* sub-corpus.

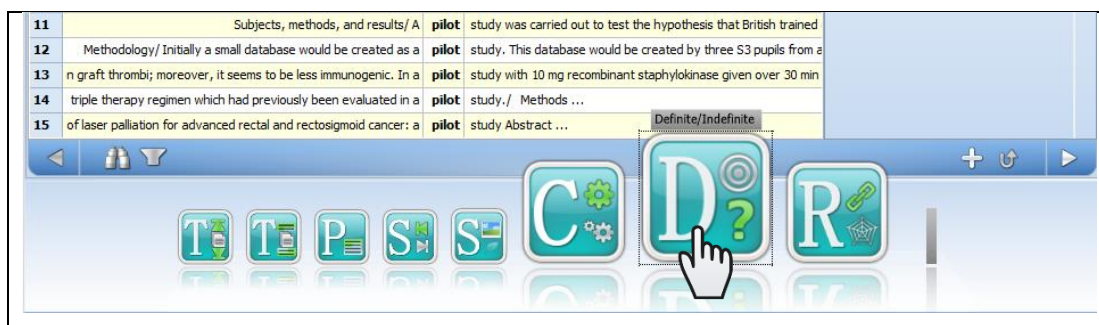


Figure 5.25: Enlarged icon showing positive evidence for a tendency to occur after indefinite articles. The hand icon represents the mouse cursor position.

5.5.2 Graphs

Whether accessed by finding the category and priming feature through the list of menus on the Graphs Tab, or clicking on a “hot” icon on the dock, information about the primings is displayed in the form of graphs. The aim of the graph is to present information in a visual way and to help the learners appreciate that these primings are almost always representative of relative frequencies rather than absolute restrictions on use.

Krishnamurthy and Kosem (2007) make many suggestions about the visual design of a corpus tool and the incorporation of icons and graphs into *The Prime Machine* was in part a response to these.

Since the default number of concordance lines to be retrieved is set to be at least 20, and since all the priming environment information for each concordance line is retrieved in the *set* columns described above, the client application can iterate through each concordance line and generate subtotals for each environment. For words or phrases which have a frequency above 20, however, it could be helpful for the user to know representative the sample which has been downloaded is, compared to the tendencies for the word or collocation to occur in the specific environment across the whole corpus. Therefore, as priming information is calculated, a summary table is also produced providing information about the proportion of instances of each node which match the features. This is done for all words and collocations which have a frequency greater than 20. The data from these summary tables are automatically downloaded with the concordance lines, so any node which has a frequency greater than 20 shows bars on the graph for both the current set of concordance lines and the entire corpus. This is particularly useful when the ranking method may have influenced the kinds of environment which were present in the downloaded results. Figure 5.26 (on the following page) shows the graph data for the node *pilot* ranked by fixed random order and ranked by the log-likelihood collocation and

concordance bonding rank method. When two search terms are compared side by side using the “compare” mode, the graph shows the data for both, providing an easy way to see any evidence for possible differences in their primings. As shown in Figure 5.27 below, the display does become fairly complex, and a small enhancement was made to the graph drawing component to permit a multi-column legend.

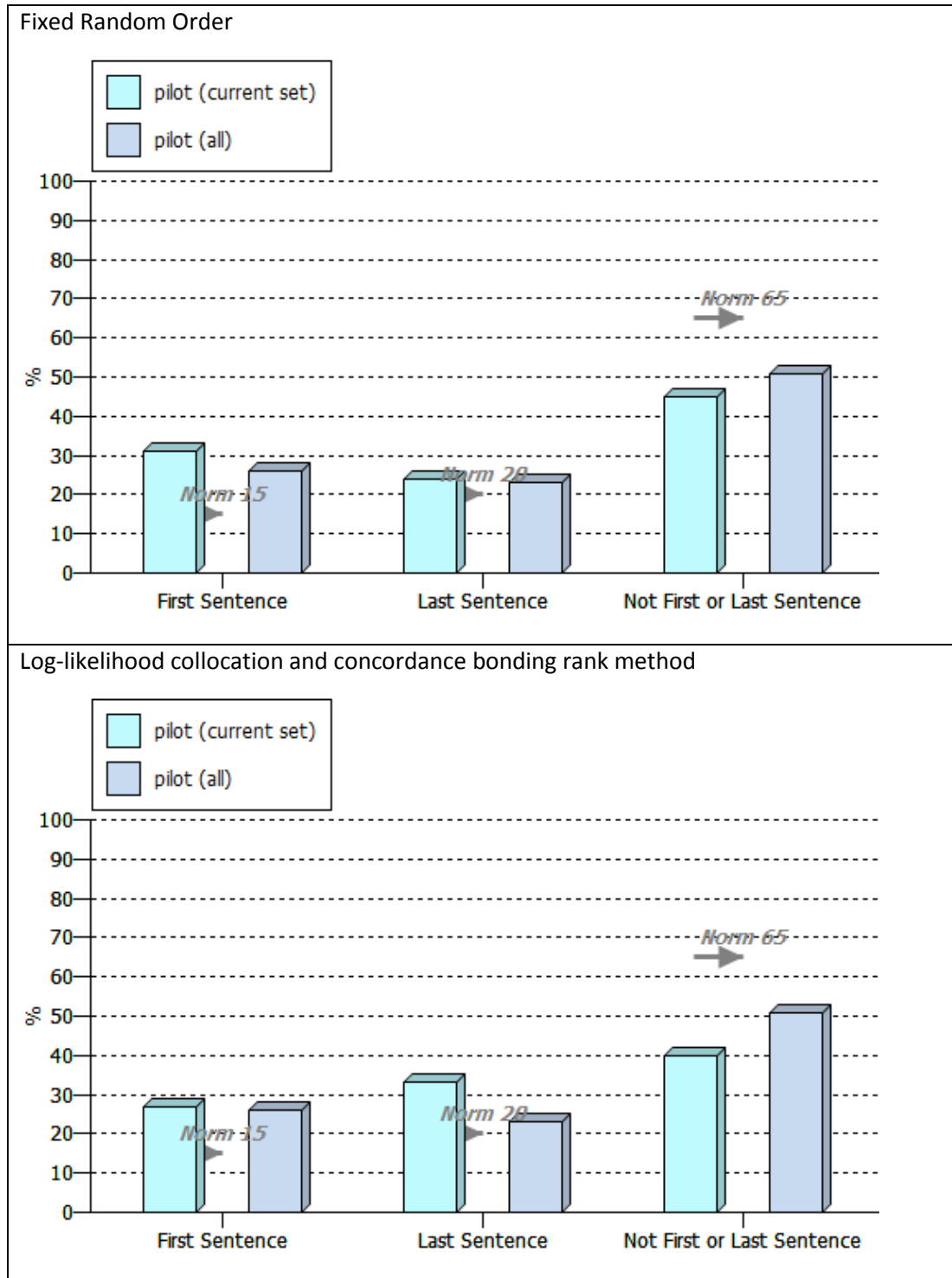


Figure 5.26: Graphs for the *Paragraph Position* submenu on the Graphs Tab for the node *pilot* in the *BNC: Academic* sub-corpus, with fixed random order (top) and the log-likelihood collocation and concordance bonding rank method (bottom).

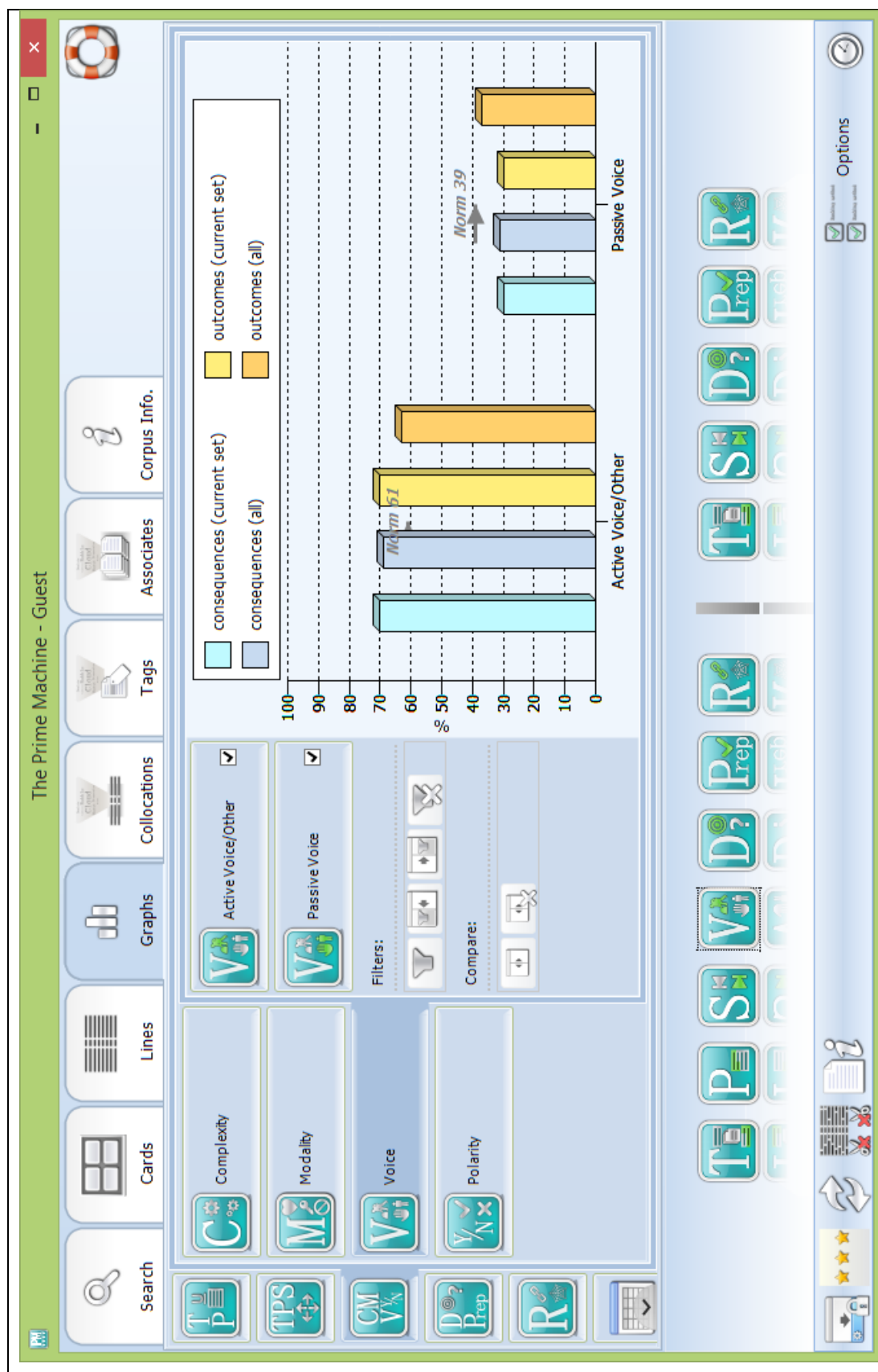


Figure 5.27: Graph display for compare mode for the Voice submenu on the Graphs Tab with results for consequences compared against outcomes from the BNC: Academic sub-corpus.

One of the striking things from Hoey's (2005) presentation of the evidence for the priming of words is the need to consider what the expected values or what typical environments for each kind of feature would be. Clearly, the number of text initial sentences will always be very small compared to the whole corpus, yet because of the differences in the length of the texts in different corpora, these proportions can vary. Similarly, some features such as passive voice tend to be much less common in some text types than in others and so it is useful to be able to highlight cases where the proportion is much higher or lower than would be expected based on a collection of texts as a whole. As explained above, priming tendencies are calculated using log-likelihood and this provides a way to balance the pervasiveness of feature against the frequency of different words. Percentages are rather easier for learners to understand, but since the expected proportions vary considerably between feature and across different corpora, some way of indicating whether the bar was above or below the expected value for each specific feature was needed. The *TAdvCharts* component from *TMS Controls (TMS_Software, 2013)* which was used to create the graphs in the *Delphi* programming environment includes arrow markers as a chart type which can be superimposed on top of other charts. Since the number of words in each priming environment is calculated and stored in the script which creates summary data for primings for each corpus, this table is also available in the fully optimized database and is automatically downloaded when the user selects a corpus. The values for the arrows indicating the expected values are therefore taken from this table. Some consideration was given as to how best to label these arrows. Using a formal statistical term may be off-putting or confusing for learners, but at the same time if they were labelled "expected" this could be quite misleading. The word "norm" was selected as although it is a formal statistical term, it is similar to the very common English word "normal". However, it should be noted that the "norm" on the graph is the proportion of instances which would be expected if the feature was distributed evenly; they are not the "norms" from Hanks' theory of norms and exploitations (Hanks, 2013). This is not a deliberate blurring; the "norm" is a question of perspective. These "norms" can provide insights into both the tendencies of words and also any unevenness in the sample of concordance lines which have been downloaded. For example, in Figure 5.27 it can be seen that the overall pattern for passive voice for *outcomes* is very close to the norm for this feature, but the selection of concordance lines which have been downloaded are actually different from this overall tendency and more similar to the pattern for *consequences*. The norm values are not just different for each set of features; they are also different across different corpora. Figure

5.28 shows the graphs for the node word *pilot* in two different corpora and, as can be seen, the positions of the norm arrows can be quite different.

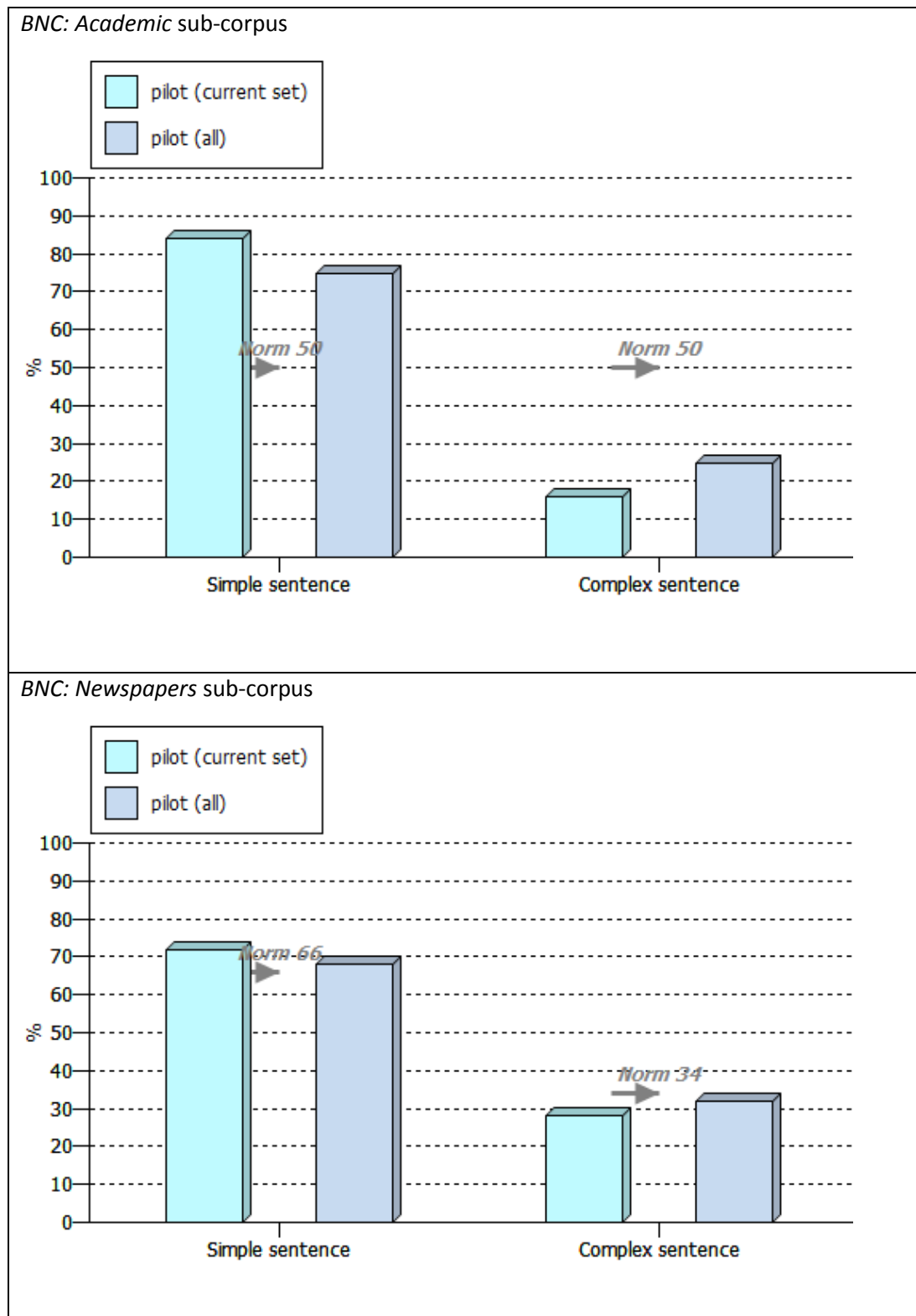


Figure 5.28: Different norm values for *Complexity* on the Graphs Tab, with results shown for the node *pilot* in the BNC: Academic sub-corpus (top) and the BNC: Newspapers sub-corpus (bottom).

5.5.3 Table

As well as visual information in the form of graphs, some users may prefer to view the actual frequencies of the node in each environment and see the BIC factor scores.

Therefore, a table icon on the Graphs Tab provides access to a complete list of all the priming features which have been measured. Table 5.39 (on the following page) shows part of the output of the primings table for *consequences* in the *BNC: Academic* sub-corpus.

If two terms are being compared side by side, the table shows results interweaved, with the aim of optimizing the potential for direct comparison of each feature. Figure 5.29 shows the output of the Graphs Tab table for *consequences* compared with *outcomes*.

Table 5.39: Table of some of the priming features for the node consequences in the BNC: Academic sub-corpus, with 100 results in the downloaded set.

Feature	Set Freq.	Set %	Set % - Norm	All %	All % - Norm	LL	Bayes Factor
Paragraph Position							
First Sentence	27	27	12	20	5	26.16	Strong evidence
Last Sentence	31	31	11	24	4		
Not First or Last Sentence	42	42	-23	56	-9		
Sentence Position							
First fifth	10	10	-8	11	-7		
First third	23	23	-8	23	-8		
Last third	43	43	7	44	8	49.2	Very strong evidence
Last fifth	21	21	-1	28	6	34.24	Very strong evidence
Not First or Last third	100	100	0	100	0		
Theme/Rheme							
Theme	20	20	3	16	-1		
Rheme	77	77	-2	81	2		
Complexity							
Simple sentence	55	55	5	50	0		
Complex sentence	45	45	-5	50	0		
Modality							
will / would / shall	2	2	1	2	1		
can / could / may / might	7	7	5	3	1		
must / should / need / ought	2	2	1	0	-1		
No modals	93	93	-2	95	0		
Voice							
Active Voice/Other	70	70	9	69	8	44.76	Very strong evidence
Passive Voice	30	30	-9	31	-8		
Polarity							
Positive	87	87	4	81	-2		
Negative	13	13	-4	19	2		

Subtype	Feature	Node	Set Freq.	Set%	Set%-Norm	All%	All%-Norm	LL	Bayes Factor
	Title/Heading								
	Position								
	Sentence Type								
	Complexity								
	Modality								
	Voice								
	Active Voice/Other	<i>consequences</i>	70	70.00	9.00	69.00	8.00	44.76	Very strong evidence
	Active Voice/Other	<i>outcomes</i>	70	70.00	9.00	63.00	2.00		
	Passive Voice	<i>consequences</i>	30	30.00	-9.00	31.00	-8.00		
	Passive Voice	<i>outcomes</i>	30	30.00	-9.00	37.00	-2.00		
	Polarity								
	Grammar								
	Definite/Indefinite								
	Definite articles/possessives	<i>consequences</i>	75	75.00	50.00	65.00	40.00	1161.47	Very strong evidence
	Definite articles/possessives	<i>outcomes</i>	42	42.00	17.00	38.00	13.00	33.77	Very strong evidence
	Indefinite articles	<i>consequences</i>	1	1.00	-7.00	2.00	-6.00		
	Indefinite articles	<i>outcomes</i>	2	2.00	-6.00	2.00	-6.00		
	No articles	<i>consequences</i>	24	24.00	-42.00	33.00	-33.00		
	No articles	<i>outcomes</i>	56	56.00	-10.00	59.00	-7.00		
	Prepositions								
	Near Prepositions	<i>consequences</i>	100	100.00	42.00	81.00	23.00	409.33	Very strong evidence
	Near Prepositions	<i>outcomes</i>	76	76.00	18.00	71.00	13.00	28.81	Very strong evidence
	Not Near Prepositions	<i>consequences</i>	0	0.00	-42.00	19.00	-23.00		

Figure 5.29: Screenshot of the table of features for the nodes *consequences* and *outcomes* in the BNC: Academic sub-corpus.

5.6 Filter and Compare Modes

Drawing the attention of language learners to the collocation, colligations and textual colligations of words was an important aim in the development of *The Prime Machine*. However, it was not thought that providing a summary of typical environments for a word or collocation should be an end in itself; rather the software should encourage learners to consider and explore for themselves whether the words they encounter or want to use in their own writing might be primed to occur with other features. To this end, a system was devised to allow users to move from the list of features on the Graphs Tab to a filtered list of concordance lines matching those features. Initially, using small icons on the collocation cards to show the priming features of each concordance line was considered, but it was felt that this would “crowd” the display too much and it would be confusing to have different icons present on each card. Instead, the option to filter concordance lines was developed. Figure 5.30 below shows the checkboxes and filter buttons available for one of the priming menus.

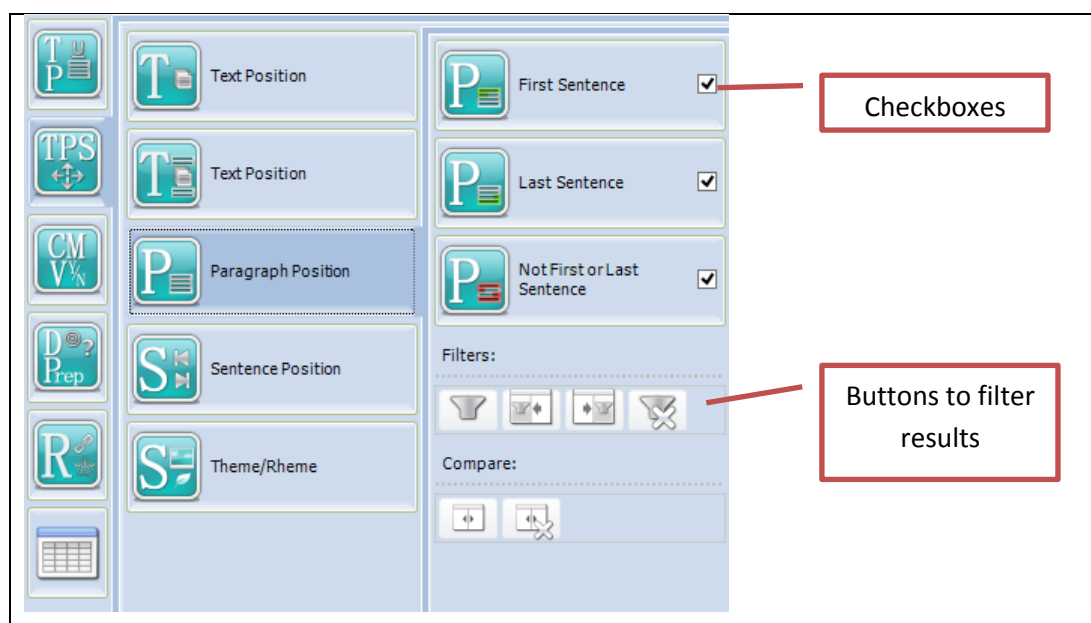


Figure 5.30: Checkboxes and filter buttons for one of the submenus on the Graphs Tab.

Checkboxes for each feature are checked as default, and by removing the ticks from some of these boxes, the user can filter down the results. In order to perform the filter, the program works through each card and compares the priming information which is stored about the line to the features which remain ticked. Cards which do not meet the requirements are marked to be hidden. The KWIC display on the lines tab also uses the information stored on the cards in order to hide or show rows in the table. It would be

possible to add functionality to retrieve only filtered results from the database since all the priming features are stored in the tables. Draft code to perform these requests was written for the middle-tier database, but in practice server-level filters could slow the system down and might be hard to show clearly in the client application. Therefore, currently filtering is only available for results which have already been downloaded to the client. This simplifies the overall query system considerably and also means that users cannot leave filters active by mistake when making a new query.

Looking at filtered results may help to show learners how a word or collocation is used in particular priming environments, but it was decided it may be helpful to provide the option to compare concordance lines for the same item to see whether patterns can be seen or conclusions can be drawn according to different contexts. Another important aspect of filtering and comparing is to allow learners to see variation as well as common patterns. Römer (2004) highlights the importance of seeing how modals are used in a range of meanings, for example. The complex categories used for some of the priming features can also be made easier to understand by showing users lines matching the features on the left and lines not matching those features on the right. Figure 5.31 shows the Lines Tab when in compare mode.

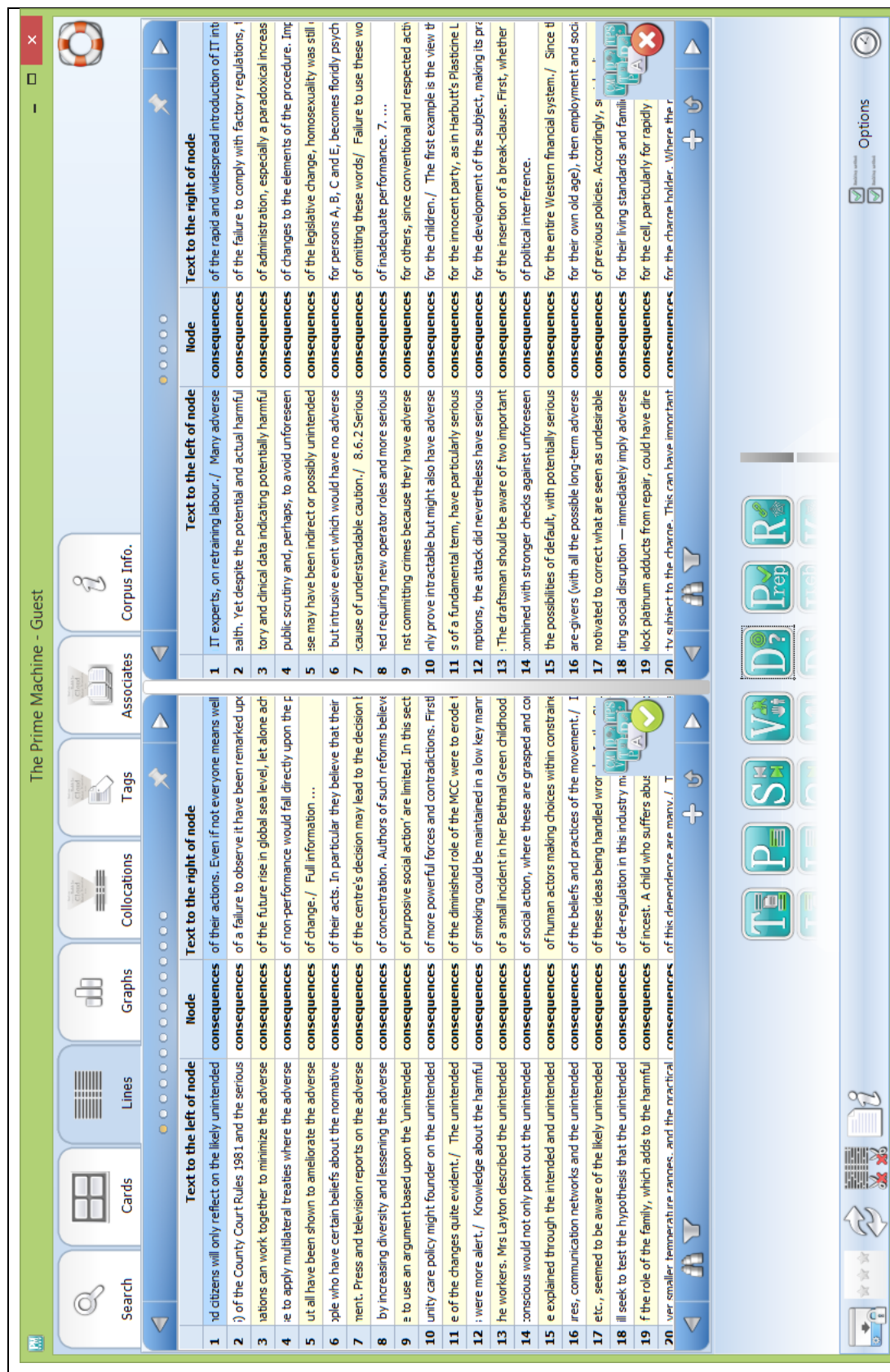




Figure 5.31: Compare mode for the node *consequences* in the *BNC: Academic* sub-corpus, filtered by *definite articles or possessives*.

The interaction of priming features is something for which Hoey (2005) provides several examples and is an area which could provide promising future research. It is hoped that *The Prime Machine* goes some way to providing a platform for greater awareness of the possibility of such interactions, but for the time being the program itself only provides the filter and compare mode as a way for an advanced user to explore these interactions manually.

It should also be noted that simpler filtering is available on the Cards and Lines tabs. Users can enter words or groups of words to be matched in a 4 word window (or more widely) and results can then be filtered or compared on this basis. As mentioned earlier, this feature could be used to filter lines containing specific modals for example. Results can also be filtered according to a user-specified rating, and this will be further explained in Chapter 7. The icons which appear superimposed on the grid of filtered concordance lines can be seen clearly in Table 5.40 below. They were designed to represent the whole range of filtering options with a green tick meaning all criteria have been met and a red cross indicating one or more have not been met.

Table 5.40: Icons used to show filtering is active

 Concordance lines matching the criteria	 Concordance lines not matching the criteria
--	--

Summary

This chapter has introduced the way in which several features from the theory of Lexical Priming are displayed prominently in *The Prime Machine* and the mark-up and calculations which are used to generate summary statistics and graphs. It has been demonstrated that the key word procedure can be applied to a range of features and the evidence suggests that words have a wide variety of primings. From a language teaching perspective, the textual colligation data seem to support the results of previous studies, and suggest that notions of topic sentences and concluding statements may need to be revised to focus more on the lexical items and discourse markers which are typically used in these positions. It is hoped that the aim of the software to draw learners' attention to the selection of features included in the colligation menus will resonate with language teachers and that drawing attention to these will help learners engage with the data in the concordance lines

more easily. Although the range of features is limited, some of the well-known trouble-spots for English for Academic Purposes have been targeted, with the use of articles and propositions, passive voice, and modal verbs included. In addition, the pervasiveness of simple and complex repetition of words at the medium frequency ranges seems to suggest that language teachers who teach students to try to avoid repeating words in their writing wherever possible may need to reconsider the validity of this advice. Clearly student essays are often shorter than academic journal texts, but it is likely that a reader may be unfavourably struck by the repetition of one word more than another due to the different primings of these words in terms of repetition.

Having introduced the design of the tabs related to searching, concordance lines, collocations and the other priming features, in the next chapter the last two ways of viewing data in *the Prime Machine* will be explained. The next chapter introduces the ways in which metadata and category labels are used to provide further ways of exploring the typical contexts of words and collocations.

Chapter 6: Metadata, KeyTags and Key Associates⁴⁷

As well as the words and sentences making up the language sample, corpus texts usually contain other information about the text or sections of the text which can be used in a concordancer. These metadata often provide details of the source including information about the participants or authors of the texts or other bibliographical information, as well as how each file fits into the corpus design: for instance which sub-genre it is intended to represent. They may be part of the header of the text file, or they may be tags (or enclosed in tags) in an XML tree. On the simplest level, when looking at a concordance line a user may wish to know more information about its source and concordancers usually offer some means of retrieving and displaying this information. For corpora which are comprised of published texts, when printing out a concordance line a service provider may need to include a copyright tag. A researcher who has a specific sub-set of documents or text types in mind may also want to use these metadata as a means of filtering the results; the query could specify, for example, that only concordance lines where the text type is identified as spoken data should be included. Another way that this kind of information might be used is to split a larger corpus into sub-corpora to allow a user to focus on one specific genre or to facilitate comparisons between the genres it contains. Comparisons may be based on searches within filtered results, or could be made through other procedures such as key word analysis which measure relative differences between corpora or sub-corpora. The purpose of this chapter is to introduce some of the uses of metadata in *The Prime Machine*. First, a summary of how these data are presented for each concordance line will be given. Then, after exploring the kinds of key word analysis which are available in other concordancers, a new procedure for measuring what are called KeyTags will be introduced. Finally, the purpose of the Associates Tab will be explained, along with the way in which the summary data it contains are calculated.

6.1 Showing category and citation information

One of the most basic labels for a concordance line is the name of the text from which it comes. Some corpora are created from collections of XML files where a tag in the file holds a suitable title with which to identify each individual text. Others have several tags which could be used in combination in the style of an academic reference. For example, the

⁴⁷ An early version of this chapter was privately produced for Xi'an Jiaotong-Liverpool University and is held by their research office.

Hindawi corpus and the *SpringerOpen* corpus have the title of each article, the journal name and information about the volume and issue stored within the file header. Similarly, newspaper corpora usually have information about the name and date of the publication. These fields can be easily combined to match typical referencing conventions. Other corpora, like the *BNC* and *BAWE*, have coded file names which can be matched to a separate table of sources, giving much more detail than is contained in the file header.

Typically, concordancers do not try to retrieve this information or display it as part of a citation for each concordance line. With *WordSmith Tools* and *AntConc*, since they work directly with text files on the users' hard drive, the header can be displayed as part of the full text. With *The Sketch Engine*, a code representing the file name is displayed in a column to the left of each KWIC line. The user is expected to know enough about the corpus which they are using to be able to look up information if he or she thinks it is relevant.

Language learners using a concordancer are much less likely to be aware of the composition of the corpus and also tend to be less sensitive to notions of how language use changes across different genres and registers. However, as mentioned in Chapter 4, an important point Hoey makes regarding all of the claims forming his theory of Lexical Priming is that they are "constrained by domain and/or genre" (2005, p. 13). Scott (2008) describes how familiarity with internet search engine result pages helps learners grasp more quickly how to understand the gathering together of snippets from many different sources in a screen of concordance lines. However, because search engine users actively consider each source on a list of results as a potential destination for web browsing and they will have entered the query with the object of getting to a suitable destination in mind, the sense in which each website on the list is a separate potential source for information is much more prominent on a results page from a search engine than it is on the concordance screen. With a search engine, the name of the website is often displayed along with the URL which in itself often gives clues regarding the nature of the resource in the domain name or country code. In the design of the Cards and Lines views for *The Prime Machine*, the question of how best to facilitate clearer information about the source of each concordance line was considered carefully.

From a presentation perspective, the design of the concordance line cards accommodates space for a text type and other information. Similarly, on the Lines Tab, pop up balloons can be easily added to cells in the table used to display KWIC results since they are a

feature of the AdvStringGrid component (TMS_Software, 2011). Figure 6.1 below shows one card from the *BNC* and one card from a *Hindawi* corpus with this citation information shown at the top. Figure 6.2 shows how the same information appears in the Lines view, although it should be remembered that unless the Compare Mode is active, the card for the currently selected row on the Lines Tab is always visible as well. Since this information does increase the height of each card substantially, not all users may want to keep the card citation information visible, and it can be turned off through clicking on a button on the task bar, or through the Options Tab.

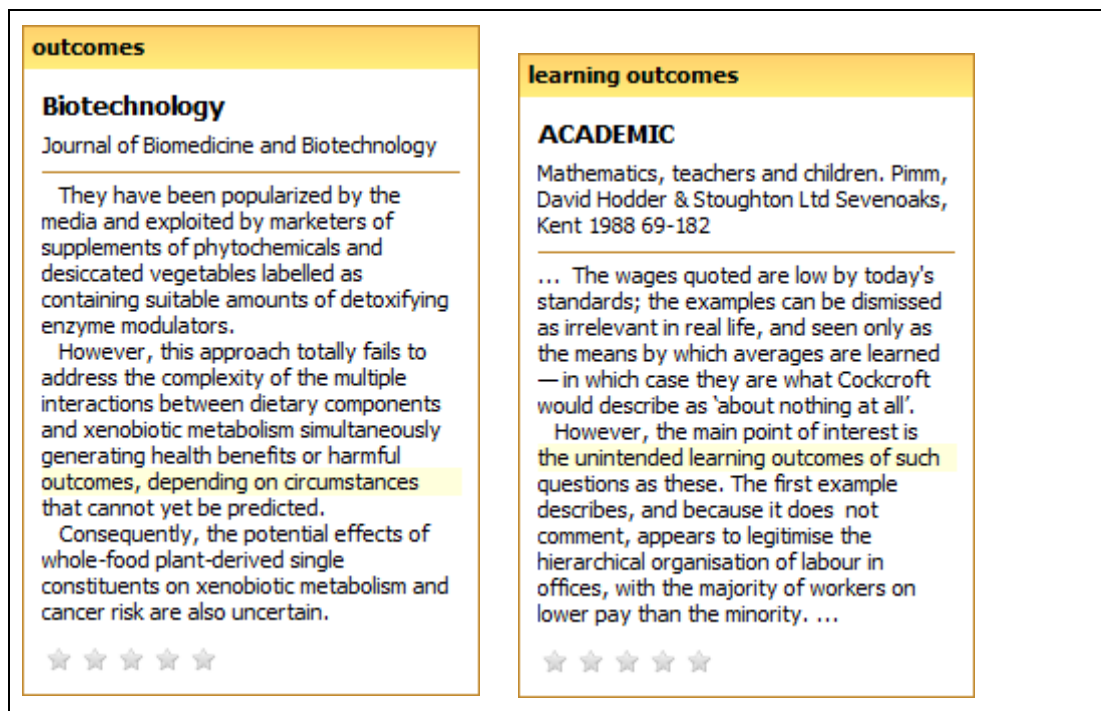


Figure 6.1: Citation information displayed at the top of a card from the *Hindawi Biological Sciences* corpus (left) and the complete *BNC* corpus (right), as shown on the Cards Tab.

	Text to the left of node	Node	Text to the right of node
1	short courses. These were to be based on a descriptor with learning	outcomes	and performance criteria, would be internally assessed, and would l
2	this session. The 29 PSD modules follow related patterns of learning	outcomes	and performance criteria and are grouped in four broad categories
	ACADEMIC	ing outcomes	and performance criteria were not being assessed or applied — all t
	The legal context of teaching. Johnstone, Susan Pearce,	ing outcomes	and performance criteria and only vary the assessment instruments
	Penelope Harris, Neville Longman Group UK Ltd Harlow	ing outcomes	and performance criteria, centres are free to implement PSD module
	1992 60-154	ing outcomes	and performance criteria throughout the module and to undertake :
7	ernal examinations, in that they will have set objectives, or learning	outcomes	, to be achieved, but will be free to plan their lessons as they think l
8	ues of various kinds are put into action to achieve practical learning	outcomes	./ These techniques may be the conscious application of ideas whic

Figure 6.2: Clipped screenshot from the Lines Tab, showing citation information for a concordance line from the *BNC* in a pop-up balloon.

From a software design perspective, however, there are further considerations regarding how text source information should be stored and retrieved. On the one hand, a relational

database which is set up with primary key links between each word, sentence and text provides a simple way for columns of data for any of these to be retrieved fairly quickly through table joins. However, with the software architecture of client – middle tier – database server, collecting all the metadata available for all concordance lines is somewhat wasteful because it is unlikely that a language learner would want to examine all this information for all concordance lines all the time. Designing a system for multiple threads serving simultaneous concordance line requests for multiple users means that some consideration needs to be made as to how many join operations between tables in the database are required, and to some extent planning for high performance means moving away from some of the stricter rules for normalization in database structure (Schwartz, et al., 2008). For these reasons, the metadata used as citations in *The Prime Machine* may be stored multiple times within the same database: once as a normalized table of strings which is connected through primary keys to texts, authors or sections; and once in the `cb_corpus_texts` table for each text in the form of a citation string which will be displayed directly in the card or balloon. The `cb_citation` row in the table holds a string of up to 255 characters and it can be set in two different ways. In the corpus refactoring application, there are three columns in the Transformation Rules table which facilitate manipulation of XML tags for the generation of citation fields. The first of these columns identifies a tag as being one of up to five ordered fields to be used to form the citation. The second contains an editable string for use as a prefix for this field, allowing punctuation or any other combination of characters to be used between fields. The third contains a list of standard operations which can be performed on the XML tag which currently includes two options: extraction of the last number or numeral from the string and conversion of YYMMDD or YYYYMMDD dates to DD/MM/YYYY. This approach works well for corpora with single or multiple tags containing information which can be ordered and combined together following a set of rules provided by the corpus manager. It generates clear source information for newspaper corpora or corpora derived from academic journal XML files. Table 6.1 shows the transformation rules for the *Financial Times* corpus, with data from the “DATE”, “PUB” and “PAGE” tags being used to create standardized citations.

Table 6.1: List of transformation rules related to citations for the *Financial Times* corpus

Tag	Pre-CLAWS Action	Post-CLAWS Action	Citation	Prefix	Transformation
DOCNO	X Ignore	Use as Filename	X Ignore		None
DATE	X Ignore	Meta Text as Date Without Adding as Text	Field2		YYMMDD
HEADLINE	Add <text> wrapper	Use as Title and Add as Text	X Ignore		None
TEXT	Process as text	X Ignore	X Ignore		None
PUB	X Ignore	X Ignore	Field1		None
PAGE	X Ignore	Meta Text Without Adding as Text	Field3	:	Extract Last Number or Numeral
BYLINE	X Ignore	Meta Author Without Adding as Text	X Ignore		None

However, for corpora which have separate tables of source information like the *BNC* and *BAWE*, a different strategy is available. The `cb_citation` field can be left blank during the refactoring process and once the entire corpus has been processed and all the other summary tables have been generated, a set of SQL update statements can be run on the server to populate the `cb_citation` fields with relevant information using each unique filename. For the *BNC* the list of XML file names and sources which is available from the website (Burnard, 2007b) can be imported into a spreadsheet. Corpora like *BAWE* contain separate spreadsheet files with columns for the filename and information which can be used to form a citation. In order to generate the SQL update commands, a fairly simple formula can be created in a spreadsheet application using a template. Figure 6.3 shows an extract of the SQL script for the *BNC* which was generated in this way.

```
update cb_corpus_texts set cb_citation:='Independent, electronic edition
of 1989-10-10: Sport section. Newspaper Publishing plc London 1989' where
cb_text_name='a4b.xml';

update cb_corpus_texts set cb_citation:='National Insurance Statutory
Sick Pay. Statutory Maternity Pay from 6 April 1991 for employers. u.p.'
where cb_text_name='a63.xml';

update cb_corpus_texts set cb_citation:='Converting old buildings.
Johnson, Alan David & Charles Publishers plc Newton Abbot, Devon 1988'
where cb_text_name='a79.xml';

update cb_corpus_texts set cb_citation:='Professionals and parents:
managing children\'s behaviour. Randall, Peter Gibb, Charles Macmillan
Publishers Ltd Basingstoke 1989 1-124' where cb_text_name='cgs.xml';
```

Figure 6.3: Examples of SQL script update statements created from a spreadsheet

In order to provide a quick sense of the kind of text from which the concordance line is taken, a table of text categories provides primary key links for each individual text to one main text category and this is used as the main heading for the citation section of the cards and for the balloons. Beyond the differences in the composition of different corpora used in *The Prime Machine*, a further complication to this is the fact that different corpora may have fundamentally different structures in terms of what a main category essentially might be. At one extreme, there are corpora which do not have any information about genre, register or major themes either encoded in the files or retrievable from other lists. For example, some newspaper corpora from the early 1990's do not include metadata about the main section of the newspaper (e.g. home news, business, fashion, etc.). In the *Guardian* corpus and the *Financial Times* corpus from the early 1990's there is no real indication of the section beyond the page number. Files in the latter part of these corpora sometimes include information about whether the text came from the main paper or a supplement and some texts have what could be category information. For example, for articles after 1993, there are <TP> tags containing categories such as "NEWS General News.", "MKTS Contracts." and "GOVT Government News" in the *Financial Times* corpus, but the list is far from complete. Another example of corpora for which sub-categories may not be very evident would be smaller specialist corpora where the genre and register or the producers of the language sample have been specifically targeted. It would not seem particularly helpful to break down any further a learner corpus such as *WECCL* which contains just under 1.5 million tokens. The main category divisions are not only important in terms of labeling concordance lines; they are also used to create sub-corpora for the key word and key associate calculations which are described in Section 6.5.

At the other extreme, there are corpora like the *BNC* which have been carefully constructed so as to provide groupings by genre or register. The XML edition of the *BNC* has several parallel major categories encoded in the files which offer different groupings and different levels of detail. In this project, the "type" attribute from the "text" or "stext" element is used. The documentation which accompanies the *BNC* explains that these are based on Lee's (2001) categorizations. These were chosen because they are easier to understand without needing to know too much more about the construction of the *BNC* itself. Just like with the other metadata, sometimes the actual grouping code enclosed in the XML tags used to determine major categories would not be very easy for students to understand. However, for corpora such as these, the major categories are very clearly defined and a simple set of *update* statements can be generated to change a code such as

“OTHERPUB” to “OTHER PUBLICATIONS” within the major category table. Another corpus which contains clear distinctions between text groupings is *BAWE*. Since this corpus was designed to capture comparable text data from four disciplinary areas, it seems sensible to use these as major categories. Again, a simple set of *update* commands was developed to expand the major category labels from “SS” to “Social Sciences” and so forth.

Between these two extremes, there is a third kind of corpus which is handled differently in terms of category labeling and key word calculations. These are corpora which are based on collections of texts from a specific group of sources, and these sources may be usefully divided into different categories. For example, for users wanting to explore words and collocations in a corpus created from academic journals related to a specific discipline, the register and genre would be constant, except in so far as the major topics or aims of a journal would distinguish it linguistically from other journals in the field. An important point to remember would be that language learners using *The Prime Machine* would choose such specialist corpora with a view to seeing how words or collocations are used specifically in that discipline. The *Hindawi* corpus provides a good range of academic journals available for open source data-mining, and these are a good source for academic sub-corpora, but the information on the website and in accompanying documentation does not provide any sub-grouping of the journals beyond major discipline areas (Engineering, Medicine, Chemistry, etc.). The organization of the *Hindawi* website is highly structured and it is possible to open pages containing the aims of each journal by adding the short journal code from the accompanying data sheet into a base URL address. In this way, the end of sentences containing indicators of topic like “in all areas of”, “in the field of”, “all aspects of” and “dealing with” were automatically retrieved to provide further support for checking the category assignments which were made during the process described below. The list of phrases used as cues for this was developed by manually looking at some of the aims pages which typically consisted of one paragraph of text for each journal. This meant that each journal had a major discipline label and full title as provided by *Hindawi* as well as snippets from each “aims” webpage containing lists of subject areas which were covered.

A useful resource which provided further information about electronic academic resources for each academic discipline was *intute*⁴⁸. Unfortunately, this searchable database which included notes and category information for each resource written collaboratively by lecturers from each discipline with its appropriateness as an academic source with students

⁴⁸ www.intute.ac.uk

in mind is no longer being updated, but the website still allowed access to the records which had been created up until 2011 for some time afterwards⁴⁹. Working through the list of journals in the *Hindawi* corpus, each was checked against the *intute* database and the sub-headings listed were entered on a spreadsheet for analysis later. For many academic subjects, *intute* provides three or four levels of sub-categorization and these were all noted and entered in the table. Some journals had been categorized in two or more ways. For academic information retrieval, it would be useful to retain several categorizations for one source, but for the key word analysis and major category assignment for *The Prime Machine* databases, a single categorization for each text was required. Fortunately, many of the sub-categories had a common parent category. In some cases, journals had multiple categories crossing disciplines and in these cases the category which matched *Hindawi's* own major subject grouping was usually selected. In a small number of cases there were two or more categorizations and the one which seemed to best match the title of the journal and aims from the journal website was selected. Not all the *Hindawi* journals were listed in the *intute* database, and for these the categories for journals with very similar names were used. In a small number of cases, no similar journal seemed to be listed, but often the main topic from the journal title or the "aims" page from the website was listed on *intute* as a sub-category, so this was used.

For all the *Hindawi* sub-corpora used in *The Prime Machine* except *Biological Sciences* this procedure provided a full list of major categories. For *Engineering*, the cross-over with *Computer Science* was evident in the listings on *intute* and where the journal was listed on *intute* under *Computer Science* but not *Engineering*, the *Computer Science* category was used. For *Biological Sciences*, however very few of the category fields had been populated at all, and none of the *Hindawi* journals had category information. By working through the list of major topics for each journal a list of 16 areas were identified. However, this seemed too many to display meaningfully on screen and also meant that what seemed to be very specific subject areas were elevated to the status of major category. In order to reduce the list of categories and ensure that the groupings would be meaningful to students in the subject area, I asked a colleague in the Department of Biological Sciences at Xi'an Jiaotong-Liverpool University to suggest major categories for these journals. She kindly provided a

⁴⁹ The information extracted from the *intute* database which is described in this chapter was accessed in January 2014. Sadly, as of January 2015, this resource is no longer accessible.

grouping which included 5 clear areas and 5 “others”. Thus 10 major categories were used for the *Biological Sciences Hindawi* corpus.

For most corpora, the major category is set for each text by selecting an XML tag which holds the category name, or by entering a default category to be used for all texts in the corpus. For corpora like *Hindawi*, the refactoring application allows for a lookup table to be loaded and a further option is provided to select an XML tag to use as the value for the lookup table. The lookup tables for each *Hindawi* sub-corpus contain a list of journal titles and categories.

6.2 Fitting labels into the database schema

There are a few pieces of information which are likely to be available for all corpus texts when they are held in raw form as either SGML or XML files. When corpora are refactored for *The Prime Machine*, as well as the citation and the main category fields each row in the corpus text table includes columns to hold a filename for the text and a copyright notice or header. However, the range and variety of other metadata are vast. A “many to many” design⁵⁰ is used in the database schema which ensures the database is as small as possible while linking metadata to linguistic data at a number of different levels. Beyond the citation and major category information described in the previous section, metadata can be added at a text level, an author level, a section level or a section author level.

The refactoring application provides a list of drop-down options for the handling of tags within the documents which are being processed. The operation on the tag can be made active if the full path in the tree matches a specified list, or if it has a matching parent node or if it occurs anywhere in the XML hierarchy. In this way tags which appear in different contexts can be grouped together or handled separately. For corpora which need to be run through *CLAWS* to produce part-of-speech vertical output files⁵¹, it is worth noting that the supplementary file which *CLAWS* also produces for each text is an XML file. Therefore, the refactoring application table of operations works with corpus files in a post-*CLAWS* phase, or if like the *BNC* they are supplied in XML with *CLAWS* tags encoded, directly on these.

⁵⁰ The “many to many” links can be seen in Figure 6.5, where the tables named “junction box” contain a list of links between the tables indicated and any item in the table can be linked to any number of items in the other table and vice versa.

⁵¹ Vertical output files in *CLAWS* hold each token on a separate line with part of speech information and use codes to show sentence boundaries and how the file corresponds to a supplementary file containing other XML markup.

The list of operations available includes using an XML tag as a label for all texts included in a single file, as well as adding it as metadata for one single text or for a section of text. Since the junction box connecting sentences to sections permits non-consecutive sentences to be linked to the same section, this actually provides a means of marking utterances from two or more participants in a spoken text to the metadata about each speaker. For a written text, the section level metadata provides a way of linking sentences to sections and facilitates marking of sub-sections and sub-sub-sections, etc. Links between sections and metadata are made in junction box tables in the database either as section text derived from headings or the text of the tag itself, or as author metadata. Figure 6.4 below shows some of the tags from the *BNC* and how they are to be handled in the refactoring process.

Tag	Mode	Pre-CLAWS Action	Post-CLAWS Action
sp	Tag Anywhere	X Ignore	Set Written Text Type as Containing Speech
stext	Tag Anywhere	X Ignore	Utterance Text Start and End
title	Full Path	X Ignore	Meta Text for All Sub-Documents in File Without Adding as Text
author	Full Path	X Ignore	Meta Author for All Sub-Documents in File Without Adding as Text
domicile	Full Path	X Ignore	Meta Author for All Sub-Documents in File Without Adding as Text
u	Tag Anywhere	X Ignore	Utterance Section Start and End
unclear	Tag Anywhere	X Ignore	Make #Name# for milestone tags
w	Tag Anywhere	X Ignore	Word tag for pre-tagged files
who	Matching Parent	X Ignore	Set section author using this ID string
event	Tag Anywhere	X Ignore	Allow only one setting for speech texts
<			Make text metadata if only one setting
			Make date metadata if only one setting
			Speech Settings Wrapper
			Paragraph tag for pre-tagged files and first one marks beginning of
			Use as Main Text Category
			Turn Section On and Off and use name as section label
			Use as lookup value for category table

Figure 6.4: Clipped screenshot from the Refactoring Application, showing examples of tags and how these are to be handled as metadata in the database

Even with a well-documented and carefully constructed resource like the *BNC*, however, there can be inconsistencies or missing data in the files. For example, the *BNC* manual explains in detail how the XML tags work and what they include, but even this level of detail does not include all the information one might want regarding aspects like the sampling method for some of the texts (see Chapter 5), and several of the common tags for describing people simply mean “unknown”. There is also some variety in the amount of information stored in the text files in the *BNC*, so for example, there are spoken data files where no “person” tags are established in the header, but different “unique identifiers” appear in the running text as and when they come up.

As explained in Chapter 3, the refactoring application is designed to be administered by an advanced user able to set up corpora for students and teachers to access. The corpus

administrator can decide whether the tags should be included as ordinary text (meaning they can be seen in concordance lines and add to the token count for the corpus), or whether they should be added to the database purely as metadata. The process described above provides a means of storing metadata in the database in a way which balances the need for speedy retrieval of some data against the optimization and normalization of other data. Figure 6.5 shows the database tables for the junction boxes and the metadata strings of text.

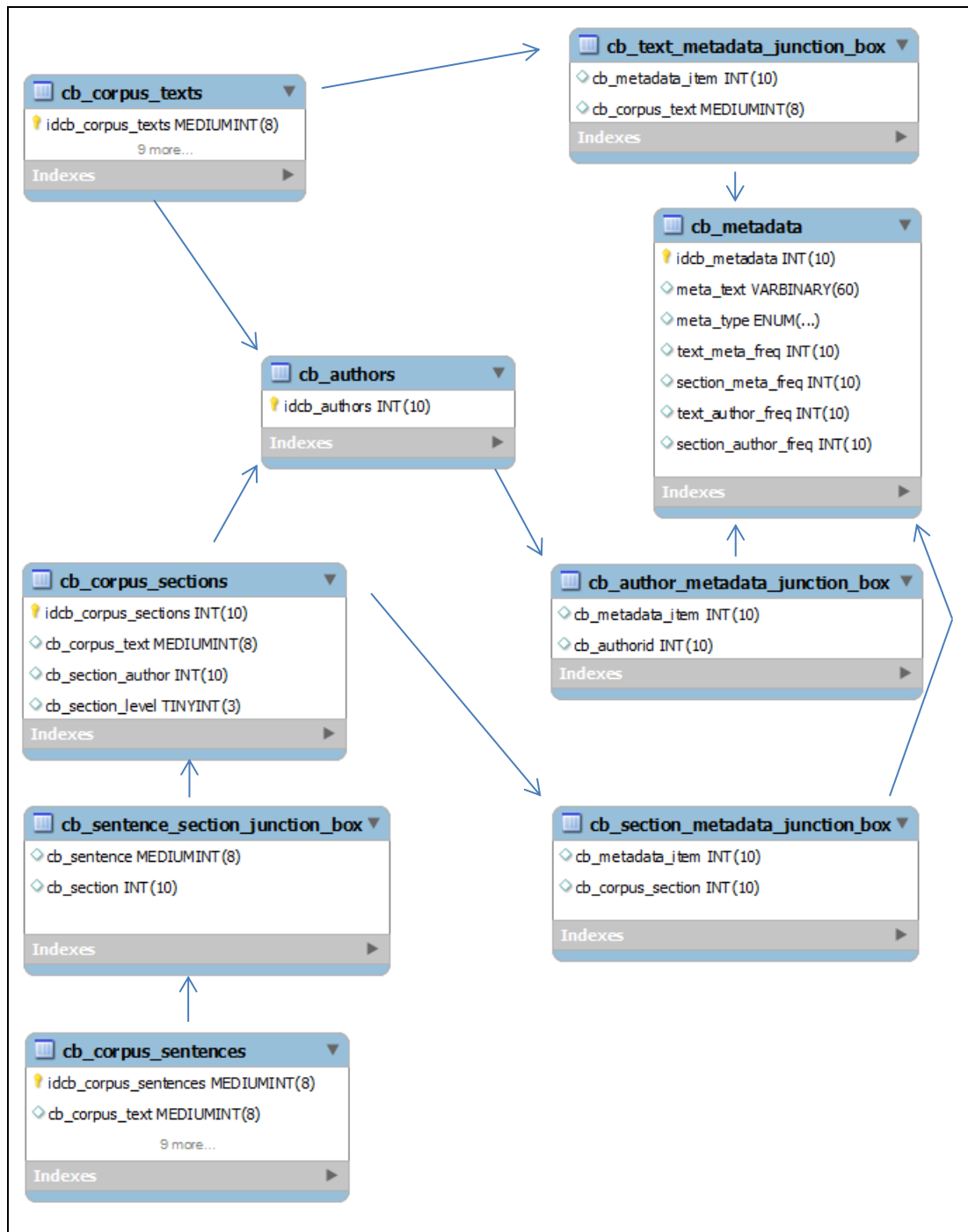


Figure 6.5: Database schema for metadata, with arrows showing links through primary keys

For the wide range of metadata stored in the `cb_metadata` table and linked in junction boxes, “join” queries in *MySQL* mean that filters can be applied so only the texts, sentences or words meeting the criteria are retrieved or counted.

As well as the citation-like information which is displayed for each line on the card or in the balloon, further information can be requested through the menu which appears when the user double-clicks on a concordance card or line. The information appears grouped according to the kind of metadata stored: a block of text stored as the header for the text and tags which have been stored in the metadata table and were identified in the refactoring process as holding information about the text, information about the author or producer, or information about the section in which the concordance line occurred. Figure 6.6 and Figure 6.7 show some of the pop-up windows which can be viewed in a scroll-box for one concordance card from the *BNC* and one line from *Hindawi Biological Sciences* corpus.

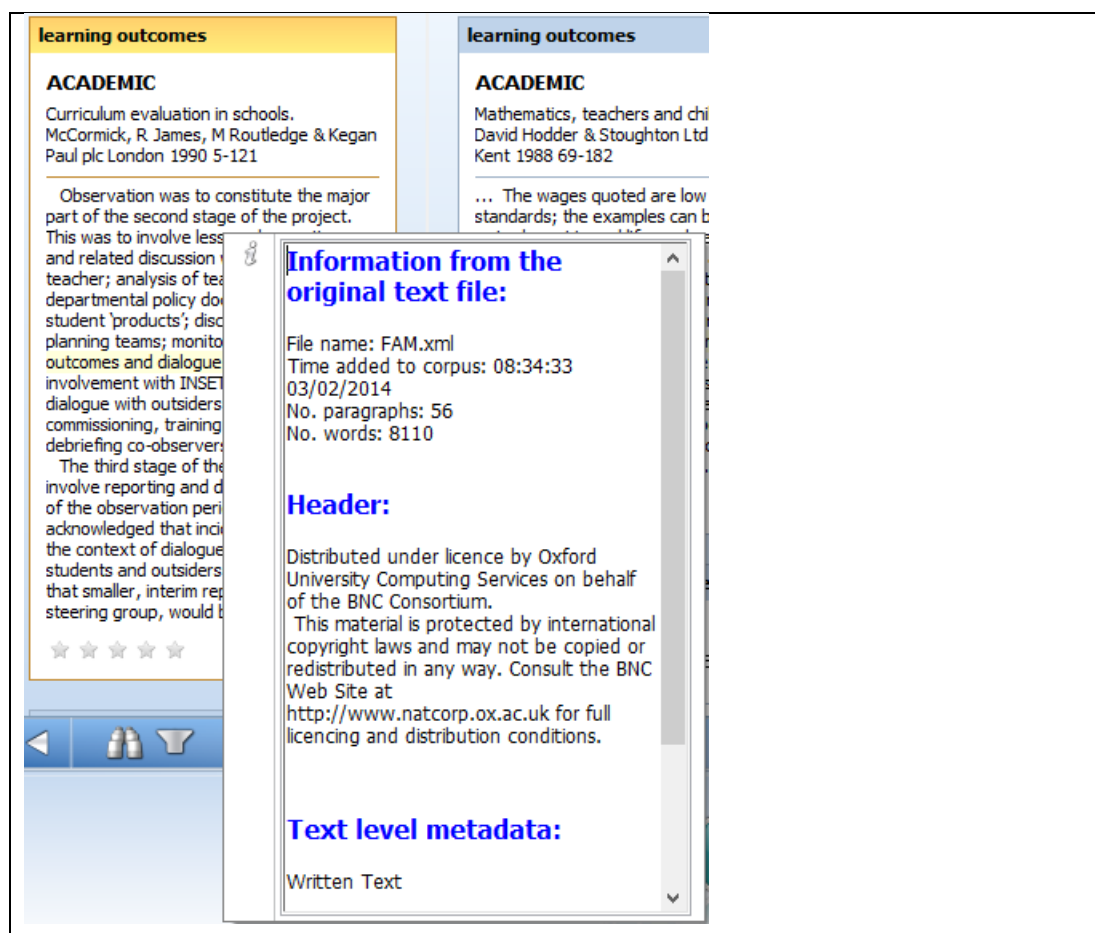


Figure 6.6: Further pop-up information for a card from the *BNC*.

	Text to the left of node	Node	Text to the right of node
1	rts that changes in the environment can be associated with positive	outcomes	[#xref#], in which older adults relocate to
2	IS analysis./ These results are validated by comparison to previous	outcomes	which confirm an improvement in terms of
3	revealed no beneficial effect of probiotics/synbiotics in term o		
4	iciency of modified RNA/DNA polymerization could naturally l		
5	rs meant to differentiate between melanoma patients with d		
6	ed, it does serve to illustrate the potential "normality" of und		
7	address all of the possible sources of failure or routes to ha		
8	utional collaboration leads demonstrably to improved environ		
9	3 F- DODA or 18 F- FMT PET has been exploited to evaluate		
10	sess the association of XMRV with HIV- 1 infection and othe		
11	ling process to allow for implementation of on- ground conse		
12	insplantation of neutrophils or their precursors might improve		
13	Concluding Remarks/ Although engraftment of UCB cells and		
14	relationship between social resources and physical and menta		
15	ial assessments of social relationships on physical and menta		
16	model is the hypothesized indirect effect of social relations or		
17	this populations' overall health status by reducing negative		
18	ed reproduction during the early days of pregnancy. These		
19	dians and Chinese, although we did not have a sufficient nu		
20	and Itk effects on molecular pathways resulting in functional		

Information from the original text file:

File name: 131505
 Time added to corpus: 17:56:19
 18/11/2014
 No. paragraphs: 40
 No. words: 4039

Header:

Copyright © 2010
 This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Text level metadata:

Journal of Biomedicine and Biotechnology
 2009

Figure 6.7: Further pop-up information for a concordance line from the *Hindawi Biological Sciences* corpus.

6.3 Key words

Having described the storage of the tags and labels which might form the basis of filtering or splitting a corpus into sub-corpora, this section considers possible uses of key word procedures. First, keyness measures available in other concordancers will be introduced. After that, some important issues regarding the selection of a reference corpus and the level of analysis for word frequencies will be considered.

6.3.1 What do concordancers offer in terms of key words?

WordSmith Tools offers the most comprehensive set of functions related to key words. This is not surprising given the contribution that Scott's software as well as his own papers on language have made to this area of corpus linguistics. Since key words are based on results of wordlists in *WordSmith Tools*, the first set of options is related to the extent of the wordlists; whether they are to be calculated from wordlists in batch mode with one list for each file, or whether the list should be for the entire corpus. Since files can also be split or merged using its range of file utility programs, this effectively provides the ability to create key words for each text, each sub-corpora or each corpus. With batch processing and Key Word Database creation, Key Key Words can also be calculated, showing words

which are key in many texts. Scott provides examples of calculating keyness on different levels and exploring the key key words of different types of corpora (Scott & Tribble, 2006). As well as many options for selecting the chi-square or log likelihood measure, the level of significance, the number of key words required and the minimum number of texts and minimum frequency required, *WordSmith Tools* also has several other automatic calculations related to key words which can be performed including associates, clusters, links and clumps.

AntConc takes some of the functionality of *WordSmith Tools* and offers a calculation of keyness using either a simple flat file containing word forms and raw frequencies or text files containing running text as the reference corpus. It succeeds in cutting down some of the complexity of *WordSmith Tools* for more limited exploration, and given the package's overall limitations in terms of corpus size the provision of only simple key word lists rather than other additional summaries and methods seems appropriate. Some of the default settings may not always be ideal, and it may be necessary to take students through three separate steps to ensure case-sensitivity does not interfere with the results⁵². However, *AntConc* provides a fairly steady path for students to take texts, compare them to a reference corpus (perhaps prepared in advance as a flat file by the teacher), and then use the key words as a useful starting point for the exploration of concordance lines and collocations of some "important" words.

Another software package which provides extensive functionality in terms of key word analyses is *WMatrix*. As explained in the development of *Matrix* and followed on in the web-version *WMatrix*, Rayson (2002) established a set of tools for comparing two texts. As well as log-likelihood based measures of keyness for words, the system also provides keyness for part-of-speech through its integration with CLAWS (Garside & Smith, 1997) and semantic meaning through integration with UCREL's semantic tagger (Rayson, Archer, et al., 2004). Part of Rayson's research in developing this system was not only to provide a new tool for these kinds of analyses, but also to establish the robustness of the log-likelihood statistic and recommendations for its interpretation. His work demonstrated how keyness can be used to examine the aboutness of competing documents such as party political manifestos.

⁵² For example, if a task involves using a flat file as the reference corpus in which all tokens appear in lowercase, sentence initial instances of words in the texts become "key" if case sensitivity is not turned off on the Wordlist tab, the Keyword tab and on the menu for selecting the reference corpus.

A web resource specifically designed for language learners, teachers and researchers is *LexTutor* (Cobb, 2000). This set of corpus tools is regularly updated and provides a range of ways of highlighting individual texts and creating gap-fill exercises using vocabulary frequency profiles or by selecting words at fixed intervals. It also provides tools for measuring the range of items across very small corpora, and includes a tool called “Keywords”. Although this key words tool is limited by the need to submit each text for analysis individually, with limits of 50,000 words (if pasted) and 1 megabyte (if uploaded), it does provide a simple list of words which are proportionally more frequent in the text than the reference corpus. The reference corpus is fixed as the spoken section of the *BNC*, but if a file is uploaded rather than pasted into the box the medical sub-section of the *BNC* can be chosen instead. The words in the text are converted to word families and any words not found in the reference corpus are discarded. There is a space for the user to enter words which will be excluded from the analysis as well as the default option to exclude words mid-sentence which begin with a capital letter. A very important difference between the key words listed in this tool and those in *WordSmith Tools* or *AntConc* is that the measure is not based on log-likelihood or any test of statistical significance. Instead, the score is based on how many times more frequently each word family occurs in the text than in the reference corpus. The output screen clearly explains this calculation using a template message which includes the top key word as an example in the explanatory message. The formula is shown in Figure 6.8 below.

$$\frac{\left(\frac{\text{frequency in text}}{\text{text word count}} \times \text{reference corpus word count} \right)}{\text{frequency in reference corpus}}$$

Figure 6.8: Measure for key words in *LexTutor*

The tools on this website are geared towards encouraging students and teachers to notice and build on vocabulary using vocabulary frequency lists, and this aim is made explicit on the Research Base page (<http://www.lextutor.ca/research/> accessed 7 February 2014). As such, the conflation of word forms into families and the restriction of words from a choice of two reference corpora is understandable. The website brings together a set of corpus tools for analysis and exploitation for teaching of texts. Although it could be one way to introduce language learners to concordancing based on very small corpora and text samples, it does not offer the flexibility or scalability of *WordSmith Tools* or *AntConc*.

Key words are also handled rather differently in *The Sketch Engine*. On the Wordlist page, the user can select a list of key words as the output, and it is possible to use other corpora or subcorpora which are available in the system as the reference corpus. The measure is based on the normalized frequency in the source corpus divided by the normalized frequency in the reference corpus and does not use a test of statistical significance. Rather than simply discounting words which do not occur in the reference corpus, the function in *The Sketch Engine* employs an adding technique which it is argued not only solves the problem of division by zero, but also provides a gauge for extracting key words for different frequency ranges (Kilgarriff, 2009b).

Each of these tools provides various means to compare two textual objects, with some focussing more on individual texts while others provide flexibility to compare sub-corpora or complete databases. Key word analysis is not built into *CenDiPede*, as the focus was on creating profiles for words rather than analysis of texts, but Garretson does consider extension of his profiling techniques as a complementary way of comparing differences in the features of whole texts as a future development (Garretson, 2010). However, in the other software described above, the differences in approach and the apparent split between those basing scores on normalized frequency and those employing tests of statistical significance is striking. In the next section further consideration will be given to the question of how measures and division of corpora can be best matched to specific research purposes. After that, an explanation of how the key word technique has been integrated into *The Prime Machine* and for what purpose will be given.

6.3.2 Limits and cautions for Log-Likelihood Key Words

Although Key Word features in *WordSmith Tools* and other packages have been used for a wide variety of studies over the last fifteen years or so, there have been some recent reservations. Gabrielatos and Marchi (2012) present a thought-provoking examination of the definition and methods used in key word analysis. The paper makes a number of claims leading towards the suggestion that %DIFF would be a more suitable method for ranking keyness. Since these claims bring together a number of important threads including definitions of keyness and keyword, the use of the log-likelihood measure and the selection of reference corpora, the aim of this section is to explore each one and to consider whether there are alternative ways to make log-likelihood based keyness measures more robust. Figure 6.9 below contains a summary of the claims, with quotations extracted from the slides of their presentation which were kindly posted on the institutional repository at Edge Hill University.

Claims:

- The usual definition of *keyword* is not consistent with the metric for *keyness*.
- The statistical significance of a frequency difference is not a good metric for *keyness*.
- A measure of effect size is needed to ensure that statistically significant results are not just a result of having a large sample.
- The best available measure of effect size would be %DIFF which is noted as “The % difference of the frequency of a word in the study corpus when compared to that in the reference corpus.”
- “The current threshold for statistical significance ($p \leq 0.01$, $LL \geq 6.63$) is arbitrary”

Figure 6.9: Claims made by Gabrielatos and Marchi (2012)

The first two claims relate to definitions and the difference between statistical significance and linguistic importance. A starting point for a definition of *keyness* might be the explanation of the purpose and means of calculation provided in the *WordSmith Tools* manual:

Key words are those whose frequency is unusually high in comparison with some norm.

(Scott, 2010c)

To compute the "key-ness" of an item, the program therefore computes

- its frequency in the small word-list
- the number of running words in the small word-list
- its frequency in the reference corpus
- the number of running words in the reference corpus

and cross-tabulates these.

(Scott, 2010b)

The criticism from Gabrielatos and Marchi (2012) is that the definition confuses statistical significance with effect size; or perhaps that the log-likelihood measurement and the definition of *keyness* given in many studies which use it blur the line between what *keyness* is and how it can be measured. However, just as Hoey (2005) reflects on the need to consider the difference between statistical and psychological definitions for collocations (as quoted in Chapter 4), definitions often given for key words are *statistical definitions*. The psychological importance of *keyness* and the clues as to why this phenomenon should exist are summarized by Scott as follows:

Many languages use the metaphor “key” to identify people, places, words, ideas as important; the term is used in text databases for retrieval and in practice it does not seem to need to be defined. Keyness is, therefore, a quality which is generally intuitively obvious. Here, though, we must think about the term more carefully... So, for us, keyness is a quality words may have in a given text or set of texts, suggesting that they are important, they reflect what the text is really about, avoiding trivia and insignificant detail. What the text “boils down to” is its keyness, once we have steamed off the verbiage, the adornment, the blah blah blah.

(Scott & Tribble, 2006, pp. 55-56)

As Wilson points out, despite “some aura of novelty”, keyness is “nothing more than an ordinary null hypothesis significance test applied to the frequencies of words or other items in two texts or corpora” (Wilson, 2013, p. 1). The contingency table and formulae usually used for keyness are given in Table 4.3 of Chapter 4. Keyness in *WordSmith Tools* and *AntConc* is based on the application of this kind of statistic, with the highest log-likelihood or chi-squared results being interpreted as being the most important.

In Gabrielatos and Marchi’s presentation, however, effect size is equated to the percentage difference in frequency. They suggest that %DIFF should be used, with a miniscule number added to any item not appearing in the reference corpus. Since this combination is not currently available in concordancers, they propose a manual method for calculating this in a spreadsheet. %DIFF is given in the presentation as:

Equation 6.1: Formula for %DIFF (Gabrielatos & Marchi, 2012)

$$((NF \text{ in SC} - NF \text{ in RC}) / NF \text{ in RC}) * 100$$

NF = Normalised frequency

SC = study corpus

RC = reference corpus

Although this calculation is a little different, it is more closely related to the metrics used in *Lextutor* or *The Sketch Engine*. Both %DIFF and log-likelihood approaches use the *relative* frequencies of words. However, since only normalized frequency is used in %DIFF, it could be argued that it reduces the data, leaving out other important information. For most analyses where individual texts form the unit of study, effect size should also consider the relative frequency with respect to the whole sample. The “(a+b)/(c+d)” part of the

equation is missing in %DIFF; that is to say some concern with how wide the phenomena is across the *combined* corpus. An important reminder given in the third claim quoted above is that large sample sizes lead to high log-likelihood scores. The main reason given for this is that very large corpora can mean that the normal p value cut-off points are too low. They propose setting a threshold “relative to the resulting range of %DIFF values” (Gabrielatos & Marchi, 2012). However, having relative thresholds determined subjectively means that different studies will set their own cut-off thresholds and cross-study comparisons will be extremely difficult. Another cautionary note regarding the interpretation of log-likelihood scores for keyness across studies is that cut-off points are usually set by each individual user (Baker, 2004). The issue of balancing for effect size is also raised by Wilson (2013), where, as discussed in Chapter 4, he proposes using an approximation of Bayes Factors. This means that as the total combined corpus size increases, the LL score has to be higher in order to meet the same BIC. As mentioned in Chapter 4, using Bayes Factors also alleviates some of the problems of comparability across studies.

In order to demonstrate some of the dangers of blindly applying log-likelihood to data without considering the actual difference in frequency, Gabrielatos and Marchi (2012) compared the proportion of overlap between the rankings of LL and %DIFF for all key words and the top 100 key words for several corpora. They claimed that low overlap would mean “one metric is inappropriate”. However, it would be fairer to say that low overlap would mean the metrics measure different phenomena. Different kinds of focus might lead researchers to prefer a %DIFF calculation over log-likelihood in certain circumstances. Reflection on the appropriateness of one measure or another needs to be keenly attuned to the purposes and aims of the research. Table 6.2 below shows my own reflections on how different kinds of research may differ in terms of their aims and the purposes to which lists of keyness values will be put.

Table 6.2: Different research aims and different uses of keyness values

Text focus	Genre / Register focus	Sociolinguistic focus
<ul style="list-style-type: none"> • Measure Key Words in each individual text. 	<ul style="list-style-type: none"> • Measure Key Words in separate genres or registers; • Measure Key Key Words across individual texts within a genre or register. 	<ul style="list-style-type: none"> • Measure Key Words in one entire corpus against a reference corpus.
<ul style="list-style-type: none"> • Aim is to determine the “aboutness” of each individual text. 	<ul style="list-style-type: none"> • Aim is to identify words which are important in a particular genre or register. 	<ul style="list-style-type: none"> • Aim is to determine whether there are any major themes which are different, as well as differences in register and style such as use of pronouns and contractions.
<ul style="list-style-type: none"> • The reader of the text might agree that the text was about these things. 	<ul style="list-style-type: none"> • A reader familiar with this genre or register would acknowledge that the Key Words or Key Key Words are important to the field. 	<ul style="list-style-type: none"> • The reader of all the texts may not be aware of some of the differences because they are likely to be widely distributed across all the texts. • A text chosen from this collection might “seem” to fit into the category; a text not exhibiting these major features might not “seem quite right” in terms of style.
<ul style="list-style-type: none"> • A researcher might use some of these Key Words as good starting points for further analysis. 	<ul style="list-style-type: none"> • A researcher might use lists of results to show differences in importance of topic indicators or to find potential candidates which occur in two different sets to explore how use of these items differs. 	<ul style="list-style-type: none"> • A researcher would need to consider the results balanced against some understanding of how other expressions or non-linguistic features could be used by the target social groups to perform similar communicative goals.

Several important considerations for the application of key word and key key word techniques are given by Scott as he explores the results obtained when the scope of the wordlist and the reference corpus used are adjusted (Scott & Tribble, 2006). For example, when reporting results for one of Shakespeare’s plays using the other plays as a reference corpus, Scott explains how some key words in the results “do not reflect importance and aboutness”, and might be considered to be indicators of style (Scott & Tribble, 2006, p. 60). He goes on to suggest that concordancing some of these “intruders” can provide useful

insights. However, when moving from individual texts to collections or entire corpus databases, Scott demonstrates that both Key Words and Key Key Words can have a tendency to become more similar to a raw frequency word list, especially if the study corpus is more general in nature. Through the examples provided by Scott (2006) in experimenting with different reference corpora and looking at key key words on genre or whole database level, it is clear that the “aboutness” is not always easily obtained and careful consideration needs to be given to both these factors.

Although these potential pitfalls are introduced clearly in these examples and the help pages of *WordSmith Tools* also point out important issues regarding these points, one of the most shocking revelations of the comparison provided by Gabrielatos and Marchi (2012) for %DIFF versus Log-Likelihood was the extremely high LL values and very high rankings for the two most common words in English (*the* and *of*) despite only small differences in %DIFF. When conducting computer lab sessions teaching sophomore English majors how to perform key word analysis on small corpora, my own experience has been that if “the” appears in the results either one of the case-sensitivity options in *AntConc* had been missed or the key word reference file had been incorrectly set up and so the results were due to a procedural error. For LL scores to be so high while the underlying differences in frequency are small, the results for these two words presented by Gabrielatos and Marchi seem to be of greater concern than some of the differences between LL and &DIFF for proper names, etc. Gabrielatos kindly provided by email the key word lists they had used to generate these results and informed me that the data were taken from a comparison of the *SiBol93* and *SiBol05* corpora⁵³. Both of these are actually substantial corpora based on newspapers with half a million texts or more in each. Figure 6.10 provides information about the sources for these corpora provided from the *SiBol* corpus website.

- SiBol 93 containing the entire output of the *Guardian*, *Times*, *Telegraph* and the *Sunday Times* and *Sunday Telegraph* for 1993.
- SiBol 05 containing the entire output of the *Guardian*, *Times*, *Telegraph* and the *Observer*, *Sunday Times* and *Sunday Telegraph* for 2005.

Source: http://www3.lingue.unibo.it/blog/clb/?page_id=8

Figure 6.10: Details about the SiBol 93 and SiBol 05 corpora

The *WordSmith Tools* key word files indicated that these corpora had been loaded from approximately 60 individual files, so the texts were not organised in such a way that each

⁵³ Gabrielatos, personal communication, 31 May 2013.

individual news article (or even each issue of the paper) was contained in a separate file. A conservative estimate of the number of actual news stores contained in each of these corpora, if the rough average from the *Guardian* corpus of 400 words per article is applied, would suggest *SiBol93* would have more than 240,000 texts and *SiBol05* would have 390,000 texts. In their presentation, Gabrielatos and Marchi showed the following results and conclusion:

<ul style="list-style-type: none">• <i>The</i> LL = 32,366.01 (2nd) <u>but</u> %DIFF = 9.7% (4302nd)• <i>Of</i> LL = 20,935.05 (5th) <u>but</u> %DIFF = 11.8% <p>What the high LL values indicate here is that we can be highly confident that there is a very small frequency difference</p> <p>(Gabrielatos & Marchi, 2012, p. 24)</p>
--

While they do highlight an important point, it could be argued that the suggestion that these are just small frequency differences is a little misleading. Table 6.3 below shows the raw frequencies and log-likelihood keyness score for their data⁵⁴, with the 1993 corpus as study corpus at the top and the 2005 corpus as study corpus at the bottom.

⁵⁴ The Key Word lists were loaded directly into *WordSmith Tools*, small differences in the keyness values from those given in the original presentation are likely to be a result of slightly different configurations. However, the rankings of the items seems to be identical to those presented in the slides.

Table 6.3 Data from by Gabrielatos and Marchi (2012)

SiBol 1993 Study Corpus with SiBol 2005 as Reference Corpus					
N	Key word	Freq.	RC. Freq.	Keyness	
1	MR	206523	176385	30473.98	
2	THE	6001857	8247131	29461.11	
3	EC	15204	623	23281.55	
4	CLINTON	19793	3264	20843.41	
5	OF	2782374	3743488	19958.21	
6	BOSNIA	13488	910	18890.35	
7	1991	18233	4008	16561.94	
8	RECESSION	12484	1101	16386.54	
9	YELTSIN	9829	217	16162.65	
10	CORRESPONDENT	14743	2521	15268.38	
11	MAJOR	41747	24322	14473.85	
12	MAASTRICHT	8669	300	13583.99	
13	WHICH	316733	359203	13253.9	
14	MILLION	84491	70938	13115.26	
15	BOSNIAN	9159	623	12804.46	
SiBol 2005 Study Corpus with SiBol 1993 as ReferenceCorpus					
N	Key word	Freq.	RC. Freq.	Keyness	
1	WWW	66838	2	68156.2	
2	UK	118161	17784	47720.79	
3	YOU	481558	183494	44568.26	
4	CO	61061	3407	41886.06	
5	I	869471	399064	39526.73	
6	COM	37723	96	37324.57	
7	BLAIR	42902	1771	32117.28	
8	T	324154	126946	27499.03	
9	2003	27853	160	26738.39	
10	2004	24912	119	24123.96	
11	2005	24294	137	23343.33	
12	YOUR	140823	46009	19520.34	
13	EU	21410	771	16562.59	
14	0870	16001	1	16304.85	
15	2001	18480	316	16224.56	

Adjusting for the differences in size, *the* occurs 1.5 million times more or one extra time in every one hundred words, and *of* occurs 0.78 million times more or one extra time in every two hundred words. Using the conservative estimates of the number of texts contained in these corpora, this would equate to an average of four and two more occurrences in each text respectively. The difference is perhaps small given the high frequency of these two items in any English text, but the main issue is that they are not usually considered to be

interesting or noticeable, and neither can really point to the aboutness of these corpora. However, in the log-likelihood data there are indications of what was important in the news in 1993, with the important *Maastricht* treaty, president names and the British Prime Minister's names, the *EC* as well as the troubles in Bosnia being evident. Looking in the other direction, the introduction of website links and other interactivity into broadsheet newspapers is evident in *www*, *co* and *com* forming URL addresses, and it is likely that some of the increase in the use of *uk* could also be down to these. The increase of *you* and *your* and 0870 (a UK prefix for "National Rate" telephone numbers which are "non-geographic") also supports this. However, despite the feeling that the years might be obvious changes to skip over, the LL measure again picks up some changes which took place, with Prime Minister *Blair* seventh on the list, and *EU* taking over as the European Community became the European Union. The other "intruders" such as "Mr", "correspondent" and "which" in 1993 and "I" in 2005 might well be reasonable starting points for further investigation of changes in style. The suggestion that these differences are small is not really accurate; it would be more accurate to suggest that key word analysis based on a wordlist of each complete corpus is unlikely to provide very detailed information about changes in what is newsworthy over two periods, or as a resource for the exploration of what was happening in the world at those times. Using wordlists from the combined texts will bring out some stylistic changes and other differences such as the internet links, but does not really answer the question of what would have been noticeably different to a reader of newspapers during each year. This is true of both %DIFF and LL rankings. In order to measure how changes took place in terms of what was newsworthy, it could be argued that it would be better to work with each text in a separate file since the key words for each individual text can correspond to what might be psychologically more salient. Measuring changes in the proportion of texts in which an item is key (using key words) ought to be a better means of investigation. In their presentation, they used *adventists* and *ex-communist* as examples where there was a large %DIFF but relatively low LL; however, the raw frequencies for these were 94:6 and 134:26 respectively. Given the corpora had hundreds of thousands of texts it is just as unlikely that a reader would notice these large proportional changes, as it would be that they might notice the decreased use of *Mr*, *the*, or *of*. The important point that these examples illustrate is not so much that one measure is superior to another, but rather that organization of texts into files and the choice of reference corpus are very important when determining the best strategy for a specific research question.

6.3.3 Priorities for keyness measures in *The Prime Machine*

In the previous section it has been argued that log-likelihood based key word calculations can be used effectively for a range of different kinds of research, but often work best with texts rather than at an entire corpora level. It was also suggested that Bayes Factors should work well in contexts where corpora are large. The log-likelihood measure is particularly good at revealing:

- What is thematically prominent in a text or a collection of texts;
- What might be Register or style indicators;

If the main group of users are to be learners at least initially under the direction of a tutor, it would seem important for the tutor to be able to be assured that the scope of the texts matches the target genres of their students. Key word processes on entire corpus collections may reveal patterns for words like *the* and *of*, but from a language learning point of view, it seemed reasonable to add into the processes a means of screening out any word which occurs in every text in a corpus as it was hard to imagine these being psychologically prominent to a reader.

With these considerations in mind, the KeyTags and Key Associates features of *The Prime Machine* were developed and each of these will be explained in the sections below.

6.4 KeyTags

This section presents a new approach which uses log-likelihood contingency tables with Bayes Factors to create a list of key metadata and section labels (called KeyTags) which are then displayed using a tag cloud. Although connecting metadata to specific instances and using keyness in this direction is fairly innovative, there are other measures and processes which look at related features of language. Some work has been done using equally sized strips of text and comparing relative frequencies of words within one strip against the others (Liang, 2012). Liang's software is able to divide each of the texts in a corpus into strips and then to use key word statistics to show which words are key in each section of the text. The idea of looking at where words tend to occur is also related closely to the well-established concept of dispersion (Oakes, 1998). One way of showing the user how words or phrases are spread throughout texts and a corpus is through dispersion plots (Scott & Tribble, 2006). Other studies have explored the centrality and connectivity of specific nodes across chapters of book (Phillips, 1985), how repetition of lexis forms part of

cohesion (Hoey, 1991), and the way in which vocabulary across wide text windows can help identify topic divisions in texts (Biber, Connor, & Upton, 2007). Attempts have also been made to search key word databases for words with a specific pragmatic function in order to see whether their role in the text can be automatically identified (Scott, 2000). While dispersion calculations and key words on strips do provide some insights, corpora often have tags and metadata which could provide much more detail. It seems that in other concordancers these metadata are currently only used to filter searches rather than to examine the distribution of specific words and phrases under investigation. In the related field of information retrieval, while XML structure could help filter results, it is argued that search engines users typically do not know how to use XML structural instructions in their queries (Croft, et al., 2010). If native speakers and highly competent computer users find XML tags difficult to include in search queries, language learners and language teachers certainly will need more support.

However, as well as calculating which words are key in different texts, it may also be fruitful to develop a procedure to look at data from the opposite direction; that is to consider which texts or text sections are key for a word or phrase. This section presents a new method for corpus consultation which provides users with information about the typical contexts in which a word or phrase occurs. Using the frequencies inside and outside XML nodes, the system pre-calculates the typical environments so that a user searching for a word or phrase can instantly see what are called KeyTags: the XML tags which are statistically significant for the search term.

6.4.1 Calculating KeyTags

The log-likelihood contingency table which is used to rank and test the significance of the relationships is given in Table 6.4 below. It can be seen that this contingency table is formed by comparing the number of instances of a word or phrase within a text or section which is mapped to a metadata tag against the number of times the word or phrase occurs outside this context. The log-likelihood formula also balances this against the overall number of other words within the same context. A similar procedure is used to calculate KeyTags for multi-word units, where the frequencies are multiplied by the length in words of the multi-word unit, since each instance of a two word multi-word unit occurring within a metadata tag would account for two words from the total word count for that tag.

Table 6.4: KeyTags contingency table.

	Sub-Corpus 1	Sub-Corpus 2	Total
Node Word	Node word inside XML node	Node word outside XML node	Frequency of node word
Other Words	Other words inside XML node	Other words outside XML node	Frequency of other words
Total	Word count inside XML node	Word count outside XML node	Whole Corpus

As with the collocation and priming feature contingency tables, the log-likelihood formula and Bayes Factors given in Chapter 4 are used to calculate scores and degrees of evidence, and only items which occur proportionally more often inside the tags than outside the tags are stored⁵⁵.

Figure 6.11 shows the table structure which holds key tag data. Tables for collocations are similar, using the primary keys for each lexical item and the collocation type and linking this to the metadata item.

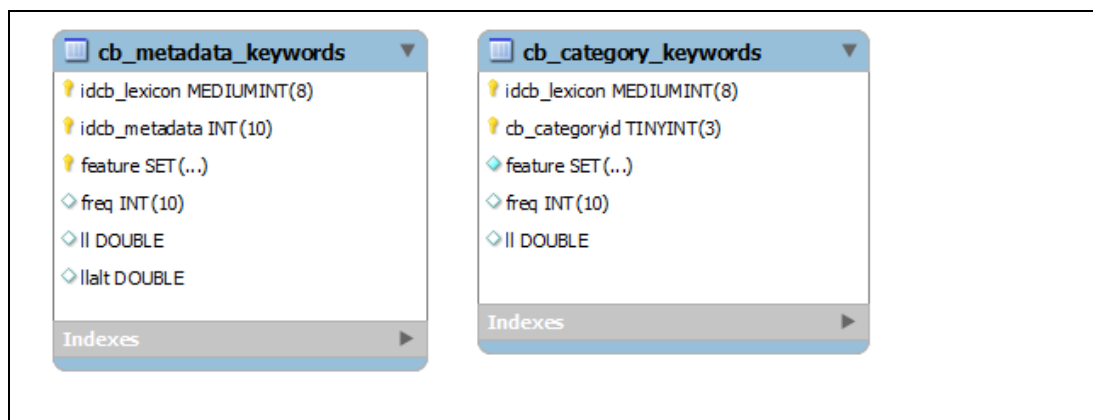


Figure 6.11: Table structure for key tag data.

The overall aim of this new concordance software feature is to provide additional information about the distribution of words and phrases to unsophisticated users of the

⁵⁵ During the early development of this approach, tendencies for words or collocations not to occur inside tags were also measured. However, the updated *SQL* scripts which generate these results no longer include these negative relationships. Although tendencies for words not to occur inside tags may be of interest to a linguist, results showing both positive and negative relationships could be confusing for a language learner and the focus in the software is on positive relationships.

system. The clouds are to be displayed alongside concordance lines and other summary data as a means of enriching the contextual clues available. If corpora include metadata that give indications of the function of specific sections of text, it could also be considered as a possible way to approach the automatic identification of what Hoey (2005) calls pragmatic association.

6.4.2 Examples of KeyTags

The junction-box tables used in the database to link metadata to elements of each corpus text (Figure 6.5) correspond to three groupings for display of KeyTags on the Tags Tab, and are distinguished in the KeyTags table using the “feature” data column. Text level and Section level form separate groups, but the third group which is called “Producer” contains links following two pathways through the junction-boxes, with KeyTags for text level authors merged with KeyTags for section level authors.

The Text level KeyTags can provide some indication of the tendency of a word or collocation to be used in texts from a particular set of sources in a corpus, or of texts of a particular type. Since the metadata mappings rely primarily on the tags which are provided in the corpus file headers, and also on decisions made during the refactoring process, it is not possible to stipulate whether these will be indicative of genre, register, style or the corpus sampling process. Essentially, as with the other KeyTags, when looking at Text level KeyTags, the user should try keep the following two questions in mind:

- Do these results suggest that the word or collocation is associated with a particular kind of text type?
- Do these results suggest that the texts which were chosen for the corpus are suitable for my purposes?.

A pair of examples for Text level KeyTags is shown in Figure 6.12. The KeyTag cloud of text metadata for *therefore* in the *BNC* provides (in descending order of keyness) “ACADEMIC”, “NON-ACADEMIC”, “W ac:polit law edu”, “Written Text”, “W commerce” and some general publishing or sampling information. As expected, this suggests strongly an association with written texts. The same search for *thus* gives “ACADEMIC”, “Written Text”, “NON-ACADEMIC”, “W ac:soc science”, “W commerce” and the publishing information, showing an even stronger tendency for use in Written Text.

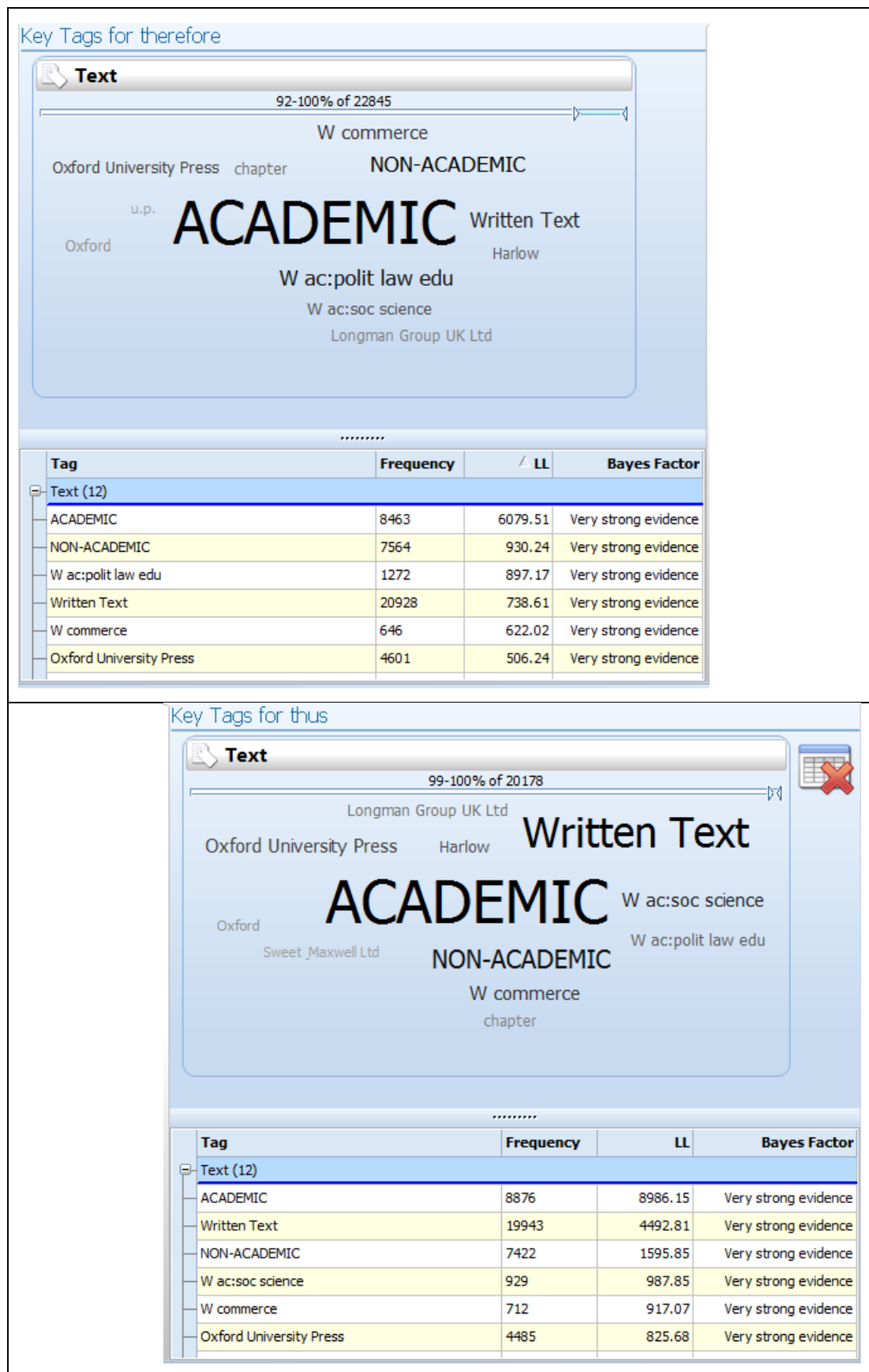


Figure 6.12: Tag clouds and tables for *therefore* (top) and *thus* (bottom) in the BNC: Complete.

The Text level results for KeyTags can also show how a word may have different meanings across different text types. Figure 6.13 shows the Text level results for *goal* in two sub-corpora of the *BNC*.

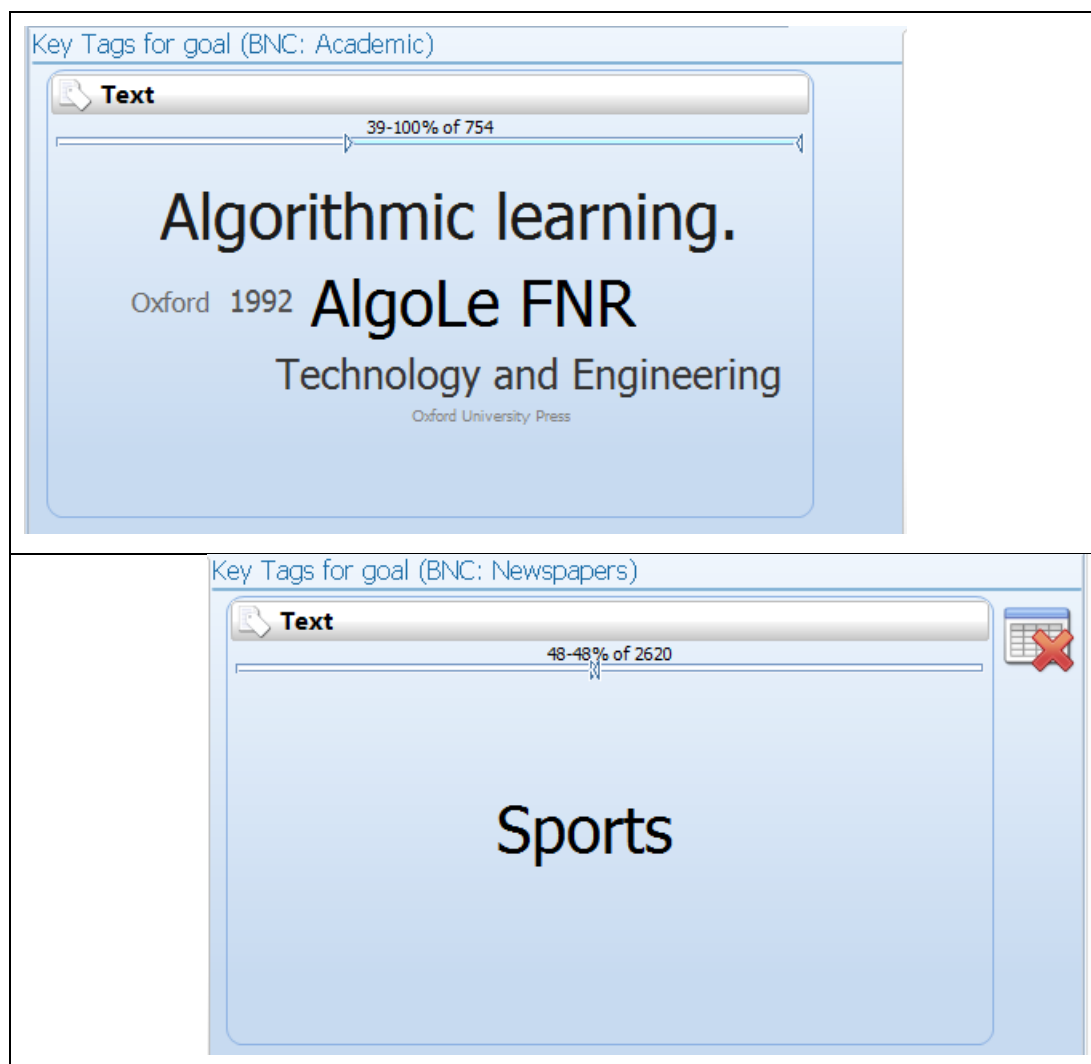


Figure 6.13: Tag clouds for *goal* in the *BNC: Academic* sub-corpus (top) and the *BNC: Newspapers* sub-corpus (bottom).

During the development of this procedure, some consideration had to be given as to how to help users interpret the significance of the KeyTags, and also how they should interpret “thin” or “empty” clouds. One way in which support for such interpretation is provided is through a range indicator which appears at the top of each cloud panel. The range indicator has a start and end arrow head showing the proportion of instances which are accounted for by the highest frequency tag leading up to the proportion accounted for by the combined frequencies of all the tags visible in the cloud. These values also appear as percentages above the range indicator, with the frequency of the search term also provided. The lower value is intended to provide the most cautious interpretation of the

cloud, showing the smallest possible coverage of the environments in which the search term would be occurring if all the other tags in the cloud were representing exactly the same set of concordance lines as the most frequent tag. At the other extreme, it is possible that all the tags represent unique instances of the search term in different environments with no overlapping, so the upper indicator shows this. The range indicators in the two clouds for *goal* (Figure 6.13) show quite different values, with the tags from the academic sub-corpus providing coverage of a large proportion of the instances, while in the newspaper sub-corpus only “Sports” is visible and it accounts for less than half of the occurrences.

Looking at Text level KeyTags often gives some information about the kinds of texts in which they occur, but the Section level KeyTags which are based on the sub-headings used in different sections of a text can give insights into aspects of text structure and the actual topics of the parts of the texts containing the word or collocation. For the *SpringerOpen* corpus, a search for *aim* produces the section KeyTags “Background”, “Abstract” and “Aim”. A few more section KeyTags can be seen in the table, but are not included in the cloud because their font size would be too small to be legible. It might be thought desirable to filter out KeyTags matching the search term itself, because each time a section heading is added to the corpus as text it guarantees increasing its identification with itself, but this issue needs to be considered more fully since users might find it helpful to see that a word they have searched for is often used as a heading. In the same corpus, the only section KeyTags for *goal* is “Background”. For this node, “Introduction” is below the strong evidence threshold and “Aim” is absent. These examples can be seen in Figure 6.14 below.

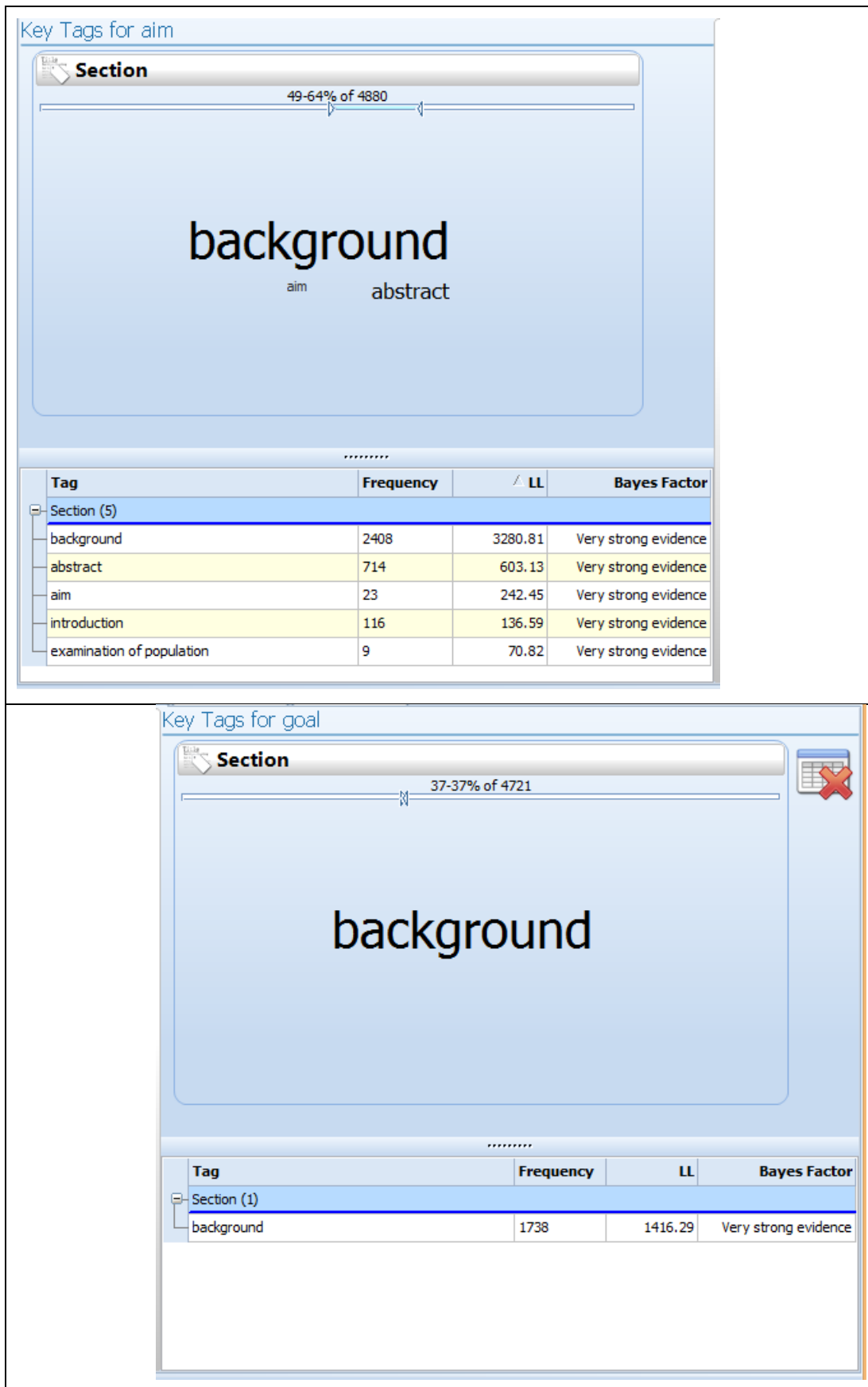


Figure 6.14: Tag clouds and tables for *aim* (top) and *goal* (bottom) in the SpringerOpen corpus.

Section level KeyTags can reveal how academic texts use words with a similar meaning in different sections of text, indicating a particular sense. In the *Hindawi Biological Sciences* corpus, the clouds shown in Figure 6.15 show how *important* differs from *significant*, with the latter clearly identified with its use in statistics. The range indicator at the top of both these clouds gives a sense of how well the Section level tags represent all instances of the node word.

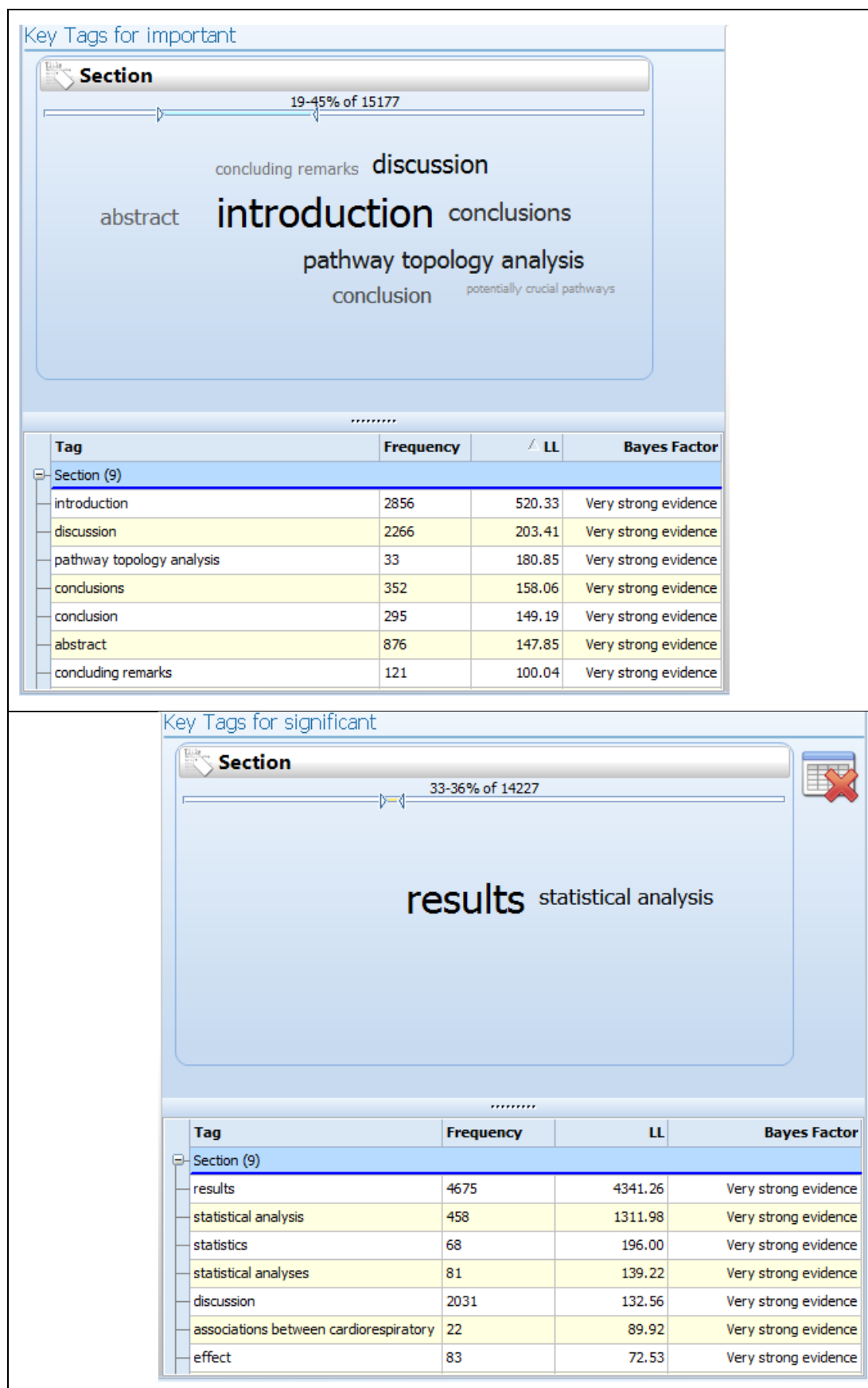


Figure 6.15: Tag clouds for *important* (top) and *significant* (bottom) in the *Hindawi Biological Sciences corpus*.

As well as text and section information, KeyTags can also include details about the author or speaker and these are shown in the software on a panel headed “Producer”. Items for this panel can include metadata about the authors or speakers of complete texts and also metadata about the authors or speakers for each section of texts. For single texts where a word or collocation appears very frequently, the author’s name and other metadata about the author may appear in this panel, complementing the information provided about the text which appears in the Text Tags panel. For example, the Text Tag panel for *marginal cost* in the *BNC* includes prominent tags for the title of the book “Economics”, “W Commerce” and “NON-ACADEMIC” as well as publishing information, while the Producer Tag panel shows the three names of the authors of this book: “Begg, David”, “Fischer, Stanley” and “Dornbusch, Rudiger”. Corpora of spoken texts tend to have more metadata available about the speakers and KeyTags for these are also shown in the same Producer panel. Some mixed results are produced for KeyTags in the *BNC* for *gosh* and *sorry*, and in Table 6.5 and Table 6.6 the results are given before and after the metadata tags had been adjusted for readability. The documentation for the *BNC* includes clear information about the meaning of each of the “ref-person” tags, and it is very straight-forward to use this information to create a short SQL script to alter the form of the metadata labels to make them clearer. For example, “Ag0” can be changed to “Under 15 years of age” by updating a single row in the database using an “update” command. An SQL script to make these updates once for all users can be generated using a template in *Microsoft Excel*. The original codes which are shown in brackets are provided here for reference, but after the SQL script has been run, the original codes are no longer stored in the database.

Table 6.5: Raw data for *gosh* before and after re-labeling the metadata tags for readability

Tag (original in brackets)	Freq. Inside	Freq. Outside	LL
Female (sex:f)	143	221	468.64
role:self (role:self)	94	270	318.91
Higher management: administrative or professional (soc:AB)	73	291	315.74
occupation:housewife (occupation:housewife)	52	312	209.19
age:70 (age:70)	32	332	192.42
Over 59 years of age (ageGroup:Ag5)	56	308	182.28
Social class unknown (soc:UU)	114	250	165.70
45 to 59 years of age (ageGroup:Ag4)	53	311	132.14
Male (sex:m)	82	282	121.57
dialect:London (dialect:London)	31	333	113.56
Lower management: supervisory or clerical (soc:C1)	37	327	111.01
London (dialect:XLO)	31	333	107.90
role:wife (role:wife)	26	338	106.65
xml:id:PS0W4 (xml:id:PS0W4)	13	351	102.82
persName:Margaret (persName:Margaret)	16	348	89.580
dialect:Home Counties (dialect:Home Counties)	23	341	79.847
age:57 (age:57)	15	349	79.78
occupation:export merchant (occupation:export merchant)	11	353	78.31
xml:id:PS05X (xml:id:PS05X)	11	353	78.31
35 to 44 years of age (ageGroup:Ag3)	33	331	76.70

For *gosh*, “sex:f” is in first position, suggesting females in the corpus use this relatively more frequently. A number of tags related to social class, occupation and age are also visible. However, on close examination, the seventh item in the list is a tag meaning the data about social class were unknown or not available.

Table 6.6: Raw data for *sorry* before and after re-labeling the metadata tags for readability.

Tag (original in brackets)	Freq. Inside	Freq. Outside	LL
Social class unknown (soc:UU)	3,115	7,588	4,109.53
No accent recorded (dialect:NONE)	2,342	8,361	3,354.53
role:unspecified (role:unspecified)	2,049	8,654	2,473.26
Unknown age (ageGroup:X)	1,822	8,881	2,342.41
Male (sex:m)	1,973	8,730	2,255.84
Unknown education level (educ:X)	1,796	8,907	2,099.60
British English (firstLang:EN-GBR)	1,281	9,422	1,625.35
Unknown gender (sex:u)	1,068	9,635	1,603.92
Female (sex:f)	1,268	9,435	1,262.05
n:W0000 (n:W0000)	668	10,035	1,106.14
persName:Unknown speaker (persName:Unknown speaker)	668	10,035	1,106.14
role:other (role:other)	669	10,034	1,101.07
45 to 59 years of age (ageGroup:Ag4)	719	9,984	835.46
role:self (role:self)	774	9,929	825.98
35 to 44 years of age (ageGroup:Ag3)	556	10,147	764.36
25 to 34 years of age (ageGroup:Ag2)	495	10,208	567.38
Higher management: administrative or professional (soc:AB)	406	10,297	503.48
Lower management: supervisory or clerical (soc:C1)	398	10,305	497.31
Home Counties (dialect:XHC)	355	10,348	488.79
occupation:student (occupation:student)	328	10,375	435.18

The results for *sorry* show a similar lack of information for some of the top tags, with (“soc:UU”, “dialect:NONE”, “role:unspecified”, “ageGroup:X” and “educ:X”). It would be important to consider whether tags like these should be excluded or whether information about the way in which the original data samples were gathered would need to be presented. Nevertheless, the tags for *sorry* also include some information related to gender, dialect, age and social class. “sex:M”, “firstLang:EN-GBR”, “sex:F”, an occupation and then “age:50+” and another occupation. Showing both genders within the same cloud may be confusing, but association with both is logically possible since only the spoken texts contain these tags. This is probably uninteresting except as another indication that the word is used in spoken interaction rather than written texts. When Key Tags are produced for *sorry* using just the *BNC: Spoken* sub-corpus, the gender of the speaker does not appear in the top 40 Producer Tags for *sorry*. This is further evidence that the association between the word *sorry* and gender tags in the complete *BNC* is mainly a result of an association with spoken texts and not with gender.

Just as Text and Section level KeyTags can sometimes suggest that the concordance lines may chiefly contain examples from a particular text type or on particular topics, the Producer level tags can show whether a word or collocation is mainly used by a particular kind of writer or speaker. With spoken corpora, it may be even more important for the user of the system to keep the two questions about KeyTags in mind, as exploration of certain features like hesitators or back-channelling may be more heavily influenced by the conventions used in transcription or variation in the transcription process. For example, Figure 6.16, Figure 6.17 and Figure 6.18 show clouds for *mm*, *hm* and *hmm* from the *BNC: Spoken* sub-corpus, where one might question whether all the transcribers handled *these* consistently across the different text types and across different speakers.

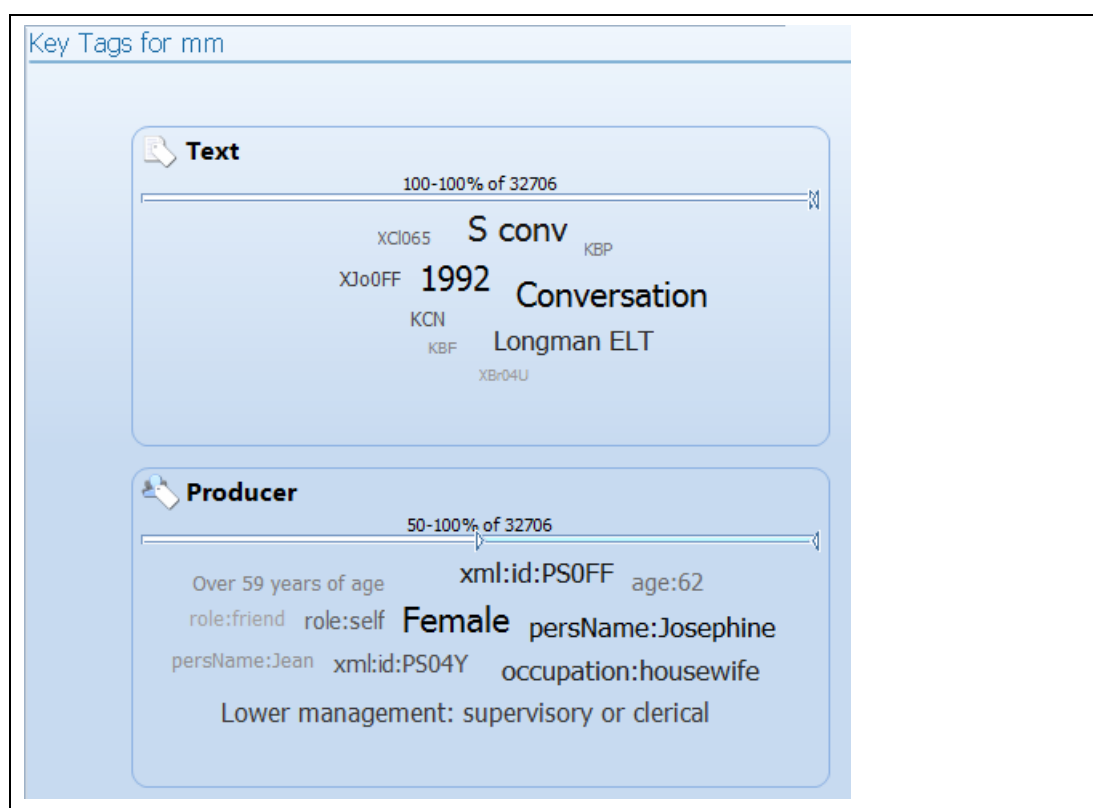


Figure 6.16: Text and Producer clouds for *mm* in the *BNC: Spoken Corpus*.

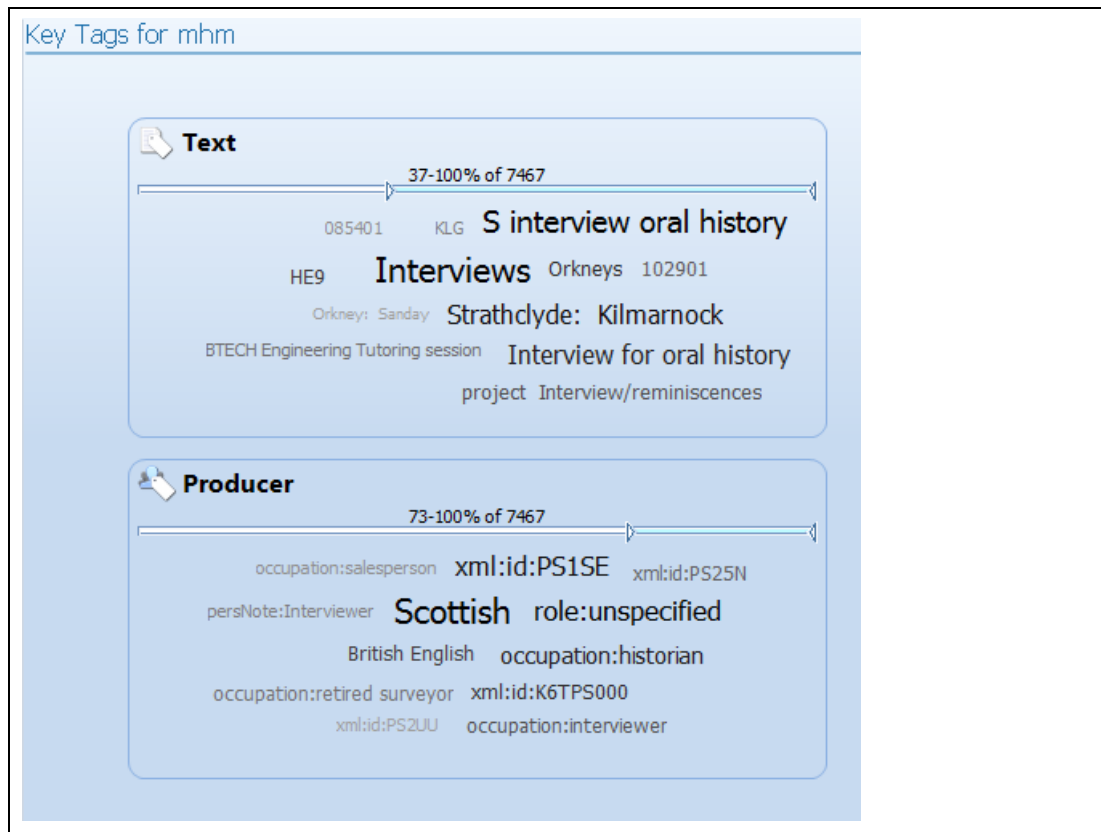


Figure 6.17: Text and Producer clouds for *mhm* in the *BNC: Spoken Corpus*.

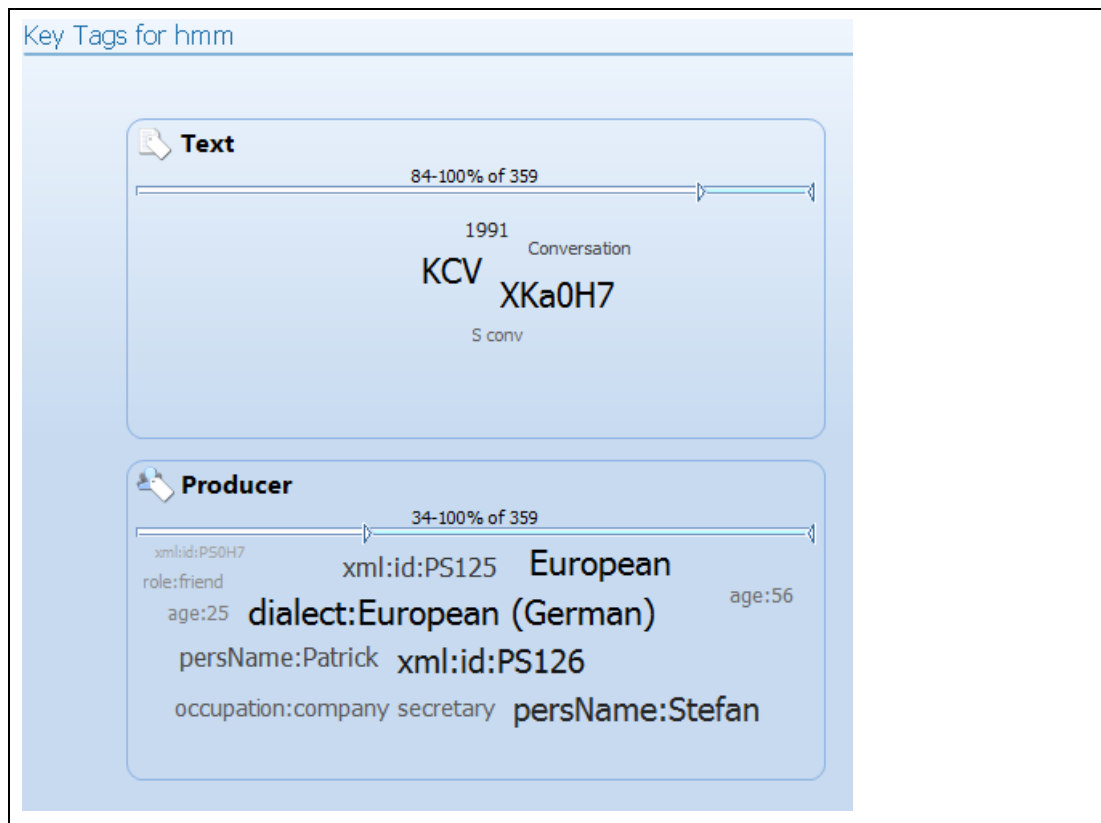


Figure 6.18: Text and Producer clouds for *hmm* in the *BNC: Spoken Corpus*.

On the other hand, as shown in Figure 6.19 below, certain kinds of discourse marker may correspond quite well to use by people in particular roles.

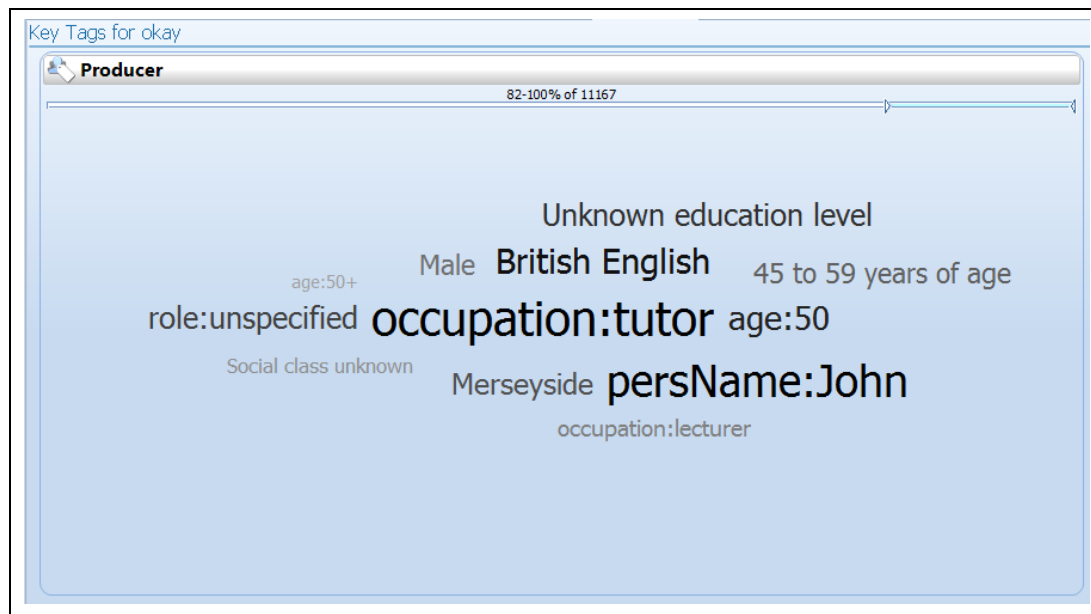


Figure 6.19: Producer cloud for *okay* in the *BNC: Spoken Corpus*.

Some of the challenges of dealing with spoken corpora are explored further in Chapter 8.

6.4.3 Potential uses of KeyTag results in language learning

Students may enjoy seeing an example like that shown in Figure 6.19 which demonstrates the frequent use of *okay* by lecturers and tutors, but there are a number of reasons why the kinds of additional information which the KeyTags procedure reveals could be useful for a corpus user, particularly users such as language learners and language teachers. Kreyer (2008) explains that when language learners consult a corpus, they may be able to see the top level differences in register such as major differences between spoken and written modes, but argues that sub-corpora divisions may not be so obvious. The example he gives is where all the instances of a word or phrase in the written mode actually come from correspondence texts. A native speaker may be able to look at the concordance lines and immediately see that the type of writing seems to be limited to letters, but a language learner may miss this and assume that the word or phrase is equally common in all kinds of written text. In *The Prime Machine*, major sub-divisions are visible on screen, so the attention of learners is drawn to these kinds of distinction. Figure 6.20 shows the information about KeyTags which is provided in *The Prime Machine* user manual.

Tags: Tags Tab

Sentences in a corpus often have labels attached to them giving details about the text, the section or the producer. A statistical measure can determine whether a word or phrase appears in particular kinds of text more often than expected by chance.

- Text tags include the main category of a text (e.g. *Fiction* or *Academic*). They may also include information about the publisher, the source or the genre;
- Section tags are the sub-headings of a text (e.g. *Abstract*, *Introduction* or *Conclusion*);
- Producer tags provide information about the writer or speaker (e.g. their name, age, gender)

The line and arrows above each box show the proportion of occurrences of the word which are accounted for by the tags visible in the cloud. If the percentages are very high, it means that tags in the cloud account for most of the concordance lines available. If the percentages are very low, it means the word occurs in many other contexts as well.

Detailed notes:

Since some occurrences of the word may be connected with more than one tag, the figures are shown as a range of values rather than a single percentage. The lower end of the range shows the proportion of occurrences represented by the most tag where the word has the highest frequency. The higher percentage shows the maximum proportion of occurrences which could be represented in the cloud since it gives the combined frequencies for all the tags the tags.

Figure 6.20: Information about KeyTags cache provided in *The Prime Machine User Manual*, Version 2.0, January 2015

In essence, the KeyTag display should provide useful information for both language teachers and language users by helping both of them to understand the composition of the corpus and the kinds of examples which will be displayed. For a language teacher the examples that a corpus can provide need to be judged not only in terms of the lexicogrammatical range, but also in terms of the appropriateness of the registers and text types represented in the corpus. A teacher using the KeyTag function would be able to quickly see which kinds of examples were most prominent in the set of concordance lines for the currently selected corpus and the teacher should be able to get a clear sense of whether it is balanced and whether it fits their intended target group. For language learners, KeyTags provide information about typical uses in terms of the major text categories, section headings and language producers. By looking at these, a student would be able to determine whether or not there appear to be any “prohibited” uses too. This is one of the ways in which *The Prime Machine’s* ability to display results for similar words and terms side by side aims to help learners select an appropriate term from a choice of near synonyms or different word forms.

As well as providing new kinds of data for corpus users, this approach also tries to bridge the gap between the sophisticated mark-up of modern XML corpora and visual presentation of KeyTags which might aid users in interpreting typical contexts for search terms.

6.4.4 Potential uses of KeyTags as search queries

The purpose of this section is to explain an alternative way of accessing data from the KeyTags tables and how this is implemented on the Search Tab. One benefit of the way in which KeyTags are stored in the database is that the same tables can be used to retrieve pre-processed key word calculations for a wide range of intra-corpus comparisons. The summary data can be retrieved using the tag as a search item, producing similar results to traditional key word analysis, but on a much wider range of tags. This is because the KeyTags summary tables store key words and key collocations based on dividing each corpus according to each of the metadata items held in each corpus. However, it is important to note that the KeyTags are not envisaged to replace or decrease the importance of direct access to concordance lines. Following Rayson (2002) and Baker (2004), just as key words are argued not to be an end in themselves but rather a starting point for investigation, KeyTags should also be a starting point for actual examination of the contexts of concordance lines. KeyTags could be considered as a navigation tool and indeed their efficient storage means that key words can be extracted to form an additional means of navigating the corpus, building on the principle of providing additional assistance to learners so they can find useful starting points for analysis (see Chapter 3). On the Search Tab, they are able to navigate through the metadata available and look at key words.

The key word lists for the major categories from the *Hindawi corpora* can often show clear divisions of topics across different academic disciplines as would be hoped and expected. Similarly, some of the key word lists for sub-sections of the *BNC: Newspapers* sub-corpus and the *BNC: Academic* sub-corpus give a clear indication of what the different categories of text are about. For some categories in both *BNC* sub-corpora and the *Hindawi corpora*, however, results are less intuitively related to topic and more likely a consequence of stylistic features, the use of quoted speech and other elements in the texts.

Key words and key collocations can be displayed for any of the tags held at Text, Section or Producer level. Figure 6.21 shows how the Auto-Complete function offers some of the tags which begin with the same string of letters and the top key words and key collocations for the currently selected tag. Individual words which are key for the currently selected tag are

displayed in a box to the left, while a separate box shows collocations which are key for the same tag. To encourage users to explore the actual concordance lines, the checkboxes are available to the left of each list and one or two of these can be used as the basis of a concordance query using the “Search” or “Compare” buttons. The keyness statistics are not visible in the drop-down lists, but, as will be explained below, they can be displayed in a table.

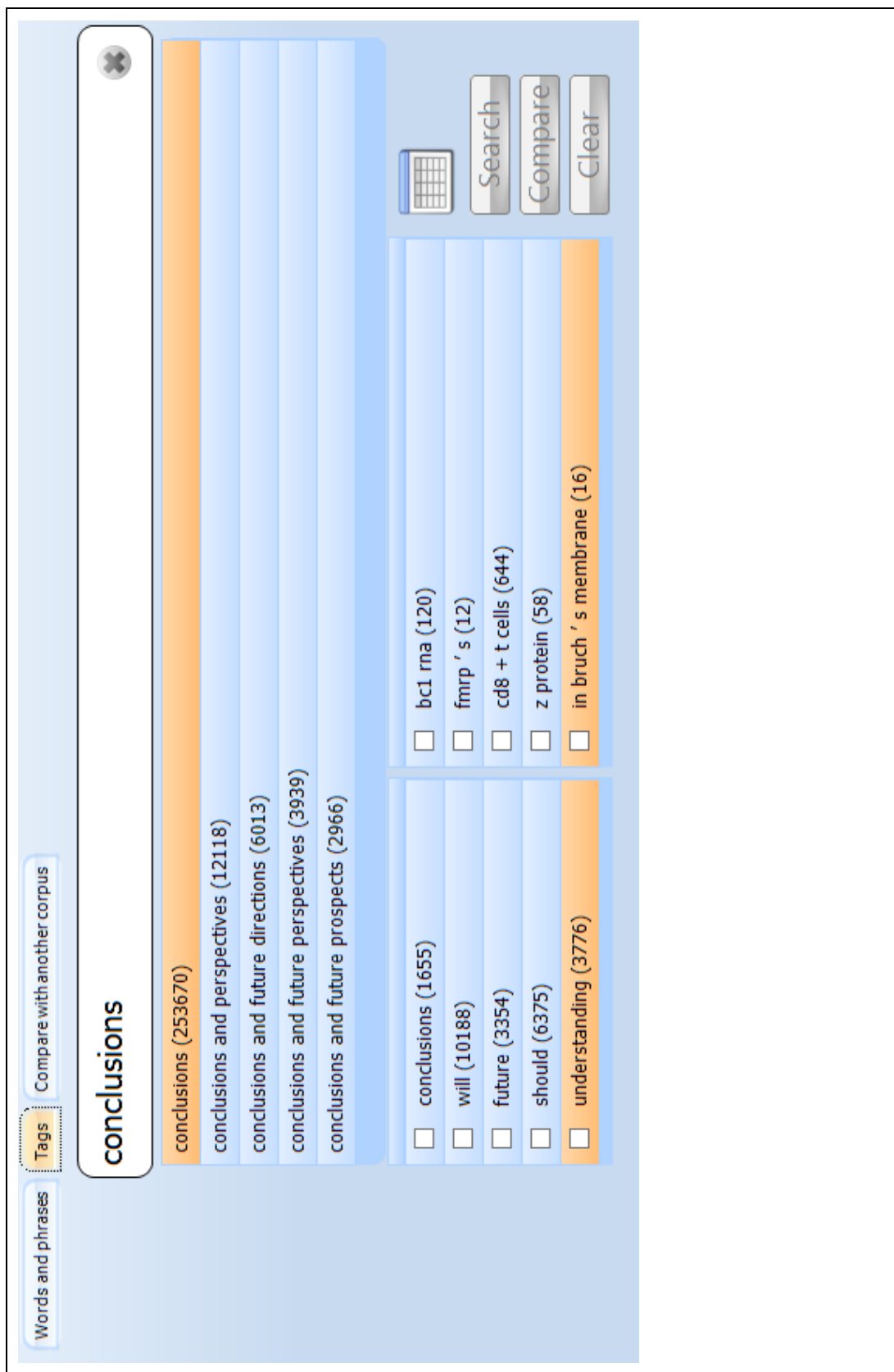


Figure 6.21: The auto-complete functionality of the Tags sub-menu on the Search Tab, with incidental data from the *Hindawi Biological Sciences* corpus.

An important point to note is that while the words are sorted in descending order according to keyness, the numbers in brackets are the frequencies in the whole corpus. Currently, although the middle tier server can apply further filtering to concordance lines, there is no mechanism in the client application to formulate the instructions for this, so requests for concordance lines which occur within texts or sections matching the metadata tag cannot be carried out. As explained in Chapter 5, Section 5.6, activating this facility would involve developing a means of showing current filters on screen, and could be something to be considered for future development, but since no summary data would be provided on this narrow basis, it is not something which has been prioritized for development at this time.

Since the Auto-Complete boxes are fairly small, a longer list of key words and key collocations with keyness statistics can be retrieved by clicking on the table icon (visible to the right of the list in Figure 6.21 above). Table 6.7 shows some of the exported results from the table which appears if a user requests more detailed results for a tag search on *conclusions* in the *Hindawi Biological Sciences* corpus. As can be seen, just as with key word lists for texts, the section level key words for *conclusions* include some features which match expectations and other items which could not readily be explained without analysis of individual concordance lines and perhaps require some specialist knowledge for the academic discipline. Nevertheless, it is interesting to consider this kind of key word analysis as providing a complementary way of accessing information about how words and collocations tend to be used in particular sections of a text. In this sense it complements the “position in text” features on the Graphs Tab. The top item in the table indicates that *conclusions* is frequently used as a section heading. Items such as *our* and *we*, indicate how academic writers use these pronouns in the *conclusions* section of a text. Items such as *future*, *should*, *may*, *further*, *can* and *could* also indicate some of kinds of speculation and forward-looking language which appear in a conclusion.

Table 6.7: Key words and collocations for *conclusions* in the *Hindawi Biological Sciences* corpus

	Key Word / Collocation	Frequency Inside Tag	Frequency Outside Tag	Log-likelihood	Bayes Factor
1	conclusions	1045	1655	7265	Very strong evidence
2	will	459	10188	614	Very strong evidence
3	future	245	3354	526	Very strong evidence
4	bc1 rna	38	120	390	Very strong evidence
5	should	274	6375	347	Very strong evidence
6	understanding	198	3776	313	Very strong evidence
7	may	728	32531	299	Very strong evidence
8	fmrp	56	199	272	Very strong evidence
9	fmrp ' s	10	12	238	Very strong evidence
10	our	422	16527	235	Very strong evidence
11	further	342	12182	229	Very strong evidence
12	can	765	38624	224	Very strong evidence
13	could	427	17524	215	Very strong evidence
14	cd8 + t cells	33	644	204	Very strong evidence
15	provide	198	5399	204	Very strong evidence
16	new	272	9184	201	Very strong evidence
17	z protein	19	58	198	Very strong evidence
18	research	229	7069	196	Very strong evidence
19	novel	163	4036	191	Very strong evidence
20	mechanisms	256	8722	186	Very strong evidence
21	bc1	38	135	185	Very strong evidence
22	we	736	39830	171	Very strong evidence
23	in bruch ' s membrane	6	16	166	Very strong evidence
24	need	134	3170	166	Very strong evidence
25	5	445	20731	164	Very strong evidence
26	in hawaii	17	65	160	Very strong evidence
27	fmrp '	10	12	159	Very strong evidence
28	important	352	15176	158	Very strong evidence
29	better	142	3715	155	Very strong evidence
30	studies	500	24792	155	Very strong evidence

6.5 Key Associates

In Chapter 4, there was some discussion about the optimal size for a window in the calculation of collocations. One possibility would be to widen the window to include the complete text and use the same contingency table to measure the likelihood of words occurring in the same text compared to their frequency in all the texts which do not contain the node. Garretson considered the possibility of incorporating textual collocations into his lexical profiling system, but decided against it for several reasons: the sampling technique used in many corpora, the difficulties of deciding whether to limit the notion of one "text" by chapter or entire book, and technical issues related to the size of the XML

summary data his system would create and the time it would take to generate them (Garretson, 2010). In terms of the incompleteness of some texts in corpora like the *BNC*, it could be argued that although a sampled section of a text will not be perfectly representative of an entire text, just as with textual colligation (Chapter 5) if a measure can be based on selections from many texts, patterns should provide some interesting data. The second question of how to define a text for these purposes and whether or not crossing chapter boundaries should be permitted is also a question for corpus compilation in general. For example, it is not always very straight-forward to divide internet derived corpus data from a website into separate texts.

Another problem with text level collocations is that as the window size for potential collocates increases, as with several key word procedures, it seems that the collocation list begins to resemble a raw frequency list. However, Scott introduces key associates as another procedure which can capture a sense of how one word is related to uses across different texts (Scott, 1997; Scott & Tribble, 2006). Rather than analyzing words that occur together in any text in the corpus, this procedure counts the number of texts in which both words are key words. Scott defines associates as “the set of words which are co-key with a given KW-node across a range of texts” (Scott & Tribble, 2006, p. 85). In the procedure Scott describes, key associates are only calculated for words which are key key words. Scott suggests that associates can give an indication of stereotype and demonstrates how these can be re-grouped into clumps (Scott, 1997) or analyzed within sub-corpora by domain and variety (Scott & Tribble, 2006).

In order to provide some indication of the kinds of topics and themes which were important in the texts in which the node or collocation were important, the generation of key associate data was incorporated into *The Prime Machine*, splitting each corpus according to the major categories. As explained in Chapter 4, only very limited information about semantic association is presented to the user, and this procedure was added to the system with the expectation that associates can go some way to showing some of the stereotypes and topics which may be associated with certain words and collocations beyond four word window boundaries. Just as it is possible that semantic prosody may extend beyond the nearby environment, showing key associates should help users of the system to see how the wider phenomena within semantic association may also extend to a text level. Language learners ought to benefit from being able to gain information about the kinds of words which occur with a node in a wider context. This information can give

them a sense of what kinds of things tend to be discussed from the perspective of the whole text, and also indicate some of the bias or common themes which can be discovered in the corpora too. In *The Prime Machine*, the requirement for words to be key key words has been loosened, and key associates are processed for all items which occur as a key word or key collocation in at least one text.

Before calculating key associates, key word lists need to be generated. As has been discussed above, when calculating key words, the composition of both the study corpus and the reference corpus may be different for different purposes. In *The Prime Machine*, the function of key associates is to give some indication of how other words and collocations are used in the same texts within the same main categories; that is to say, the key word lists generated for the calculation of key associates are specific to each individual text in the corpus, but the reference corpus used to compare relative frequency is comprised of the frequencies in texts from other main categories and the display of key associates is primarily grouped according to main category. If a corpus only has one main category, the reference corpus used is one of the other corpora held on the database server. Currently, only one such reference corpus is stored at a time, and an *SQL* script can be run on a corpus after the refactoring process has been complete to copy word frequencies and n-gram frequencies to be used as this reference corpus. In order to provide a good range of text types and a large number of n-grams, the *BNC* is currently used for this purpose.

As well as key words, key collocations are also calculated in a similar manner. However, the number of these is relatively small because they must occur at least twice and have a BIC of at least 2 in each text in order to qualify. Key Associates are then calculated by running through the entire lexicon and all the lists of collocations and finding the top 20 words or collocations which occur as key in other texts. After potential key associates have been extracted for key words and key collocations, a further data cleansing process is run in order to prevent two or more elements from a longer collocation appearing multiple times in the shortlist. Figure 6.22 below shows the table structures in the database for key words and key associates for single lexical items. Similar tables are created for key collocations and key associates for these collocations.

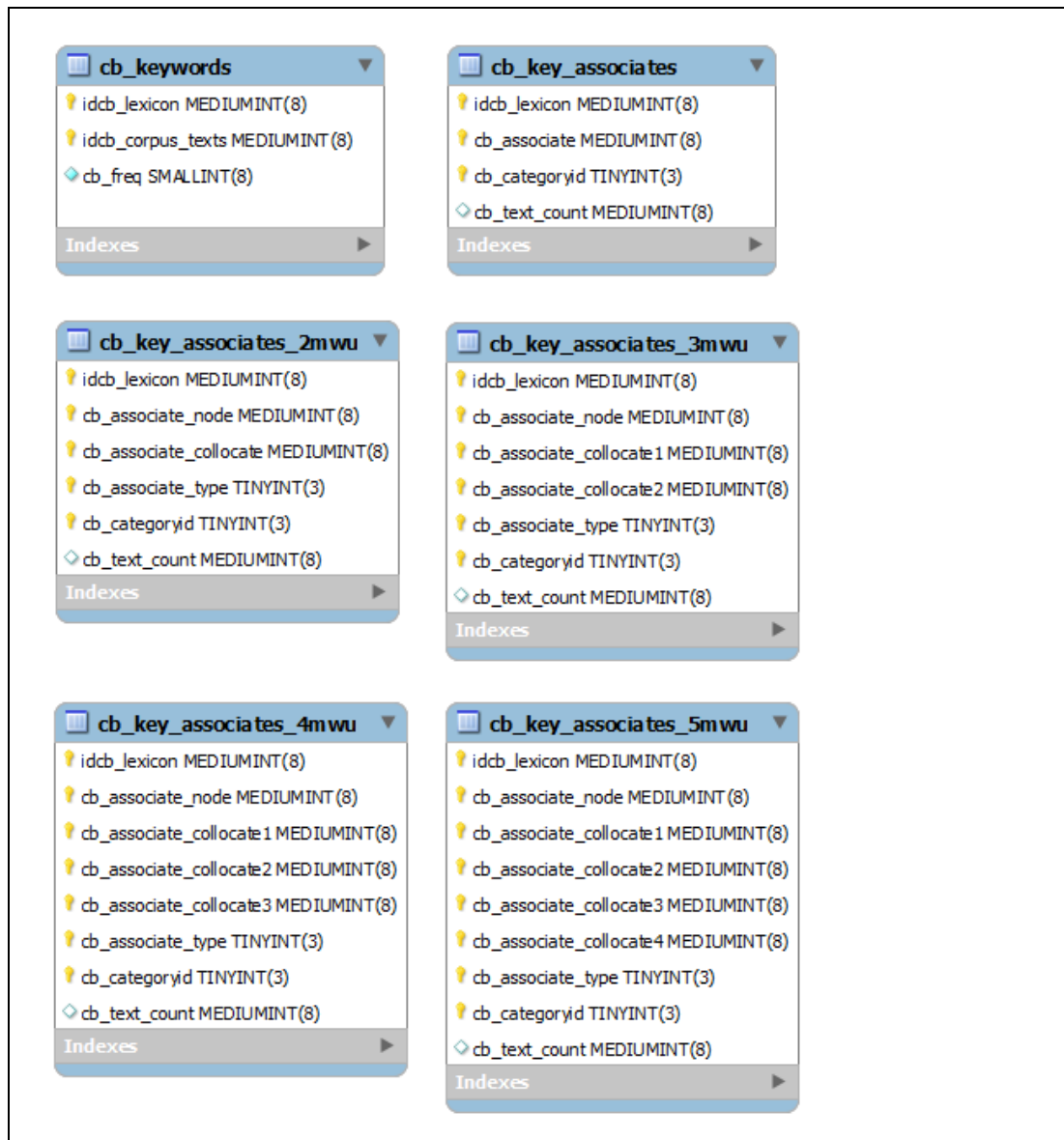


Figure 6.22: Table structures for key words and key associates.

The Associates Tab provides both a quick overview of the two most common categories for the search term as well as clouds or tables containing the key associates from each of these. A third box contains key associates from the other categories, if the text counts for these bring them into the list of top 20 rankings. A fourth box appears if the search term does not occur at all in one or more categories. The apparent avoidance of a word or collocation by language users in a specific text category could provide very useful information to a language learner since appreciation of genre and register is not just a question of using common terms for these, but also a question of avoiding using certain words or collocations in certain contexts. Since major categories usually have very different word counts, as well as showing the percentage of instances, an indicator shows the “norm”

which is actually the proportion of tokens which is represented by each major category. Summary information is stored in the database for each word and collocation according to the proportion of occurrences in each major category where this is greater than 10% or if it is 0. This provides a quick indication of the distribution of the query word or collocation across the major categories. It can also show how a word may have different meanings across different text types. Figure 6.23 shows the results for *pilot* in two different sub-corpora of the *BNC*.

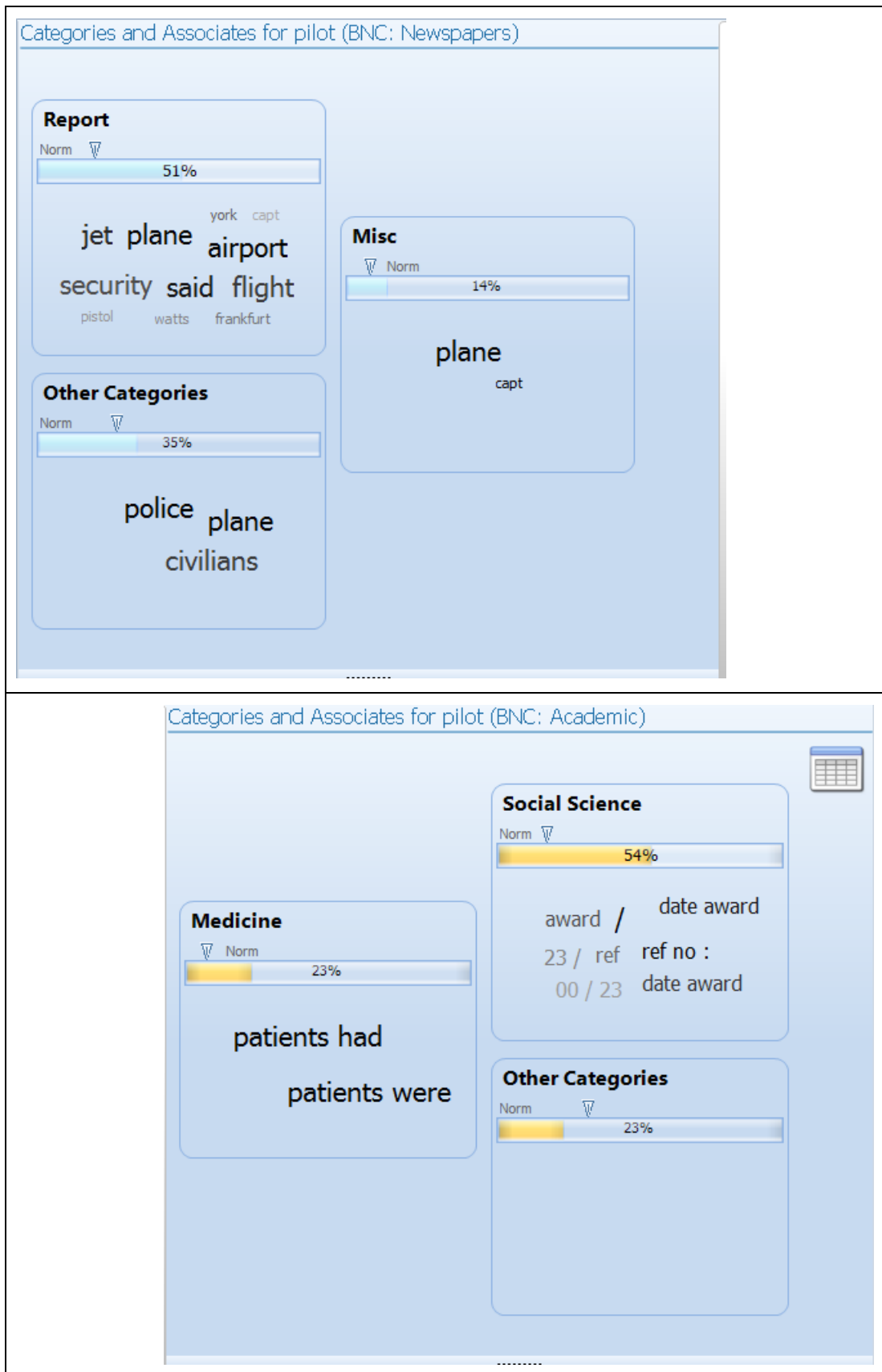


Figure 6.23: Key Associates for *pilot* in the *BNC: Newspapers sub-corpus* (top) and the *BNC: Academic sub-corpus* (bottom).

As can be seen, in newspapers, *pilot* most frequently occurs in the “Report” texts, and each box shows a clear relationship with the aviation sense of the word. From the results for academic texts, it would seem that *pilot* is particularly important for “Social Science” and “Medicine” texts, but while the “Medicine” box shows a clear relationship between discussions of *patients*, the “Social Science” box suffers from an association between texts containing details of grants and awards and those reporting on a pilot study. Nevertheless, some interesting links between *pilot*, categories and associates can be seen.

In order to provide information about the composition of each corpus in terms of the major categories, the Corpus Info Tab displays information in the form of a graph, with HTML text which can link to a website for more details. Figure 6.24 shows the information for the *BNC: Newspapers* sub-corpus.

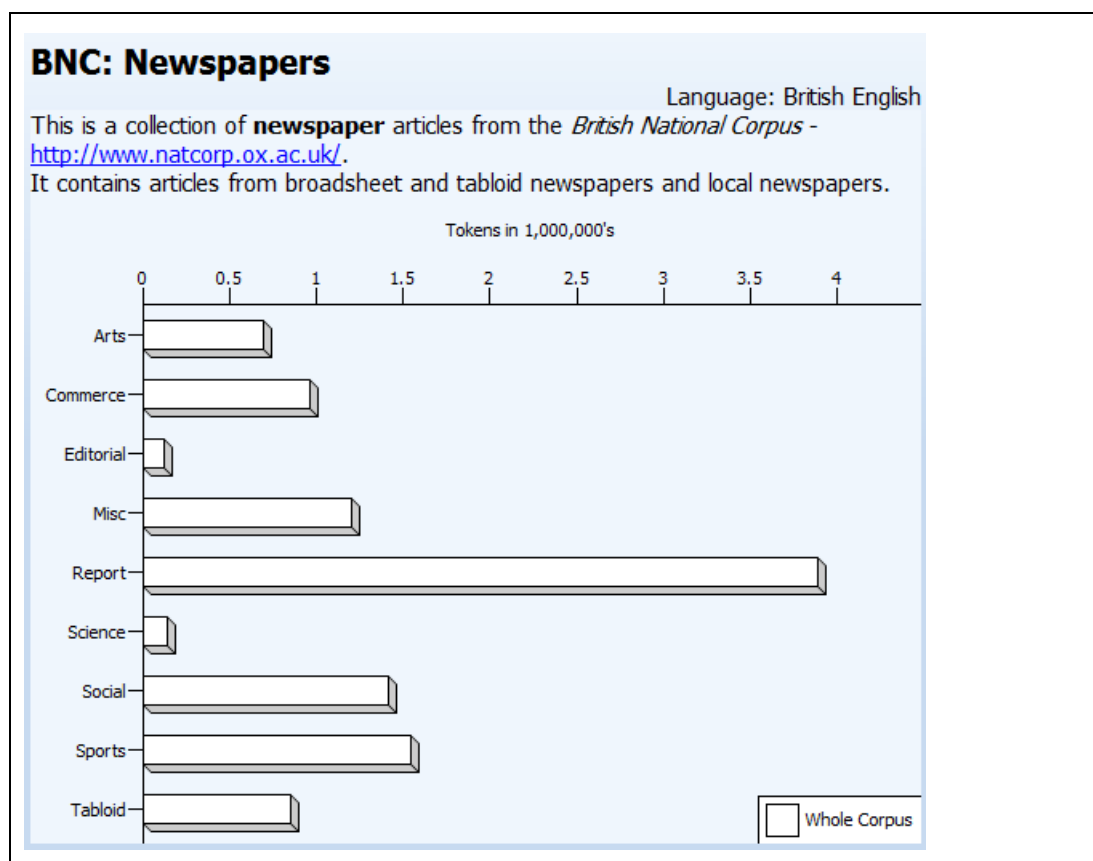


Figure 6.24: Information about the corpus and the major categories which is provided on the Corpus Info. Tab.

Figure 6.25 shows how the categories of text and the Key Associates can give indications of the topics associated with the word *drugs* across two very different corpora.

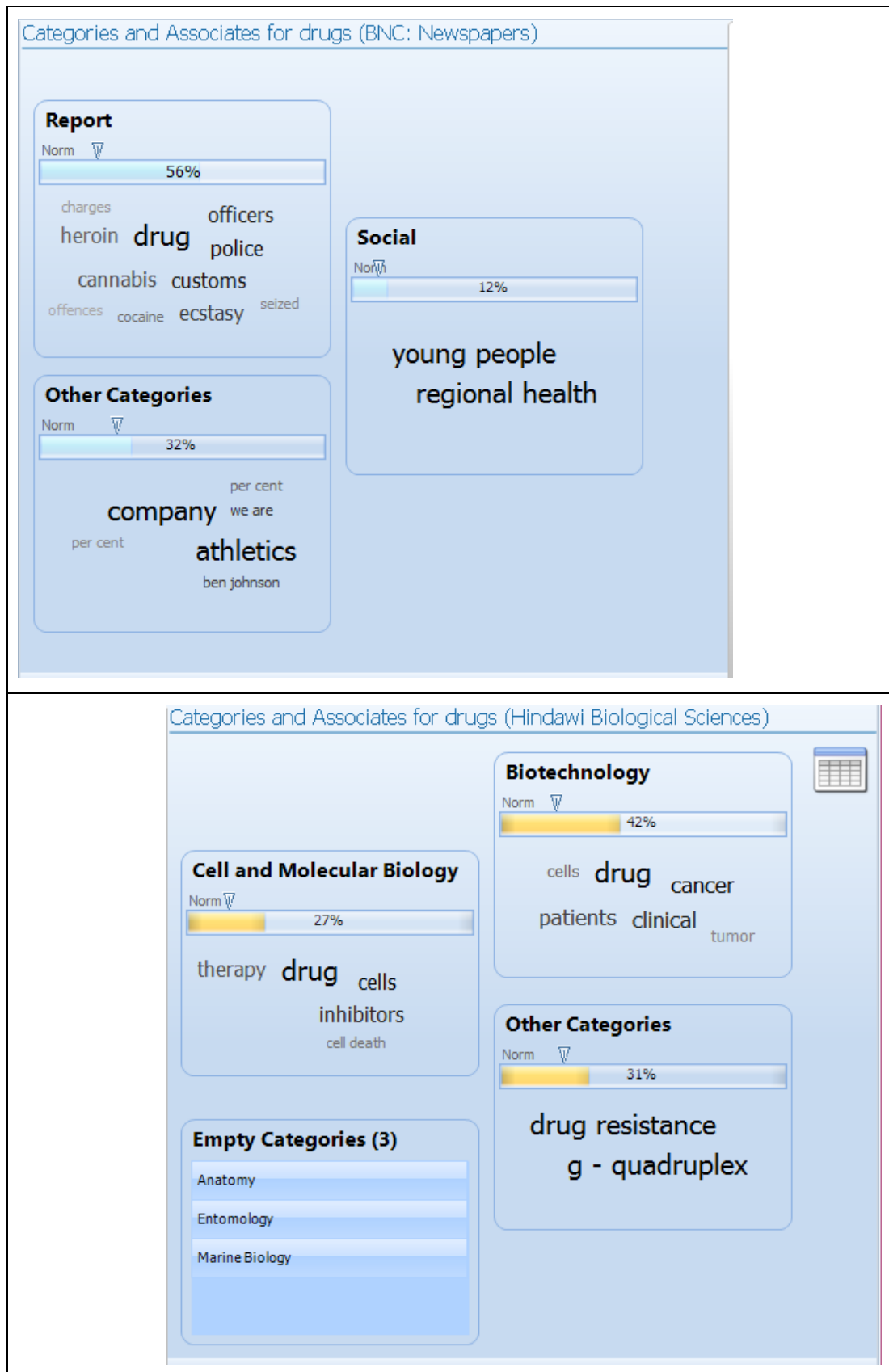


Figure 6.25: Key Associates for *drugs* in the *BNC: Newspapers sub-corpus* (top) and the *Hindawi Biological Sciences corpus* (bottom).

Summary

This chapter has introduced how the labels and tags which are provided in corpus texts are used and displayed. It has explained how major categories were assigned to the texts in the corpora which have been tested so far, and how information about the source of each concordance line is made available to the user. It has also considered some of the issues surrounding key word analysis, presented a new application of this technique named KeyTags, and introduced the use of Key Associates in the project. The two Tabs of results described in this chapter make it possible for language learners and teachers using *The Prime Machine* to explore the patterns of words or collocations occurring in texts or sections labelled with a wide range of metadata, and as they occur with other words and collocations in different text categories. This brings an end to the introduction of the main tabs of information provided in this software. The following chapter will introduce a few more of the design features which facilitate monitoring and configuration of the system, before presenting the evaluation of the software as a whole.

Chapter 7: Evaluation

The last three chapters have introduced some of the main features of the software, providing a pedagogical and linguistic rationale for the design as well as examples. The purpose of this chapter is to introduce a small scale evaluation of *The Prime Machine*. First, the scope of the evaluation will be introduced and the specific research questions explained. Then, several features of the software which have been designed to support evaluation will be explained. Some of these features were used in the evaluation which has already taken place; others have been put in place to support potential future evaluations. After that, the methodology, results and analysis of the evaluation will be presented. The final chapter which follows will consider how the results relate to the larger aims of the project, providing evidence for priorities in on-going software development, as well as areas for future evaluation beyond this current PhD project.

It is possible to evaluate a piece of software like *The Prime Machine* by carrying out a series of system evaluations or by conducting a user evaluation. With the innovations this project has introduced in the refactoring process, several new methods for statistical analysis of different features of text and its unique interface, a system evaluation would take the form of a series of tests checking each link of the chain and considering how these stand up in terms of precision and recall, as well as speed and efficiency and so forth. The results of such evaluations would be used to enhance each part of the system and ensure that users are presented with accurate and complete results. A user evaluation, on the other hand, considers how well the system meets the expectations of its users, how performance and accuracy affect the attitudes and actions of the users, and these can be measured through both feedback mechanisms such as questionnaires, interviews or focus groups, and through looking at the preferences expressed in records of users' interactions with the software. Following a user evaluation, priorities for further development become clear as software engineers can focus on ways to build on the more positively viewed aspects of the software, or they can look at which parts of the system were underappreciated or neglected and use system evaluation techniques to focus on these in isolation and attempt to improve them.

As will be evident from the previous chapters, the development of this software has involved considerable time and reflection devoted to processes looking at adopting and extending existing methods in corpus linguistics to support language learning as well as trying to make results meaningful and helpful for language learners. Chapters 3, 4, 5 and 6

have addressed the first research question, introducing methods for storage and retrieval of information about the contextual environments of concordance lines according to some of the features of Lexical Priming. As will be evident from these chapters, the development of this software has also involved considerable time and reflection devoted to processes looking at adopting and extending existing methods in corpus linguistics to support language learning as well as trying to make results meaningful and helpful for language learners. These chapters have also addressed the second research question which was concerned with pedagogical considerations, language learning tasks and the user-friendliness of the software.

The user evaluation presented in this chapter, therefore, focuses on the third research question:

To what extent can these methods provide language learners with examples that they find useful and provide them with insights about language usage which they find helpful?

In order to address this question, this chapter reports on attitudes of language learners who used the software in a language learning activity. The research question has been broken down into several smaller questions, and the evaluation considers the following:

- 3.1 Can the students find examples which they find helpful?
- 3.2 Which kinds of information do they look at most? How many results do they look at?
- 3.3 Which of the search support features are used most frequently?
- 3.4 How do they feel about the software? Would they want to use it in the future?

7.1 Evaluation considerations as design features

Before providing details of the methodology used in the evaluation, details of a range of software features which were designed to facilitate and enrich the current evaluation and were put in place for future evaluations will be introduced. While the majority of these features have a direct benefit for the user of the software, they have all been devised to also provide forms of data which would enrich data collection for specific evaluation projects.

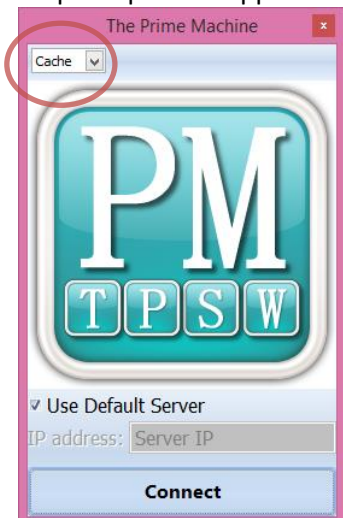
7.1.1 Authentication and user settings

There are some features of the concordancer software which rely on the system being able to identify the user and access data which they have stored locally. For example, in order to hold information on the local computer about past browsing history, it is important to be able to distinguish between different users – especially since it is envisaged that the software would be used in such shared computer environments as computer labs.

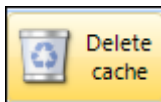
Some information is stored in the *roaming* folder of the current user. Figure 7.1 below, shows how information about this is presented in the user manual.

The application saves a small amount of data in your user “roaming” folder. If you wish to remove all these files, it is recommended that you open the application one last time and select the **Delete cache** button from the **Cache menu in the top-right hand corner of the Connect Screen**. After doing this, you can safely close the application using the **X button** and delete the **.EXE file**.

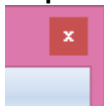
Step 1. Open the application.



Step 2. Select Delete Cache from the Cache menu.



Step 3. Close the application.



Step 4. Delete the .EXE file.

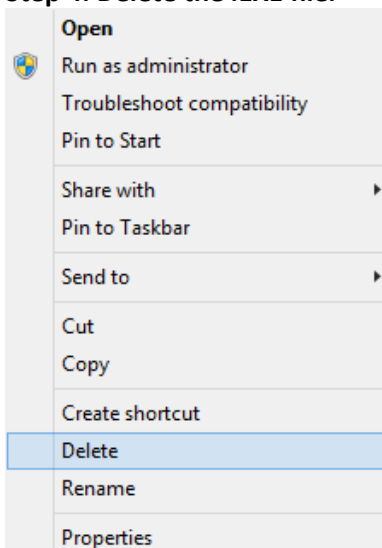


Figure 7.1: Information about the cache provided in *The Prime Machine User Manual*, Version 2.0, January 2015.

Although there is less fine tuning available in the software for configuration of some of the calculations compared to other concordancers, there are still a variety of options which users are likely to want to set and to have remain set for their future uses. For example, a student wanting to work with random samples of 100 concordance lines to explore several words or collocations may want to keep the settings for the number of concordance lines and the concordance ranking. Font sizes, colour preferences and other features are also configurable and hopefully add to the students’ sense of ownership of the software and

cater for different individual preferences. For an evaluation, however, it is even more important to be able to keep accurate records of which participants took which actions.

The table of data which is used for user settings could have been stored locally, but the records are saved in the database on the server for two main reasons. Firstly, students may want to use the software on machines which are not connected to the same local area network and synchronization of their roaming profile may not be automatic or possible across various different parts of a university network. Secondly, user-settings can also be tailored for specific groups on the server. This would seem like a good way to introduce “versioning” as a future evaluation approach. Rather than comparing the performance and attitudes of users of new software against a control group with no access to the software, versioning would allow an evaluation of different settings and would permit control groups to have access to at least some features rather than nothing at all (Cobb, 1997).

With an evaluation phase lasting several days, there would be the risk that other students might use the software or outsiders might try to gain unauthorised access, so for the evaluation described in this chapter and as a provision for the future there was a need to find a reasonably secure method for identifying each user of the system, allowing access to those who have permission and storing feedback information for those who are participating in the project. The issue is a fairly standard question of how best to implement a form of authentication.

The Datasnap technology (Delphi_Enterprise, 2010) which lies beneath all the client to middle tier server communications in *The Prime Machine* architecture does provide some level of security and the possibility for authentication, but one of the limitations of the version used in the development of this project is that it relies on a shared secret⁵⁶.

Modern life in this internet age seems to require more and more passwords, and with ever more systems requiring more passwords there is a danger that users will either forget their password, or use the same password for several different systems. From a security point of view, it was important to try to avoid a situation where authentication data would need to be sent backwards and forwards over the Datasnap connection. One means of avoiding the need to store hashed passwords in an application’s own database is to link to open authentication services. At the time this aspect of the software was developed, version 2

⁵⁶ More robust authentication systems can handle a secure exchange of password information rather than rely on a fixed shared secret which is embedded in the client and server applications.

of OAuth (Internet_Engineering_Task_Force, 2012) had just been released. Although this kind of protocol can be integrated more easily into a website environment, it could have been possible to implement an authentication solution based on OAuth into the software for this project, but there were a few problems. Firstly, it requires all the participants to have accounts with one of the partners. There were also news reports that providers (Yahoo vs. Google vs. Facebook, etc.) had not standardized with the new version and slightly different code would have been needed for each provider.

For the evaluation presented in this chapter, all the participants were enrolled at the same university, had university accounts and were users on the university's *Moodle* system⁵⁷. Virtual learning environments like *Moodle* and other internet based services like *Microsoft SharePoint*⁵⁸ operate in a similar way when a user tries to open a page which has restricted access and requires prior authentication; the browser will be redirected to the login page for the service and upon successful authentication, the original target page will be loaded. With evaluation purposes in mind, a robust but flexible way to authenticate users was designed to take advantage of this browser redirection by using a pop-up web browser within the client application, requesting a landing page which requires prior-authentication and then extracting some information from that page once it has loaded. Access to the software is therefore granted to users by adding them to a course page on *Moodle*, or a site on *Microsoft SharePoint*, or by setting these courses or pages up to accept new enrolments from within an institution.

The "Sign In" screen and user manual tries to explain how this works as transparently as possible. Users of similar systems should be aware that pop-up internet browsers can be monitored, and in theory a *Delphi* programmer could extract the username and password from the pop-up window. When trialling the software using the University of Liverpool's *Apps Anywhere* system (which allows software to be run on its computer systems remotely) it became evident that an embedded browser is considered a security risk by certain anti-virus software systems if the software is downloaded from a website rather than copied from a local folder. The source code for *The Prime Machine* which handles the authentication has been put into a separate unit and each step of the process is carefully labelled with comments in the code to demonstrate its safety. In order to make it clear


⁵⁷ www.moodle.org

⁵⁸ www.office.microsoft.com/en-us/sharepoint

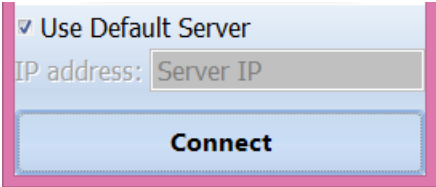
that the username and password entered on the pop-up window would not be not tampered with in any way, the username is not extracted from the login page, but rather taken from the downloaded course page, where the name is provided in plain text. Figure 7.2 below shows how information about this is presented in the user manual.

Each time you open the software, you will need to **log in**. There are different **Authentication Options** for different sites.

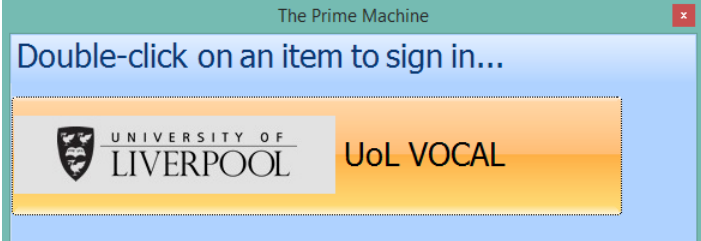
Step 1. Open the application.



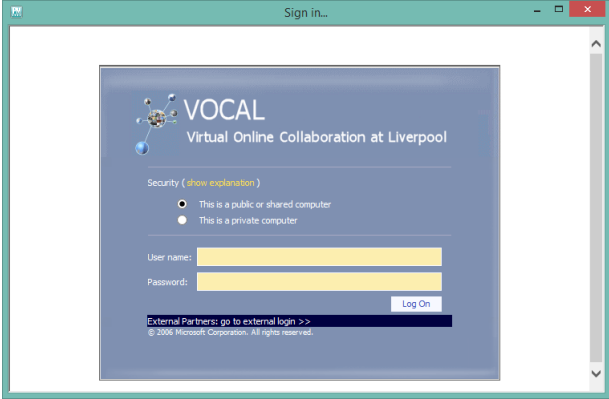
Step 2. Click on the Connect button.



Step 3. Choose an option from the sites listed by double-clicking on it.



Step 4. A pop-up browser will display your school/university page. You can log in as usual. Example (for UoL students and staff):



Notes:

- You should use the Default Server unless your teacher/tutor gives you other instructions.
- The browser connects directly to your school/university and your password will **not** be sent to *The Prime Machine* server.

Figure 7.2: Information about the “Sign In” process, provided in *The Prime Machine User Manual*, Version 2.0, January 2015.

Through these processes the client application is able to store a hashed string containing the username of the current user, and use this to obtain customised user settings and for the logs which are described in Section 7.1.4.

7.1.2 Stars

A common form of feedback, whether it is measuring satisfaction with a bank teller, a telephone service or the music tracks in media playing software is to provide the customer or user with a rating system based on stars. One reason for wanting users to provide this kind of feedback in *The Prime Machine* was to be able to store user rating data for evaluation purposes. However, given forced feedback might irritate or distract the user, and given over exposure to this kind of feedback may mean users click stars without much thought, a sensible way forward was to ensure that the starring feature would also provide a useful function for the user; if clicking on the stars would benefit the users of the system in some way, they would be more likely to do it. Therefore, the “star” rating and “pin” features which are described in this section have been designed to provide user bookmarks and ratings as well as the facility for the user to change the behaviour of the local cache of data, so that concordance results which have been ranked highly by a user will be kept locally on that user’s own cache file for longer. If the user enters a query in the concordancer and the concordance lines and other data are available in the local cache, the results are displayed much more quickly than if the data have to be retrieved from the remote server. The other way in which star ratings provide a useful feature from the user’s perspective is through the filter bar. Advice for teachers wanting to use concordance line results with intermediate level students sometimes includes the suggestion that filtering out some results could be helpful (Woolard, 2000) and selecting several examples can avoid overwhelming learners with too much data (Michael Lewis, 2000b). In *The Prime Machine*, a subset of the concordance lines and cards can be displayed and exported by marking the required lines with star ratings, and then setting the filter to screen out lines which do not meet a minimum number of stars. If logging is activated, these star ratings are also added to the logs so that information on a user’s satisfaction with the last action performed can be stored and analysed. Figure 7.3 and Figure 7.4 show the star rating elements as they appear on cards and lines. If users do not wish to make any use of this feature, they can hide the star rating elements, making the cards more compact and providing more space in the table for additional context to the left and right of the node.

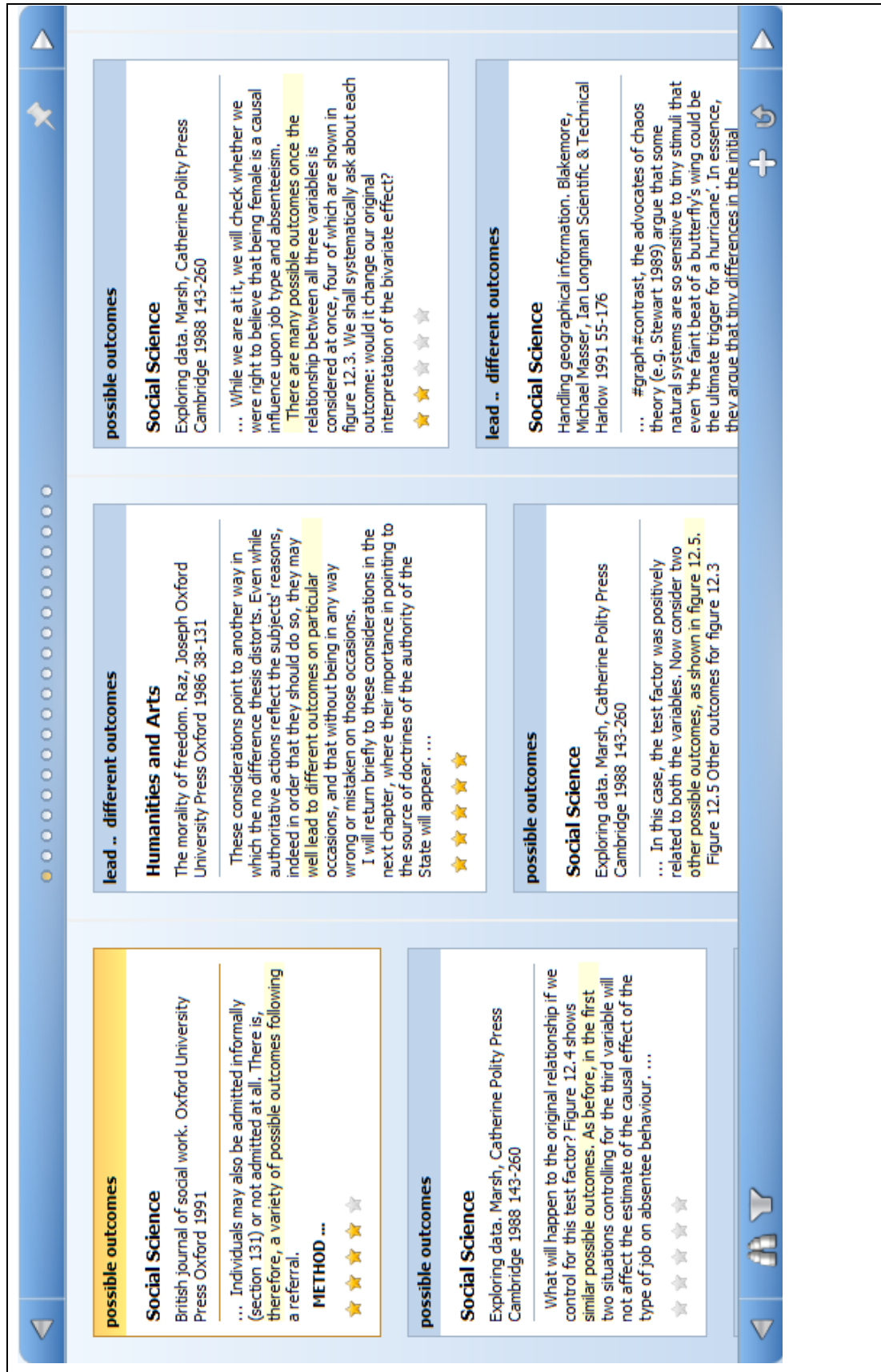


Figure 7.3: Concordance cards with the star rating elements visible; with incidental data for the node *outcomes* in the *BNC: Academic* sub-corpus.

	Text to the left of node	Node	Text to the right of node	Rating
1	itted at all. There is, therefore, a variety of possible	outcomes	following a referral.// METHOD ...	★ ★ ★ ★ ★
2	at they should do so, they may well lead to different	outcomes	on particular occasions, and that without being in a	★ ★ ★ ★ ★
3	b type and absenteeism./ There are many possible	outcomes	once the relationship between all three variables is	★ ★ ★ ★ ★
4	x this test factor? Figure 12.4 shows similar possible	outcomes	. As before, in the first two situations controlling for	★ ★ ★ ★ ★
5	both the variables. Now consider two other possible	outcomes	, as shown in figure 12.5./ Figure 12.5 Other outcc	★ ★ ★ ★ ★
6	ditions of many systems can lead to widely different	outcomes	since the systems exhibit stochastic behaviour withi	★ ★ ★ ★ ★
7	fferent scenarios include 'time slice' maps of possible	outcomes	and of the exchange of material and energy within	★ ★ ★ ★ ★
8	rocess, and what the effects of the various possible	outcomes	might be. It is to these issues that the rest of the br	★ ★ ★ ★ ★
9	who assesses the circumstances, evaluates possible	outcomes	and decides what to do; and while these properties	★ ★ ★ ★ ★
10	in these conflicting trial results suggest four possible	outcomes	. Firstly, cardiac death would be reduced but not de	★ ★ ★ ★ ★
11	1. fully ordered preferences (for any pair of possible	outcomes	, the agent prefers one to the other or ranks them e	★ ★ ★ ★ ★
12	rit individual switch settings. There are four possible	outcomes	: It gains red from the mother, and red from the fat	★ ★ ★ ★ ★
13	DDDD total: 64/ Out of these 64 possible	outcomes	, the 27 marked+ ' have just one fruit fly with both:	★ ★ ★ ★ ★
14	iment offers the theory. If an experiment's possible	outcomes	are O1, O2, O3, ... and a theory predicts that each c	★ ★ ★ ★ ★
15	plex, since profit maximisation can lead to different	outcomes	depending on the structure of the market in which t	★ ★ ★ ★ ★
16	tity of the sentence), there will be only two possible	outcomes	: If it is a property semantically compatible with the	★ ★ ★ ★ ★
17	rnative values, pupils can investigate other possible	outcomes	./ Wordprocessors, desk-top publishing software, e	★ ★ ★ ★ ★
18	which modify fiduciary duties, there are four possible	outcomes	. First, a court might hold that there was no authorit	★ ★ ★ ★ ★
19	government of the left, for example, produce policy	outcomes	which favour the working class and organized labou	★ ★ ★ ★ ★
20	dividual leader. One approach is to imagine different	outcomes	in the past and ask 'what if?' For example, what if M	★ ★ ★ ★ ★

possible outcomes

Social Science
 British journal of social work.
 Oxford University Press Oxford
 1991

... Individuals may also be admitted informally (section 131) or not admitted at all. There is, therefore, a variety of possible outcomes following a referral.

METHOD ...
 ★ ★ ★ ★ ★

Figure 7.4: Star rating elements on the right-hand side of each concordance line on the Lines Tab; with incidental data for the node *outcomes* in the *BNC: Academic* sub-corpus.

As well as being able to rate individual concordance lines with the star system, users can also mark an entire set of results using a pin. Again, this has the dual benefits of providing a way for the user to mark results in their search history for easy retrieval and longer term storage, and also being a further indication of how useful a set of data has been for a user. The pin icon is visible in the top right hand corner of Figure 7.3 and Figure 7.4. Figure 7.5 shows an example of the search history screen with some items marked with a pin.

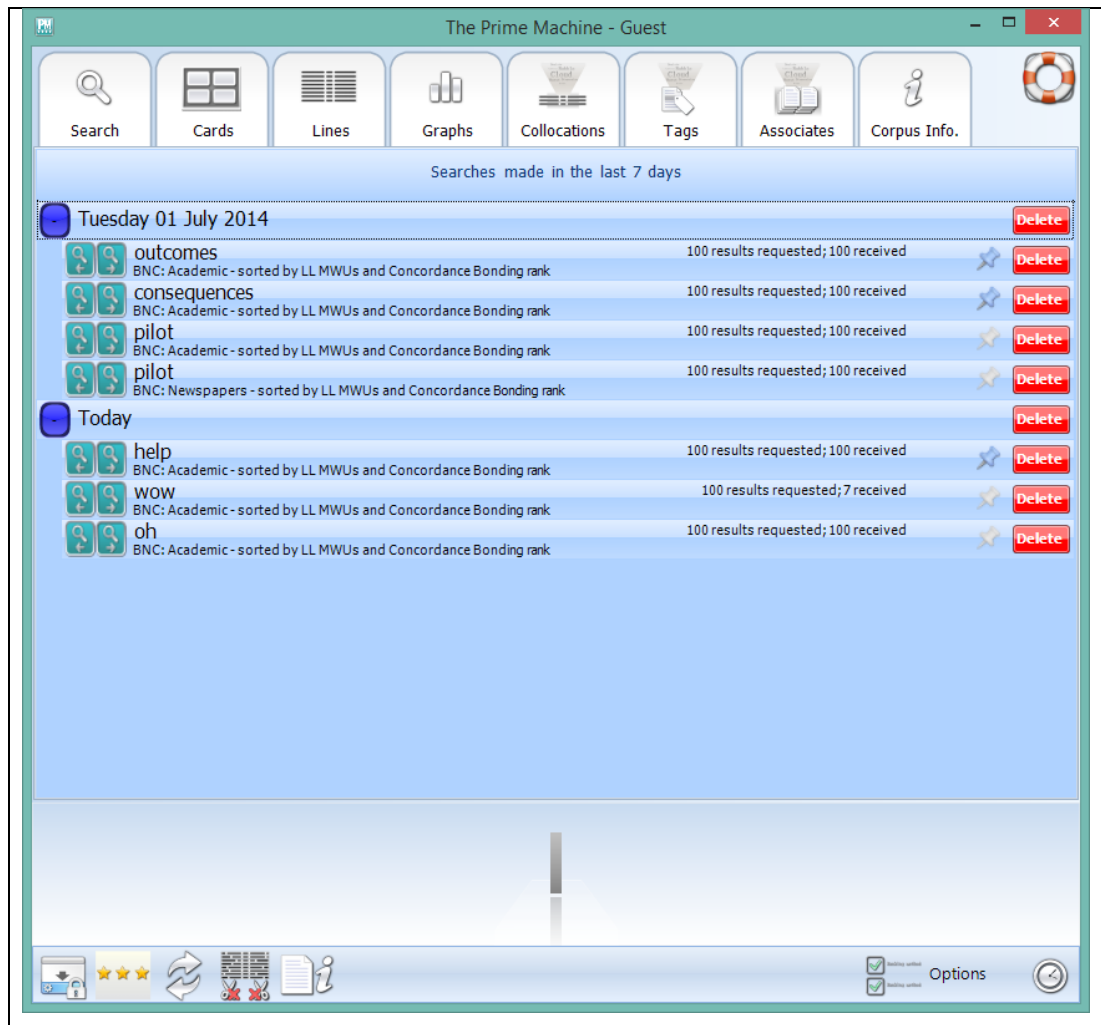


Figure 7.5: The search history screen with pins highlighted in blue for sets of results which the user wishes to bookmark and retain in the cache beyond the usual timeframe of 7 days.

7.1.3 Exporting results

Marking results as useful using stars or pins would indicate usefulness and intentions to return to review similar pages. However, teachers and students are also likely to want to use results in other documents such as classroom materials, notes, feedback or reports. From the beginnings of Data Driven Learning, handouts of printed concordance lines have been used as teaching materials (Johns, 1994). As Charles (2007) points out, printed results

from concordancers mean that students are able to review data after class even when the corpora from which they were taken are not available. Boulton (2010) has shown that printed concordance lines can provide effective input for DDL activities, meaning that learners who are not keen on making direct use of software can also benefit. For the evaluation which is explained later in this chapter, one way to help the participants consider the potential of the software was by providing individual feedback on their essays using concordance data. For all these reasons, some functionality for exporting data from the software was needed.

Printer support for some of the visual components in the programming libraries used for *The Prime Machine* was not very extensive, and some careful consideration went into how best to implement features which would allow teachers and students to extract clear and useful output, without requiring complicated “page setup” or “print preview” screens, or the use of additional third party components. Since many language classrooms are equipped with data projectors and such office productivity software as *Microsoft Office* is frequently used to create slides or word-processed hand-outs, the development of the facility to generate images and text data which can be easily imported into other applications was a priority. In *The Prime Machine*, when the user double-clicks on a results page, a menu appears which provides options to copy or save the results in several different formats. Copying or saving as “Picture” files allows the user to import the results as an image and is the easiest way to incorporate concordance cards, lines, tables or clouds if the size does not need to be changed dramatically. However, the MetaObject format provides an alternative way to copy the data which allows the target application to use its own drawing processes for text and vector graphics, meaning if the size of the font is increased after the image has been copied, the letters and symbols used in the text are drawn smoothly. Cards can also be saved in multiple files using the “Save all...” options. For table data, a further option is to copy or save the results so they can be imported into a spreadsheet. Figure 7.6 shows the export options for the Cards Tab and Figure 7.7 shows the export options for table-based data.

More about the Cards Tab

The Cards are useful if you want to see a wider context, and they also show you the word in its **position in a paragraph** or **heading**.

If you want to copy or save a card, you can double-click on it.

Cards can be copied or saved in two different formats: MetaObjects or Pictures.

MetaObjects allow you to re-size the card in another computer program without losing picture quality.

Pictures are saved as JPEG images of the text, so cannot be edited and may not appear very smoothly if enlarged.

If you want to save all the cards, you can use the **Save all** options. This will create a set of MetaObject or picture files.

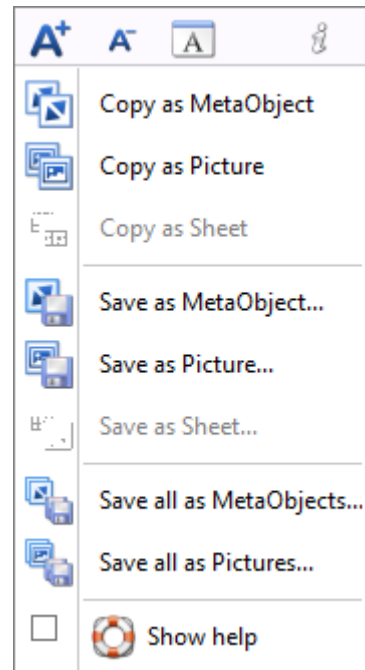


Figure 7.6: Export options available for cards.

As with all the other results pages, double-clicking brings up a choice of ways in which you can copy or save the results.

For the Lines tab, you have the option to copy or save as a spread sheet.

The default file format for saving as a sheet is as a Microsoft Excel 97-2003 Workbook (.XLS file). However, you can also choose to save as a CSV (comma delimited file).

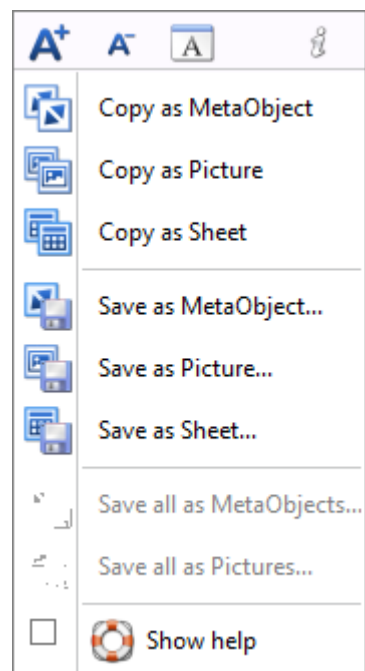


Figure 7.7: Export options available for tables.

7.1.4 Logs

One of the reasons for developing a client-based system using a language like *Delphi* was to maintain much more control over the user-interface and to facilitate the collection of detailed information about how the application was used. For research into the use of corpus tools with language learners, Pérez-Paredes, Sanchez-Tornel et al. (2011) argue that tracking of user actions through logs is essential in order to determine actual use rather than reported use. On a website, it would be possible to create logs for clicked actions, but in order to know how many concordance cards had been viewed, one would either need to run some of the scripts on the browser through code like *Java* or to request each view one by one. In *Delphi*, it is simply a matter of adding actions to events triggered by mouse or keyboard movements.

Table 7.1 below shows a summary of the kinds of actions which are logged. The procedure in the *Delphi* code which adds actions to the log, first checks to see whether the `log_mode` for the current user has been marked for record keeping. During formal evaluations where participants have consented to the collection of this kind of data, the `log_mode` can be made active and logs are sent when the application is not busy retrieving data from the server or when the application closes.

Table 7.1: User actions which can be automatically logged by the software

Action Category	Examples	Details	Further notes
Search Support	<ul style="list-style-type: none"> • Auto-complete for single words; • Auto-complete for collocations; • Suggestions for words with similar meanings; • Spelling support request; • Request for a word or collocation to be checked in other corpora; • Alternative corpus selected after other corpora have been checked for a word or collocation not found in the current corpus. • Use of other navigation buttons (“Back”, “Forward”, “Home” or “Swap”). 	Words / collocation clicked	For several of these logs, the item number from the list of choices is also recorded
Query Blocked	<ul style="list-style-type: none"> • Rules for query syntax not followed; • Too few or too many words entered in a single query; • Word or collocation not found in the currently selected corpus; • Combination of words not found in the currently selected corpus. 	Search string	
Query	<ul style="list-style-type: none"> • Single search; • Compare mode search for two different queries; • Compare mode search for two different corpora; • Requests for more lines or collocation data. 	Search string	Other details for some of these are logged including the number of results requested and the ranking method used.
Tab	<ul style="list-style-type: none"> • Cards Tab; • Lines Tab; • Collocation Tab; • Graphs Tab; • Tags Tab; • Associates Tab. 	Number of seconds viewed	Other details such as the number of cards viewed and other settings are stored
Other	A variety of other actions including the use of filters, access to help screens, changes to options, changes of the main corpus and use of various visual elements including the “Priming Dock”.		

As can be seen, a range of categories have been created, allowing the grouping of log data in terms of search support features, actual queries, viewing of results and other features such as changes to options and access to help.

The remainder of this chapter introduces the method and results for the evaluation of *The Prime Machine*. The features outlined above helped facilitate several aspects of this evaluation and could also be used more extensively in future evaluations.

7.2 Method

7.2.1 Participants

Volunteers from across the university where the author and creator of *The Prime Machine* worked were invited to participate in the project through short announcements before lectures and through the student email system for the Department of English, Culture and Communication. Students who were currently studying a module with the researcher were not permitted to participate. Announcements calling for participation were made by the researcher before lectures in a range of disciplines across first, second, third and fourth year programmes. Three sessions were scheduled for the same day, and these face-to-face sessions took place on a Saturday to avoid any conflict with class teaching. Students were able to indicate a preferred slot through the university's virtual learning environment system, *ICE (Moodle version 1.9)*, and an information sheet was also provided for them to review before the first session. In order to sign up, students could scan a second generation QR code, or type the full URL address for the special course page on the *ICE* system. However, since the research was open to students from across the university, they were also informed that provided sufficient seats remained they would be welcome at any one of the sessions and would be able to register on the day.

7.2.2 Materials

The materials for the evaluation included two questionnaires, a set of instructions demonstrating various aspects of the software, a brief user manual for the software and a set of essay question prompts. There were also consent forms and information sheets complying with the ethical guidelines of both XJTLU (where the research took place) and the University of Liverpool.

The first questionnaire included demographic questions as well as questions relating to the students' own views on their use of a range of language learning reference tools such as dictionaries, electronic dictionaries and search engines, etc. Several of the questions were based on those from the survey of students which has already been reported in Chapter 2, with a view to providing several comparable points. Following the prominence of search engines as a preferred tool for checking words in that earlier study, this was added as one of the choices in the relevant MCQ questions. Furthermore, given the growth in popularity of mobile phone apps over the last 2 or 3 years mobile apps were also added as a specific choice. In this way, with a broad range of relevant study resources available as choices in the early part of the first questionnaire, for the questions relating to students habits and their attitudes regarding the best resource for several specific language learning issues, the option of concordance lines was not in any way foregrounded. The first questionnaire also included questions about peer review and more general attitudes towards language study.

The second questionnaire explicitly picked up on one of the questions from the first questionnaire and asked students whether their view of the importance of examples had changed as a result of taking part in the project. There were also questions about how much they used several of the main features of the software and how useful they perceived them to be. There were also a range of questions designed to gather their views on appropriate future uses of the software and any suggestions for improvements.

Both of these questionnaires were delivered electronically through the university's virtual learning environment system, ICE (Moodle version 1.9). Examples of resources were provided on a printed A3 sheet, so that students would not need to flip between screens.

The instructions provided step by step guidance on the overall procedure from answering the first questionnaire, downloading the software, working through the examples, writing the essay, and performing the follow up tasks later. In order to make the writing task relevant to students from a wide range of university programmes, prompts were written on a range of topics related to contentious but non-threatening issues which had been discussed in the news, following the style of popular language proficiency examinations. Originally, a peer review element had been planned as one of the follow up tasks, but this element was later removed.

7.2.3 Procedure

Participants volunteering to take part in the project were required to attend a face-to-face session in one of the university computer labs. At the beginning of each session,

information sheets and consent forms were distributed and then students were invited to complete the first questionnaire on the *ICE* system. After completing the questionnaire, the students were free to start working through the instruction sheet, download the software and look through the user manual. When the questionnaires had been completed, the researcher worked through all the examples using a computer attached to a data projector. The participants were free to just watch or to try using the software themselves. At the end of the presentation, blank lined sheets were distributed to students who preferred writing essays by hand, while others loaded Microsoft Word and started to work on their essays on the computers. The students were then given one hour to write their essays. During this time, they were free to consult any other resources and to make use of the software. Formal examination conditions were not enforced.

Once students had submitted their essay to the researcher, they were free to leave. Within the next two days, individual feedback on each essay was sent to each participant. The template used by the researcher for this feedback included some comments based on each of the four criterion from the public band descriptors for IELTS (www.ielts.org). The feedback also included three screen shots showing sets of concordance lines related to three words or phrases used in the essay, as well as two *Microsoft Excel* spreadsheet attachments showing up to 100 more of the lines for these. A table of other single items or pairs of items to compare was also given. This feedback was then sent to each participant and he or she was invited to complete the second questionnaire online once he or she had reviewed the feedback, making use of the software again if he or she wished.

As was explained in the initial invitation and in the instructions, students who wrote an essay during the face-to-face session and completed both questionnaires were entered for a prize draw to win 200 RMB of goods from an online retailer, Amazon.cn. The departmental secretary helped to ensure three winners were selected randomly from this group.

Four students participated in a pilot study several days before the main sessions took place. This gave a valuable opportunity to try out the facilities in the computer lab, to check the timing and pace for the demonstration and to identify any problems with the wording of questions or the software itself. Three specific issues came out of this pilot. Firstly, in terms of the questionnaire design it transpired that the students were not so familiar with the meaning of “extensive” or “intensive” reading. Therefore, an explanation was added in brackets for these items with “reading a long chapter quickly for the main ideas” for the

former⁵⁹, and “reading a short passage in detail” for the latter. Secondly, it became apparent that a peer review task as part of the computer lab would not really be practical or received with great enthusiasm. While peer review has been used at the institution in the past and the results of the questionnaire looking at this topic show that many students are regularly asked to do peer review tasks, having a peer review for an assignment which is merely part of a research project with partners who are co-participants in research rather than peers in a regular class group perhaps made it seem too artificial. Based on this consideration and given that the first questionnaire and software demonstration took over an hour, the peer review activity was removed from the session plans. This also meant that a question about the usefulness of the software for the peer review activity had to be removed from the second questionnaire, but in the question about possible future use, peer review was left as an option. The third aspect was to do with the design of the window which appears on the screen when the application is first opened and when it is closed. This window displays a message while the software contacts the server to synchronize user settings, to collect information about the default corpus and to transmit logs. With network connections, it seemed prudent to include a “panic button” which would allow the user to halt all processes in the event that data could not be sent or received. However, at the end of the pilot session as students were closing down the software to leave, I noticed that one of the four students clicked on this button almost as soon as the closing window appeared. The student obviously thought that the panic button was a second confirmation to close. This unfortunately had the consequence that none of the logs for that student were transferred to the server. In order to prevent this from happening in the main study, and also to improve the design of this window so it had less of a resemblance to a non-essential confirmation dialog box, the application was recompiled and the button was placed inside a drop-down menu labelled “Check connection” as can be seen in Figure 7.8 below. A help message explains the reason for the delay and how to access the panic button through this drop-down menu if the application needs to be halted.

⁵⁹ This explanation was designed to capture the sense in which it was believed this term was typically used in China, but in future it might be interesting to explore student attitudes to the use of reference tools when reading graded readers or reading novels as opposed to completing close reading activities.

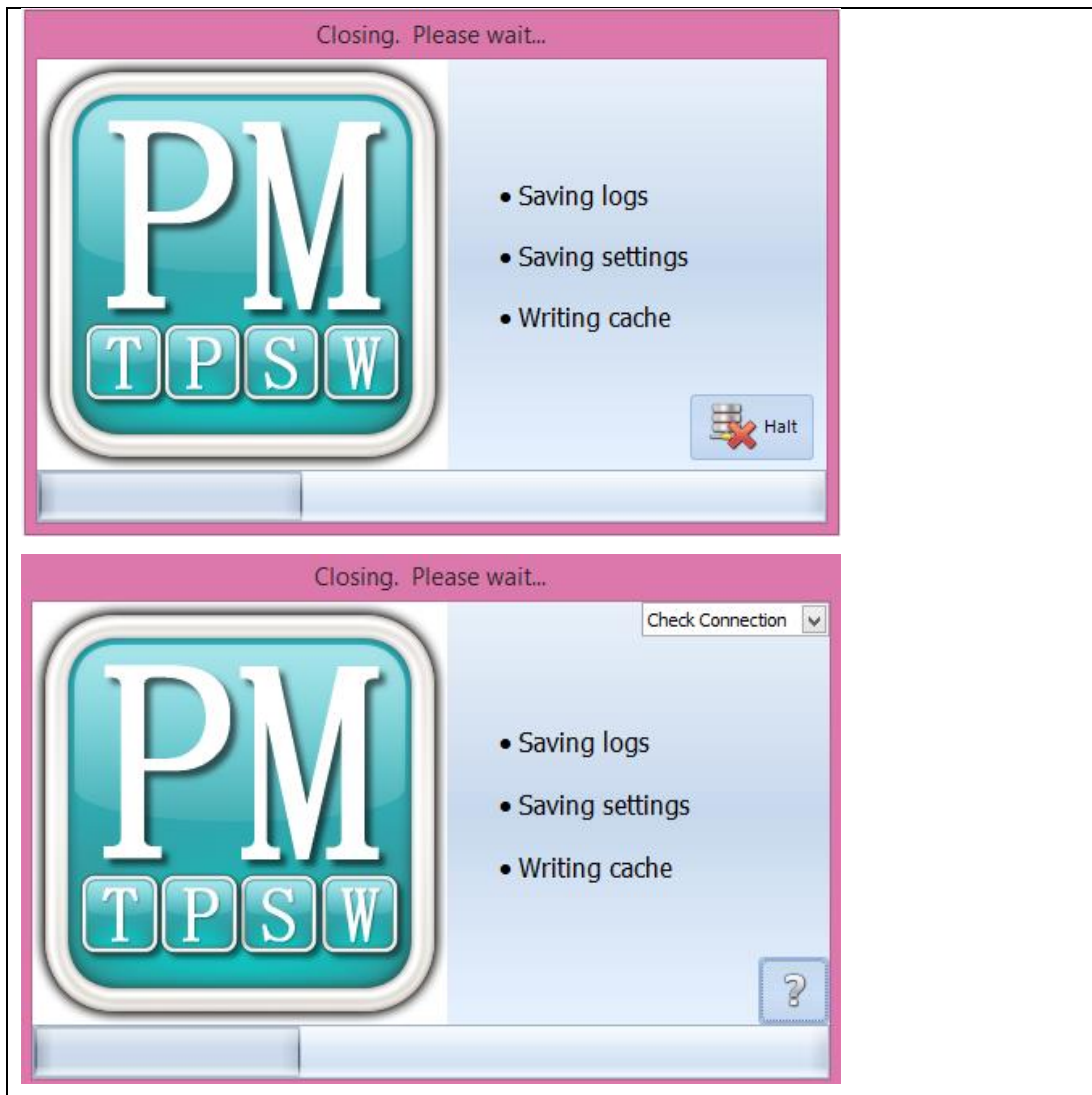


Figure 7.8: The design of the closing screen before the pilot (top) and after the pilot (bottom).

It should be noted that although the long-term aim would be to locate the server inside the university network, during this evaluation the server remained off campus. Approval processes for an experimental server application which is understood by the university to be related to the professional development of the researcher are somewhat complicated. Positioning the server off campus had two important consequences. Firstly, since the home internet service provider (ISP) was different from that used by the university, the time taken for a message to be sent from the university to the server and back again was considerably longer than would have been the case if the ISP could have been matched, and much longer than it would have been if the server had been positioned inside the university firewall. This meant that sometimes there was a delay of a few seconds before auto-complete data appeared on screen. The second consequence was that a dynamic domain management service was used to automatically update the IP address of the server

since home ISPs in China do not have fixed IP addresses. The change of IP address is unpredictable, but seemed to happen frequently at around 1pm or 2pm, and this meant that students trying to use the software for the five minute window around this change-over period would have received a message that the server was not available and had to wait a few moments before gaining access. The failure of the commercial domain name resolution service seemed to be responsible for short period of approximately 2 hours during the few days when students had been asked to review the software and complete the second questionnaire and one student reported problems accessing it during this time. Fortunately, it was possible to send instructions on how to access the server using the IP address, and normal DNS services resumed by late afternoon.

A further problem with the software itself is also worth noting. The final testing of the code for filtering and comparing concordance lines according to priming was only made during the same week as the evaluation took place. As well as being a feature which may have been too advanced to introduce in a single session, it did not seem prudent to risk introducing potential crashes to the session given that some of the features were not finalized until after the pilot. More importantly, a few small bugs were also found during the week of the evaluation and one important change led to a further problem with the software which appeared during two of the sessions. Following the face-to-face sessions, the code was checked and it transpired that there were problems updating the grid with new results when a row in the KWIC view was currently highlighted. This problem was fixed before the feedback and further instructions were sent out and participants were told to download the updated version if they wished to use the software again.

7.2.4 Summary

The first session took approximately 2 hours including:

- 30 minutes for a briefing, signing of consent forms, completion of questionnaires;
- 30 minutes to introduce the basic operations of the software;
- 1 hour to write an essay (roughly 300-400 words);

Within 2 days, feedback was sent containing:

- General comments on Task Response, Coherence and Cohesion, Lexical Resource and Grammatical range and accuracy;
- Three sets of screen-shots containing approximately 12 concordance lines for three specific language issues in the essay;
- A table of other suggestions for words or phrases to look up in the software, some of which contained comparisons.

Over the next few days, students were free to review the feedback, use the software again independently if they wished, and finally to complete the second questionnaire.

During the whole of this time, logs were collected automatically.

7.3 Results and Analysis

A total of 25 students attended one of the face-to-face sessions, completing the questionnaire, and submitting an essay. The vast majority of the participants were female, with just 3 male participants. In terms of the academic programmes from which the students came, the most common was Financial Mathematics with 14 students, and this was followed by English and Finance (5 students), and 3 from engineering or computer science programmes, 2 from Chemistry and 1 from Economics. Only one of the male students came from Financial Mathematics, but the gender balance in the enrolment of XJTLU students on Financial Mathematics and English and Finance is predominantly of female students. The ages of the participants ranged from 18 to 22, with 3 students from Year 1, 7 students from Year 2 and 15 students from Year 3. There were no Final Year students or Masters students participating in the project. One reason why more Year 3 students may have volunteered is because formal EAP classes run for the first two years and students in Year 1 and 2 will have been studying credit-bearing EAP modules. Meanwhile, third year students at XJTLU will have opted to complete all four years of their

degree programme in China, and it is understood that the majority of these students plan to go overseas for further study. Without regular EAP classes but with plans for overseas study starting to firm up, students in Year 3 would probably have been more open and interested in availing themselves of an opportunity to practise writing an essay in a style similar to IELTS and would have probably greatly valued the opportunity for additional feedback.

All 25 participants were Chinese and came from Mainland China, with 8 from Jiangsu province (including 3 from Suzhou itself), and the remaining 17 students from 10 other provinces including Anhui, Shanxi (陕西省), Shanxi (山西省)⁶⁰, Heilongjiang, Hebei, Hubei, Shandong, Fujian and Hunan. Again, this reflected the range of provinces from which the whole student population is drawn, with a large portion of the quota for undergraduate places coming from Jiangsu province, as well as some of the university's strongest areas for recruitment including ShanXi (陕西省, where the parent university, Xi'an Jiaotong University, is located) and several Eastern provinces (Shandong, Fujian and Anhui). Students reported that they had studied English for between 7 and 15 years, with 19 out of 25 students having studied English for 10 years or more.

Following the demographic questions, the first set of questions in the questionnaire was related to the students' reported use of reference tools to help them with their English. As well as the names of the different kinds of resources, the A3 sheet provided examples of dictionary entries, popular search engines or mobile phone apps and a picture of concordance lines.

As can be seen from Figure 7.9, by far the most popular choice was mobile phone dictionary apps, with 21 students claiming to use these very often, and 3 students selecting 4 out of 5 for this item. Just one student reported a lower score (2/5) tending towards never. Interestingly, this student was the same student who indicated very often for concordance lines and one of the four students who indicated 5/5 for English-English dictionary with Chinese translations. Following mobile phone or electronic dictionaries, the next most popular choice was search engines, confirming one of the conclusions from the survey of students reported in Chapter 2. Interestingly, in the earlier study, mobile phone or electronic dictionaries were not provided as a separate choice, and Chinese-English

⁶⁰ These two provinces are adjacent to each other and have very similar sounding names. The Chinese characters are provided here as a means of identifying the two provinces.

dictionaries in that study had a tendency towards more frequent use. However, in this study, it is clear that paper dictionaries are disfavoured, and electronic means through mobile phone apps or search engines are clearly favoured. As expected, the other clear finding was that for the majority of students concordance lines are not at all regularly used, with 72% of respondents claiming never to use them at all, and a further 20% choosing the second lowest rating. Three of the 5 students who chose 2/5 for concordance lines did not rate any of the resources below 2. The student who rated concordance lines 3/5, also selected neutral scores for half of the resources and did not select 1 or 5 for anything.

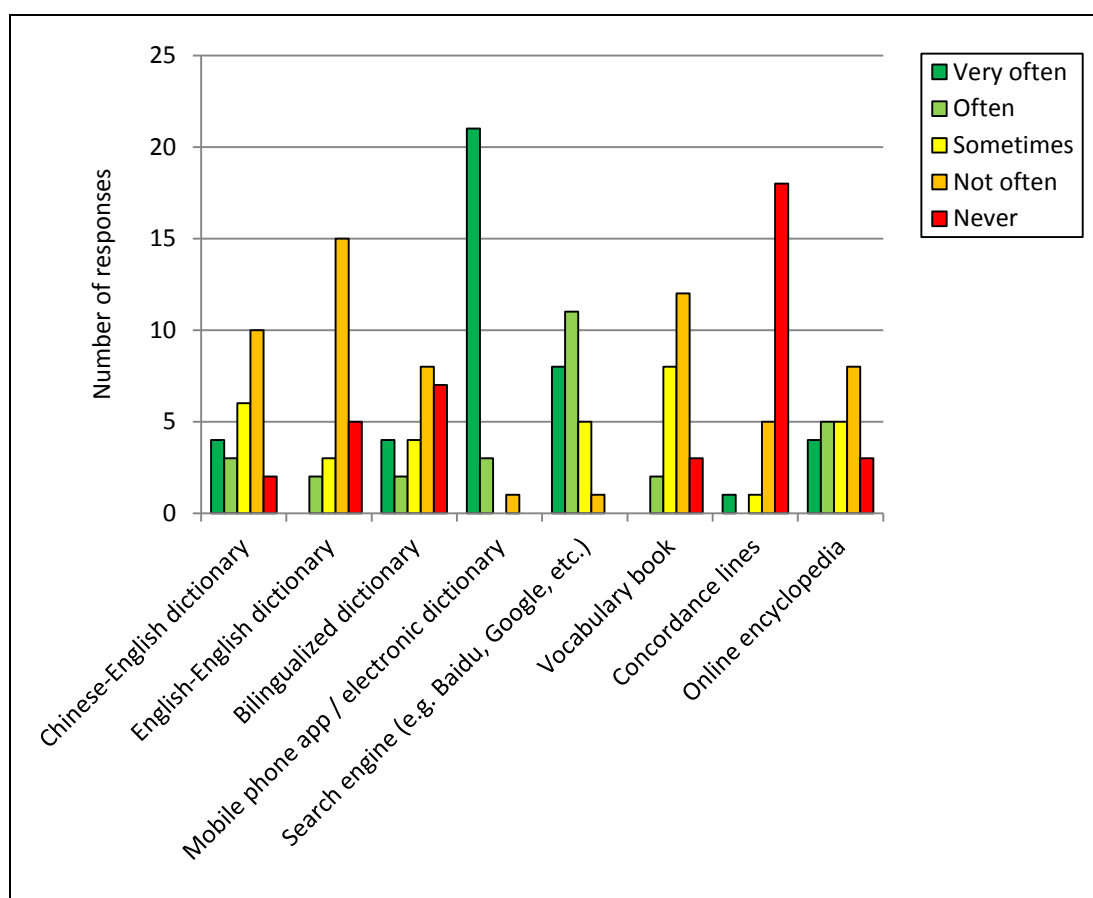


Figure 7.9: Reported use of different resources.

The next set of questions was related to which resource listed on the handout students thought would be the most useful for five specific kinds of language problems. Figure 7.10 below shows the number of students who selected each of these.

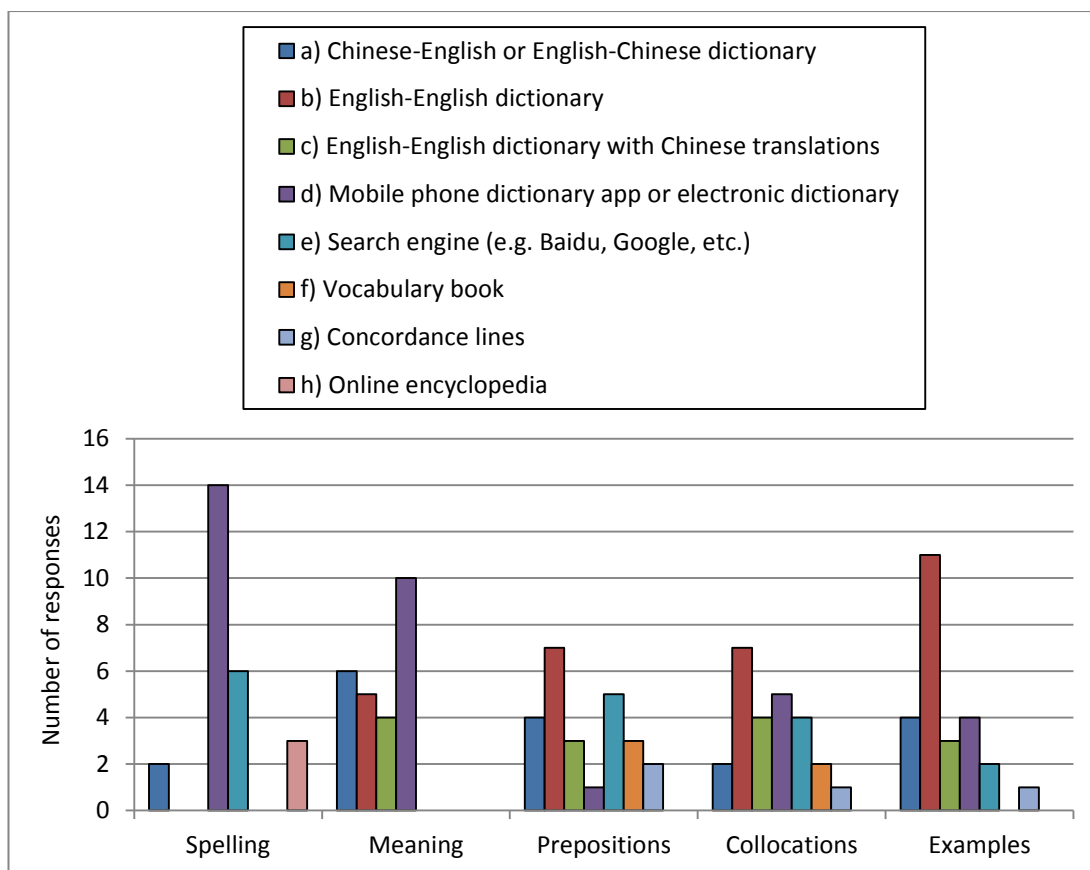


Figure 7.10: Judgements given by participants on the best resource for a variety of language issues.

It is clear that mobile phone or electronic dictionaries were perceived to be the best choice for spelling and meaning, while English-English dictionaries were considered best to check prepositions, collocations or to find examples. Interestingly, search engines were not considered the best choice by any students when checking the meaning of words and were less popular than all three paper dictionary types and mobile phone dictionaries as a source for examples. The only three areas where search engines were considered the first choice by 16% or more of the students were for spelling (24%), prepositions (20%) and collocations (16%). This would suggest that search engines are used for language purposes by the students to check spelling and co-text rather than to provide information about meaning or examples. This could also mean that the strong tendency for reporting search engines as a choice which was reported in Chapter 2 may have been heavily influenced by the task, given that in that previous task the focus was on finding a correct answer for a decontextualized sentence rather than to produce language with a communicative purpose.

It is also worth noting that one of the questions in the first questionnaire asked students to report whether or not they used a spell checker on their computer, and only 76% responded “yes”. A similar question was included in the survey of students reported in

chapter 2, and at that time over 97% of students from the same institution said they did, while the figures for the other institutions were much lower at between 40% and 47%. The 4 out of the 6 students who said they did not use a spell-checker selected it not being installed as part of Chinese software as one of the reasons.

Again, it is evident that concordance lines were not considered the best resource for any of these problems by the vast majority of students. There was also an interesting mismatch between the answers to the previous question about reported frequency of use and the resources which were considered most useful. Only three students chose concordance lines for any of the problems, and all three of these students had reported actual use of concordance lines as being 1 (never) or 2. The student who had rated concordance line usage so highly in the earlier question chose the option for “Chinese-English or English-Chinese dictionary” and the option for “Mobile phone or electronic dictionary” for all of the problems. This suggests that the student who had reported using concordance lines very frequently was perhaps using them for other work or considered them to be a supplementary resource rather than a key one.

Another obvious conclusion which can be drawn from these data is that the vast majority of students (16 out of 25) consider translation dictionaries or mobile phone and electronic dictionaries to be suitable resources to check meanings. The wording of this question was “Checking a word which has several different meanings” and it is surprising that students place confidence in dictionaries which often only have a limited range of translations.

When asked to rate how often they looked up words and phrases in a dictionary during different kinds of task, the highest ranking activity was “Writing”, followed by “Checking or editing your own work”. “Intensive reading” came close behind, with fewer students rating “Extensive reading”.

The questionnaire also included a question about attitudes to different aspects of what is important for good writing in English. As can be seen in Figure 7.11 below, the strongest agreement was shown for the statement “Good writing in English means getting your reader to understand what you are trying to communicate”. Being able to express something in a “natural way” was rated at almost the same level as “paying attention to the rules of grammar”. The lowest rated area was “paying attention to [the] specific meaning of words and phrases”. Almost half of the students rated rules of grammar higher than specific meaning, and only 3 students rated specific meaning higher than rules

of grammar. Although the overall figures for natural expression and rules of grammar were very similar, 14/25 students rated these two aspects differently, with equal numbers higher and lower.

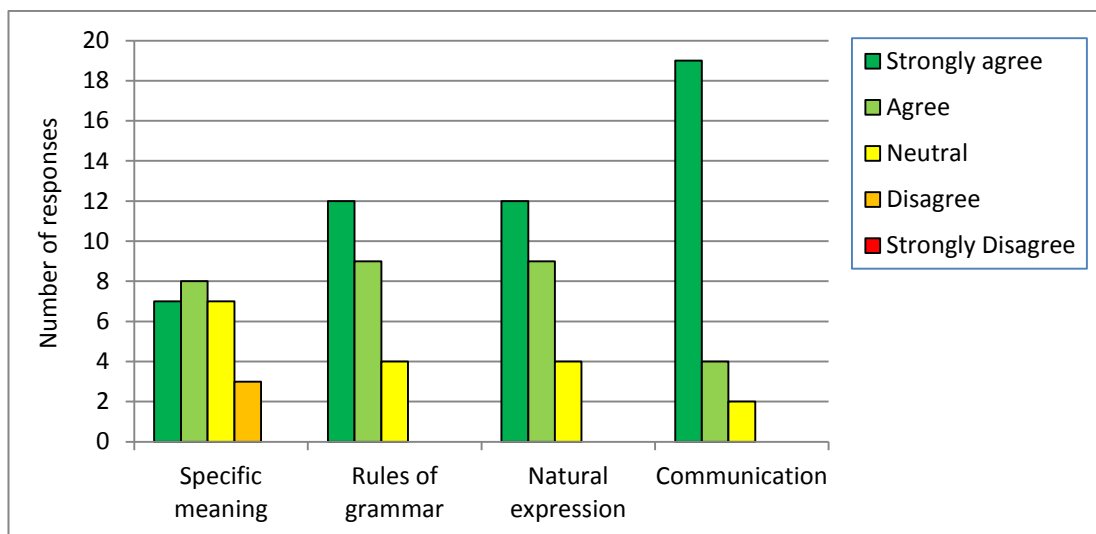


Figure 7.11: Attitudes to the importance of different aspects of language for good writing in English

As explained earlier, after submitting the essay, students left the first session and were sent individual feedback within the next two days. They were then invited to complete the second questionnaire. Although 25 students took part in the face-to-face session, two students did not complete the second questionnaire.

In terms of reported use during different stages of the session, the results were fairly evenly spread. Figure 7.12 shows that the “Writing”, “Checking/Editing” and “Reviewing feedback” stages were all rated as “Often” or “Very often” by at least 13 students. The “Planning” stage, however received fewer positive responses, with only 6 students selecting “Often” or “Very often” and this was the only stage where any students reported never making use of the software.

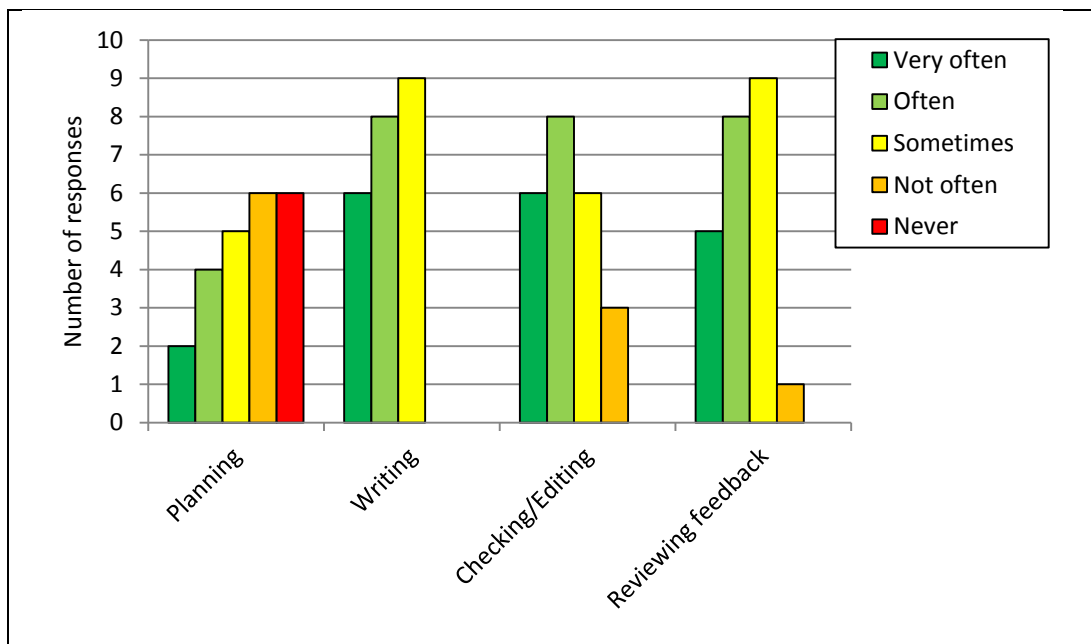


Figure 7.12: Reported frequency of use during different stages of the writing task.

Average ratings were 2.57 for planning, 3.87 for writing, 3.74 for checking or editing before submission and 3.74 for reviewing feedback from the teacher. The similar average scores mask individual differences, however, as different students reported use of the software at different levels for Writing, Checking/Editing and Reviewing. Only three students rated these three areas equally.

However, it is hard to find evidence of actual use of the software in the logs, which suggests that students were either exaggerating their use of the software or reporting attitudes rather than actual use. The strength of the results is somewhat weakened if the question is interpreted as being representative of attitudes, but the varied results do suggest that different students feel that the software would be useful for different stages of the writing process.

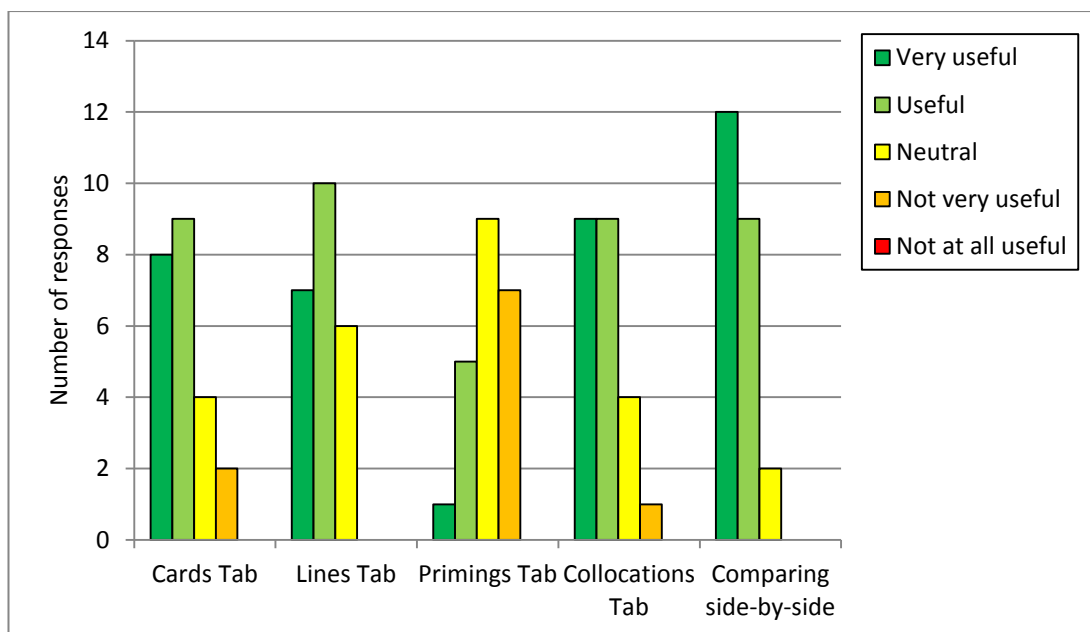


Figure 7.13: Evaluation of the usefulness of some of the main features of the software

From the graph in Figure 7.13, it is clear that students rated both the cards and lines tabs quite positively, with approximately 74% of those who answered the second questionnaire choosing Useful or Very Useful. It is worth noting that although the Cards Tab seems more mixed with 2 students reporting it was not very useful, 6 of the 23 students (26%) rated the Cards Tab above the Lines Tab. Having both ways of viewing the data may cater for different learner preferences and different uses.

The Graphs Tab⁶¹ received the least positive feedback, with a much lower average rating, however it is worth noting that 6 out of the 23 students (26%) rated it as very useful or useful. The student who rated the Graphs Tab as “Very useful” had lower ratings for all the other features except the Cards Tab.

The Collocations Tab was generally very positive. The average rating for students where logs showed tables had also been viewed as well as clouds were slightly higher (4.375 vs 4), but since the participant numbers were so low this should be treated with caution. What can be said with some confidence is that students seemed to think the collocations tab was among the most useful aspects. The student who rated the Collocations Tab at 2 also rated the Cards Tab and Graphs Tab as 2, but rated the Lines Tab as 4 (useful). Clearly, this

⁶¹ At the time of the evaluation, the label on this tab was "Primings Tab", and the questionnaire asked respondents to comment on it using this name. However, the label was subsequently changed to "Graphs Tab" as this better matches the purpose and scope of the tab.

student preferred looking at the information in the KWIC view, but from the logs it seems that he or she did not view the tables for collocations. Students with a preference for KWIC view may also prefer tabulated data and perhaps also value the position information from Mutual Information data, but from the logs it is evident that none of the students explored collocations other than the default measure.

By far the most striking result from Figure 7.13, however, is that being able to compare results side-by-side was rated very highly indeed.

The results of the questionnaire questions related to the frequency of use during different stages of the task and the students' evaluation of the usefulness of some of the main features provide evidence that the first part of the research question has been positively answered: the learners reported that they could find examples which they considered to be helpful.

Table 7.2: Logs showing the number of views and time spent on different tabs in the software

	Number views	Total time	Average number of seconds
Cards Tab	160	6485	40.5
Lines Tab	113	9328	82.5
Graphs Tab	53	2479	46.8
Collocations Tab	70	4325	61.8
Tags Tab	35	813	23.2
Associates Tab	48	6615	137.8

Table 7.2 shows that the logs seem to support the views regarding the usefulness of different tabs, with Cards and Lines having much higher event counts and generally more time being spent on Cards, Lines and Collocations. When looking at these figures, however, it is worth noting that the Cards Tab was set as the default results tab for all users, so this will have received a log for every search which was completed. However, looking at the number of cards viewed for each event, the logs show that an average of 15.1 cards were viewed with a range between 1 and 65. Only 17 out of the 160 events had fewer than 10 cards marked as having been viewed. Since only a few cards are visible unless the user scrolls down, this seems to confirm that some users viewed quite a few results on the Cards Tab.

It is worth bearing in mind, however, that the vast majority of the events were from the sessions on Saturday, and the time in Table 7.2 above should be treated with caution since it is likely that students may have left a tab visible when stopping to listen to another part of the demonstration. The times are calculated for the whole time that the application is “active” (in the sense of being the window with the current focus), so this kind of data is more reliable when students are completing a task in another window rather than switching attention to a data projector during a demonstration or working on a paper-based activity.

From the logs, only 4 students seem to have made use of the software after Saturday, and figures for use across different tabs for later use are shown in Table 7.3 below.

Table 7.3: Logs showing the number of view, time spent and the number of different users for the results tabs after the main input session.

	Number views	Total time	Users
Cards Tab	10	186	4
Lines Tab	9	1679	4
Graphs Tab	4	91	3
Collocations Tab	7	74	4
Tags Tab	4	22	2
Associates Tab	3	26	2

Again, it is clear that most time was spent on the Lines Tab.

The logging of changes to visual elements both from the options menu or the bar at the bottom of the screen showed very few changes being made. The show/hide ratings and citations buttons were each switched off and on again 6 times. However, the button to reduce the Lines Tab display to complete sentences was used much more frequently. This was demonstrated briefly in the session and although a total of 96 show-or-hide events for this feature were logged from 14 different students, more than half of these were events where the setting was immediately changed again afterwards. Nevertheless, it would seem that this feature was something which several students found useful⁶².

⁶² This feature was originally requested by a fourth year student who had been using the software for research purposes.

Although, figures for the Graphs Tab may seem a little disappointing, it is worth noting that there were a total of 188 clicks on the priming icons on the dock and 18 users made use of this feature to switch to the Graphs Tab.

In terms of use of the ability to compare results side by side, the logs show that a fair proportion of searches were made like this. Of the 281 logs from 22 users, 56% of searches were for one term only, while 44% were made in compare mode. 3 users did not appear to make any queries. Using the logs for the right-hand retrieval only, 85% of the compare mode searches were to compare different queries across the same corpus, while 15% were comparing the same query across two different corpora.

The summary of the log data which has been provided here addresses the second part of the research question, which was concerned the kinds of information viewed and the number of results. It is clear that overall the students spent most time on the Lines Tab, followed by the Cards and Collocations tabs. The logs also showed some engagement of the students with the different kinds of information and the number of results, measured by the number and range of events logged and the number of concordance cards viewed.

As well as being able to compare results easily, another set of important design features which were explained in Chapter 3 were related to search support. The third part of the research question was to ascertain which of these search support features would be used most frequently. A total of 54 queries from 16 users were logged as having been blocked by the software. Six of these were related to spelling errors, 1 was because a Chinese word had been entered. Nine blocked queries contained collocations where the incorrect format had been given (lack of spaces or additional full stops, etc.), and 20 blocked queries were because the phrase was not stored as a collocation in the system. Four queries were blocked because it seems nothing had been entered in the search box. A further 14 queries were blocked but information is not provided in the logs.

As well as preventing users from making queries and waiting only to discover that no results are found, the software also included other features such as auto-complete, collocation suggestions, synonyms, other word forms and spelling support. From the logs, auto-complete for words was used 12 times, and 9 of these were for words or word forms which did not form part of the demonstration. Collocations were selected from the drop-down box 9 times, 8 of which were for collocations not part of the demonstration. Spelling support was requested 5 times, but from the logs it does not seem to be the case that the

student made a subsequent search using the correct spelling. This suggests that either the spelling component was too slow or did not provide useful suggestions, or perhaps that students were trying it out rather than actually wanting to use it to assist with their spelling.

The provision of several forms of help is important for any computer application. However, from the logs it seems that very few students felt the need to click on the Life Ring to gain further information. This could have been because the demonstration covered some of the main uses of each tab, or it may be that students focussed more on what was immediately being taught rather than trying to discover how to use the software for themselves. In some cases, the time stamp for two adjacent logs for a participant were the same, and these are likely to be the result of the application interpreting a double click action as two separate single clicks, thus creating two identical log entries. A total of just 11 clicks on the Life Ring were made by 8 users, 2 clicks on the Associates Tab by the same user with the same time stamp, 1 click on the Collocations Tab, 4 clicks on the Graphs Tab (only two of which were made after a specific priming category had been selected, the other two made by the same student with the same time stamp). A further 2 clicks were made on the Search Tab, and 2 students clicked on the Life Ring when it was on the Please Wait Tab. If clicks made by the same student with the same time stamp are excluded and the clicks on the Please Wait Tab are also excluded, only 7 requests for help were actually made. No formal record of the use of the software manual was kept, but anecdotally, most students returned the copy of the manual at the end of the session, and very few seemed to have looked at it in detail.

A quarter of the all the search queries in the concordancer were made for words or word forms not part of the worksheet, and these were made by 13 different users. In the second questionnaire, students were asked to report on whether or not they had looked up words or phrases not connected with their task. Eleven students reported that they had, and 7 said that the search was useful and 4 interesting/fun, including one student who chose both useful and interesting/fun. Just 1 student said that this was a waste of time, but it is worth noting that overall this student was highly positive in his/her responses to the questions about the usefulness of each tab, having rated everything 5/5 except the Graphs Tab which was still rated positively at 4/5. These results might suggest that overall the software is likely to have potential for the kind of serendipitous learning which has been reported in DDL and “discovery learning” activities (e.g. Bernardini, 2004).

From these results, the most frequently used search support features seem to be those which can be found on the main search screen such as the spelling support and the auto-complete features for words and collocations, rather than the Life Ring help screens or the software manual.

Another set of questions on the second questionnaire related to whether or not students thought corpus examples, collocation information and the software itself would be useful for students like themselves. These questions were framed to be Yes/No questions with a required comment box to explain their reasons. Only 2 out of the 23 students who completed the second questionnaire responded negatively to the question of whether corpus examples were important. From their comments, it seems that both of these students were unsure of the relevance of the corpus examples to their own language production, with one stating “we do not use those examples very often”, and the other stating that he/she did not think it was useful for academic writing. However, of the vast majority of students who responded positively, 6 mentioned examples, 8 mentioned usage, 4 mentioned collocations, and 2 mentioned reliability. Encouragingly, one student wrote, “the examples helped me to think differently and get some information”, and another mentioned that corpus examples were useful because students have little opportunity to see how native speakers express themselves.

The second question in this group related to the importance of understanding collocations. All 23 students responded positively to this question. In the comments, 9 students mentioned the need for this kind of information to avoid making errors or to improve accuracy, and 8 students mentioned the importance of knowing how to use words.

The last question in this group asked students whether the software tool was useful. Twenty-two out of twenty-three students responded positively. The one student who selected “no” was one of the two students who used the software most after the Saturday session. However, the actual comment made by this student is still positive about the usefulness of the software; as is clear from the full response, his/her reservation is due to his/her belief that other software packages may be able to provide similar information in a more convenient way:

“It has many many tools and looks useful, but some important usage can be replace[sic] by other APP.”

Overall, it seems that the software was received very positively, especially considering that from the results of the first questionnaire it is very clear that very few students had used concordancers before. All but one of the students responded positively to the question about the usefulness of the software, and even the student who responded negatively did so in a highly positive manner. As explained earlier, two students chose not to complete the second questionnaire and their reasons for dropping out are not known. Neither student withdrew formally from the project and it is likely that other pressures such as coursework deadlines and mounting pressure for the final examinations may have influenced their choice not to complete the second questionnaire. Nevertheless, even if the non-participation of these students is interpreted as being lukewarm or negative towards the usefulness of the software overall, the proportion of positive responses as a total of all 25 participants is still 88%. Students who completed the second questionnaire gave a variety of reasons why they thought it was useful, with 4 mentioning being able to compare or see differences between words. Two mentioned the resources specifically. One student simply stated “It help [sic] students like a teacher”. Another student demonstrated a good understanding of how different resources will be suitable for different occasions:

“This software may not be my first choice when I look up a word, because [an] electronic dictionary is much more convenient. However, [the] function of the software is complete and I would like to use it as the complement of my first choice.”

One other student mentioned that it was not so “convenient” to use; however, 4 other students commented favourably on the “convenience” of the software. Another student focused specifically on the way in which the software can help students discover semantic associations of words writing:

“I think, it can tell us whether a word is positive or negative. This is interesting and useful!”

Other comments included positive evaluations of the software in terms of helping students to learn effectively (1), the amount of detail (3), and its potential in helping with academic writing (3). One student also said that it was useful for students from different “levels”.

The positive response is also evident in all of the responses to the question “In future, do you think you would like to use software like this again?” 10 out of 23 students chose “Yes, definitely”, and the remaining 13 chose “probably”. None of the students chose “Not sure”,

“Probably not”, or “Definitely not”. When asked to select from three situations when the software should be used, 7 chose “In class with a teacher”, 16 chose “In class for pairwork activities”, and 14 chose “Outside class independently”. Given that almost 70% of the students thought the software was suitable for pairwork, and 2 of these students had reported that they did not think peer activities were useful in the first questionnaire, it seems that the software may have potential to as a teaching tool to enhance pairwork tasks. Some details from the first questionnaire can provide some background as to the importance these results might have in terms of pairwork and peer editing activities. From the first questionnaire, it seems that most students have been asked by a teacher to review each other’s writing, with almost all of the students selecting “sometimes” rather than “regularly” or “never” in response to this question. It is worth noting that all three students who selected “never” for this response chose “sometimes” when reporting how often they choose to review each other’s essay outside class. Therefore, peer feedback seems to be part of every student’s study experience and it seems that the software may have potential as a teaching tool for pairwork. Furthermore, 6 of the students who selected “Outside class independently” did not choose any of the other options, meaning 26% of the students seem to consider the software most suited for independent work, and 60% suggested it could be used outside class. This suggests strongly that the software does seem to be easy enough to use for students outside class. Indeed, none of the students who chose “in class with a teacher” did not also select either or both of the other more student-led situations. Four students chose all three situations. The positive responses to the questions about corpus examples, collocation information and the software itself, coupled with these highly positive responses to questions about possible future uses of the software go some way to addressing the fourth part of the research question. However, one factor which needs to be considered in relation to these largely positive responses is that in China there is a cultural desire to please. It is hoped that the influence of this on the questionnaire responses was reduced through the precaution of not revealing that the software had been created by a member of staff at XJTLU until the debrief message was sent. Nevertheless, the results should be considered in the light of these cultural influences.

Finally, a number of suggestions for improvements were made by the students in the questionnaire. Out of the 23 students who answered the second questionnaire, 16 wrote something. Two students mentioned the need for more support for synonyms, although one of these did mention that he/she was not sure if support was already included.

However, interestingly, the other student suggested that different kinds of synonyms should be provided. This seems like a useful suggestion as the current list of 5 words based on the free Chinese-English dictionary or *WordNet* seems to be a little limited. Four students mentioned that they felt that the software was too complicated. Looking in more detail at these responses, 3 of the students gave an indication that this was a sense of it being too complicated when using it for the first time, or highlighted the need to emphasize some functions more than others. Three students requested a Chinese version, and two students mentioned the need for Chinese definitions or translations. Having a Chinese user-interface is an achievable goal and multi-language support for interface design is something which is available in the software development tool. However, providing definitions or translations in another language other than the one used in the corpora themselves is likely to be problematic and in a sense goes against the aims of the project. Despite such concerns, while students were writing the essays in the session, it was noticeable that many of them were referring to online English-Chinese dictionaries, and if development of a single tool solution is desirable, integrating access to web-based resources like these into the software might be one way to achieve this. However, it is more practical and probably better in terms of students' lifelong learning skills to encourage them to continue to make use of a variety of resources. Two students mentioned speed and one of these reported having some technical problems. This is likely to have been the result of the small software bug mentioned earlier, as the student who reported this did not use the software again after the main session. One student mentioned that he or she did not feel that so many concordance lines were necessary. This suggests that the aim of encouraging students to analyse concordance lines as a researcher may not have been fully met, and also demonstrates a lack in his or her understanding regarding the fact that the results presented in the software had not been manually selected as the best possible examples. Nevertheless, given the learning history of the students and the prevalent views of both the role of the teacher and the separation of grammar and vocabulary, this kind of misunderstanding is entirely predictable and would be something for a teacher to challenge over a longer period of time.

Summary

This chapter has presented the results of an evaluation of the software which was carried out through a face-to-face session followed by individualized feedback and a follow-up

questionnaire. The chapter has focussed on responding to the third research question, breaking this down into four parts. There is some evidence to show that the students found that the software was both useful and able to provide them with helpful insights. Some key points from the results are:

- The Cards, Lines and Collocations Tabs were considered to be very useful;
- Being able to compare words or phrases side by side was considered to be particularly positive.
- Evaluation of the Graphs Tab was more mixed, but the compare and filter by primings feature was not shown;

Actual use of the software as evidenced by the logs was actually quite limited, but the evaluation served its purpose in:

- Demonstrating students could see its benefit and use basic features with minimal input;
- Identifying some areas to develop further (e.g. synonym support and interface language).

The implications of these results in terms of future software enhancements will be considered in the next chapter. The next chapter will also consider how this study contributes to ongoing plans for possible broader evaluations including the scope and focus of these evaluations as well as methodological considerations.

Chapter 8: Conclusion

This closing chapter will draw together plans for future development of the software and plans for its future evaluation. Both of these threads will consider issues arising from the evaluation which was presented in the previous chapter as well as some of the limitations and unexplored avenues which have been noted in some of the earlier chapters looking at various aspects of the software itself. Before exploring these aspects of evaluation, two scenarios will be presented in order to suggest how the software might be used by teachers and students.

8.1 Scenarios

8.1.1 Scenario for a teacher's use of *The Prime Machine*

There are a number of ways that the software could be used in teacher-led situations, but this scenario will focus on the potential of the software as a tool for feedback on written work. Let it be assumed that the teacher has set a written assignment and the students have submitted their essays and are expecting some guidance and feedback on their work before making revisions and resubmitting a final draft. The teacher begins by reading through each assignment, highlighting or making comments on the student's language use according to the teacher's own preferred marking system. However, rather than supplying suggestions for reformulation or simply highlighting sections that contain an error, the teacher makes a list of some of the words or phrases for each student which have not been used correctly. These are selected according to whether or not the teacher feels that providing further examples or alternative expressions could be helpful, and they are selected according to the teacher's knowledge of each student's language level. The teacher would then choose some words and phrases from this list and check concordance lines for these and for the alternative words or phrases. For example, if a student has used the word *habit* in a context where *routine* would be more suitable, concordance lines for both of these could be prepared and the student could be asked to look at the contexts and to think about how the meanings and uses are different. The meanings and uses of different verbs could be shown to students by preparing concordance lines for collocations such as *control .. time* and *manage .. time*. Concordance lines could also be prepared for more straight-forward problems such as whether or not personal pronouns usually follow *approve*, or which prepositions are usually used with certain nouns. Some of the errors which students frequently make such as problems with *does not like* compared to *is not like* could be addressed by providing concordance lines for both of these expressions.

Using the double-click menu, the results for two or three pairs of expressions for each student would then be saved as files to be printed out or emailed back to them. If the teacher feels that a small number of results should be sufficient for the student to begin making appropriate changes, the results from a screen view of the data could be saved or printed out. If it is thought that providing many more examples could be helpful, the results could be saved as spreadsheets. By giving students the results as attachments or printouts, the teacher will help the students start to see how looking at examples from a corpus could be beneficial without requiring them to perform multiple lookups in the software. However, in order to provide guidance to the students for further exploration and direct consultation of the corpora, the teacher could add a list of other words or phrases, perhaps also adding some suggested alternatives or words with similar meanings which might usefully be compared. This would encourage students to use the software actively and to see its potential as a tool which they could use independently.

For larger groups, it would also be useful to select some words and phrases to present as group feedback. If, for example, many students made similar mistakes when using vocabulary items associated with the essay topic, the teacher could create handouts or slides showing concordance lines which demonstrate how these words should be used and how they are used differently in different contexts. This would also provide a good opportunity to explore differences between genres and registers. For example, many students use *besides* in academic writing, and some teachers may simply tell them it is not “academic”. However, by looking at concordance lines for *besides* in the *BNC: Academic* sub-corpus and comparing these with lines from *besides* in the *BNC: Newspapers* sub-corpus teachers could ask students to focus on the typical contexts and uses of this word in these two different kinds of writing, with *besides being*, and *much else besides* common in academic texts, while examples from newspapers show a predominance of the sentence initial (often paragraph initial) use with several examples showing quoted speech. This example would build on students’ prior knowledge and understanding of how *besides* can be used in certain text types, but demonstrate some of the uses with which they may be rather less familiar from academic texts.

These approaches benefit the teacher by:

- a) providing support to help students notice problems in their writing, while prompting the students to make the corrections themselves;

- b) providing teachers with evidence-based guidance which they can pass on to the students, avoiding the need to simply comment on the strangeness of expressions or dismissing words or expressions as being not “academic”;
- c) providing feedback which covers more aspects of the context than merely the grammatical form or the lexical choice; having evidence at their fingertips to be able to show tendencies for position in text, and other aspects of colligation and collocation;
- d) providing with some achievable follow-up activities, so students who want to start exploring other words and phrases can consult the concordance lines directly, while those who are more wary of starting to use concordancers can learn from the evidence provided in email attachments or printouts;
- e) engaging the students by asking them to think critically about how words and phrases are used in multiple examples, and encouraging them to think about the contexts and meanings related to each.

8.1.2 Scenario for students' use of *The Prime Machine*

The other scenario which will be presented here is for a group of students using the software during the writing process for an academic essay. These activities could take place in a computer lab or through access to laptop computers in the classroom. They could also take place outside the classroom on the students' own computers in unsupervised situations.

EAP teachers often encourage students to work through the writing process starting with brainstorming and making notes on the topic and planning the structure of the essay. When working through this stage, students could be encouraged to look up some of the words from the essay question or words which are associated with their chosen topic area and to explore some of the collocations. By looking at words with a similar meaning (for example synonyms or words with the same translation in their first language), they would also be able to see collocations for these and note down some of the main sub-topics and terminology associated with the subject of the essay. This should help students generate ideas to include in their essay as they see some of the strong collocations associated with the topic, and it should also help them see how to express some of the central concepts and avoid mis-collocations. For example, if an essay topic is related to health, by typing *healthy* into the concordancer with the *BNC: Newspapers* or *BNC: Other Publications* sub-

corpus selected the drop-down list of collocations reveals *healthy eating*, *heathy diet* in both corpora and *healthy lifestyle* or *keep .. healthy*.

As students write the first draft, they could consult the software to find examples of how to use words and phrases correctly. For example, if they are uncertain about word forms or irregular forms of verbs, the software could be used and concordance lines generated to show how these can be used. They may also find it helpful to access the software in order to look at how synonyms or words with similar meanings are used. For example, students studying business or finance modules alongside language classes may want to check to see how some of the vocabulary from their academic modules is used or is not used in less specialised contexts. Students might compare *goods* with *products*, or they might want to see differences between *product* and *production*. However, it would be important to consider learner differences and to ensure students felt free to focus on completing the first draft without being forced to use the concordancing software; although some students may want to use a concordancer during this stage, others may be more productive if they focus on expressing their ideas during this stage and are encouraged to focus on form later.

Following the completion of a first draft, students may be asked to revise and edit it independently. However, as was evident in the results which were presented in the previous chapter, peer editing is also a common activity for students. Peer editing typically involves students looking at a partner's essay and providing feedback on how it could be improved. The use of *The Prime Machine* in the editing stage or as part of a peer editing activity is something which could be done initially as a group activity. One way which peer editing can be directed in a classroom situation is by prompting students to read through their partner's essay to locate potential problems in specific areas. They could be given a set time to complete a short list of words or phrases which could be analysed, and then the teacher could ask the students to start using the concordancer to look up words or pairs of expressions in the system. For example, if a student was unsure whether an expression would be suitable for use in academic writing, the compare corpus mode could be selected. For words and expressions which are not highly specific to a particular field, comparing results from the *BNC: Academic* sub-corpus against those for the *BNC: Newspapers* sub-corpus should provide clear evidence not only of whether an expression is more common in one of these, but also whether there are differences in the way in which the expressions are used.

Chapter 8: Conclusion

While there has been some debate about the value of peer feedback, as a tool for editing and peer-editing, *The Prime Machine* provides clear benefits. If the feedback is purely based on each student's own language competence, students who are more confident about a language point may feel awkward about criticising their partners openly, while students who are less sure about a language point may feel reluctant to risk revealing their own lack of expertise. By using the concordancer, however, students can focus on tendencies of language use, rather than simply whether something is right or wrong. The ability to explore how words and phrases are used in different text types or fields also means that the exploration can help students focus on meaning for specific situations. In a peer-editing setting, students could be encouraged to highlight words or phrases in their partner's essay which they find "interesting", and then explore together whether alternative wordings seem to fit this context better or whether their suggested alternatives would be more suitable elsewhere. If peer-editing tasks are set up in this way, with the focus on exploration of different possibilities, the benefit is that both students in each peer-editing pair have a much greater chance of learning something from the experience. Rather than feeling that they are sitting in judgement on a peer, or that they are risking exposure of their own limitations, the students can share suggestions and explore how the words or phrases are used in the different texts shown in the concordancer. If it happens that both the original expression and the partner's suggested alternative are both suitable, both students gain an additional way of expressing an idea from one-another. If it happens that only one of the expressions is correct for the intended meaning and context, the student who used an unsuitable word or phrase can compare differences in the word choice or context with those of the more suitable expression. The other student would also receive positive feedback along with a set of examples to strengthen and deepen this knowledge.

As will be clear from the scenario for a teacher which was presented earlier, students could also make use of the software after they receive feedback from the teacher.

These approaches benefit the student by:

- a) raising awareness of collocation, and providing a means to find and select collocations for highly specific fields;
- b) raising awareness of the importance of genre;
- c) ensuring a focus on form for specific communicative purposes in both reading and writing;

- d) helping students notice and focus on specific features in the typical contextual environments of the words and phrases they are interested in;
- e) providing a platform for students to explore and engage with examples as a way of checking and correcting their own language use;
- f) opening up feedback and revision to be more about selecting from a range of possible choices and considering the suitability of each, rather than simply getting limited answers from translation dictionaries;
- g) developing analytical skills for life-long learning, so students can continue to use evidence from corpora to develop language competencies and to make expert language choices in the future.

8.2 Implications for future software development

The suggestions in this section will follow the order of issues as presented in this thesis, starting with considerations of the architecture and search screen and moving through features related to collocation, primings and finally key tags and associates.

In terms of software architecture, the system seems to have coped with the demand on resources well considering that the server was not integrated into the university network as originally planned. During the period in which this software was developed there has been an enormous increase in the number of people both in China and the West who use mobile technology, and the design of a small application which connects to a larger institutional server seems even more suitable than it did at the beginning of this project. While the software has potential as a funded external site serving users anywhere, there does seem to also be an increasing number of apps for mobile devices which have been customized for individual institutions. Clearly, a major limitation of the current version of the software is that it only runs on *Windows* operating systems. However, since the newer versions of *RAD Studio* allow *Delphi* code to be transformed into native code for *Linux*, *Mac OS*, *iOS* and *Android*, as well as 64 bit *Windows* and given that the *Dataspap* client functionality which allows *The Prime Machine* server to communicate with devices is also available for all these, developing a version for large sized tablets and other platforms should be very straight-forward. It would also seem desirable to make a version which would run on smaller tablets and mobile phones, but this would involve much more development as the visual design of the application would need to be changed so as to provide useful data for a smaller screen. Development for smart phones would, however,

provide a range of ways in which users could share material through interfaces with social media. This could be an interesting area given that teachers may wish to share their own results or results from different students as part of class or pair-work activities. The export features allowing tables, images and text to be copied provides some good integration with other software on a computer platform, but developing the application further to fit in with smart phone use should also take into consideration the way in which users may wish to share results across devices, or share findings with friends through content rich instant messaging services like *WeChat*⁶³ or virtual learning environments such as *Blackboard*⁶⁴ or *Moodle*⁶⁵.

In terms of further development of the computer based search screen, some further enhancements would seem sensible based on the feedback from students during the evaluation. Firstly, from responses to the questionnaire, synonyms seem to be particularly important both in terms of the way that students view the importance of collocation information and also from the point of view of the highly rated compare mode option in the software. The current drop-down list of 5 synonyms derived from a Chinese-English dictionary and/or *WordNet* seems too limited. Automatically grouping concordance lines according to different meanings is unlikely to be feasible unless other resources like the *UCREL* Semantic Tagger (Rayson, Archer, et al., 2004) or more detailed models based on collocation and colligation information are incorporated into the database. Development in this direction might diminish the overall aim of getting language learners to explore the examples and actively consider the meaning of each concordance line. However, providing more detailed thesaurus type information for the purposes of helping students select suitable words and phrases for comparison does seem much more feasible. A plus button or something similar could be placed on the search screen allowing users to request a fuller list of synonyms. The lists derived from the Chinese-English dictionary and *WordNet* would both allow for grouping of synonyms, and the impact of this on the performance of the server is likely to be negligible.

Moving to the features related to collocation, it seems that the display of log-likelihood collocations using the method presented in Chapter 4 was well received by the students.

⁶³ <http://www.wechat.com/en/>

⁶⁴ <http://www.blackboard.com>

⁶⁵ <http://moodle.org>

Given that little was discussed in the demonstration or feedback regarding semantic prosody or the emotion panels, it is notable that one of the students mentioned this specifically in the feedback on why the software was useful. Currently, the emotion information is based on a very limited lexicon and as well as looking at larger resources, another area of development could be in making it possible for the user to interact with the emotion panel data more. For example, providing access to the data in the form of a table might be useful. In addition, given the strong positive response to the usefulness of the compare mode by students in the evaluation, it could be that being able to filter or compare results according to the presence of words associated with specific emotions could be useful too.

The chapter on collocation (Chapter 3) also introduced the concordance line ranking methods available. It can be said that the project has been successful in terms of providing students with a list of concordance lines which they thought were useful, but during the demonstration which took place as part of the face-to-face session of the evaluation only the ranking methods based on the fixed random order and the log-likelihood and concordance bonding score were described or presented. One of the issues which came up as part of the demonstration was that the collocation “High Court” in the *BNC: Newspapers* sub-corpus was weighted too strongly when the latter ranking was used meaning almost all the results for 100 lines contained this. This problem was used in the presentation as an example of when using a random sample for a high frequency word may be more desirable. However, the evaluation did not explore the advantages and disadvantages of the concordance ranking methods and it is likely that further adjustments and developments could be made. Although two students mentioned speed as an area to improve, as has been mentioned earlier, the issue was probably made worse by the software bug which was only fixed after the face-to-face session, and in reality the speed of the application was more of an issue for the auto-complete features than for the retrieval of results. Because of network latency (the time it takes for a message to be sent from the server to the client and back), retrieving 100 lines is almost as fast as retrieving 20 or 30 lines. Providing higher numbers of results may be desirable and, as will be argued below, the ranking methods are likely to be an important area for further evaluation.

As was reported in the previous chapter, the features gathered on the Graphs Tab were rated less positively by students, although many still did consider them to be potentially “useful”. Nevertheless, an important point regarding the Graphs Tab is that it is by no

means the only way users of the software can gain access to the additional information about the contextual environment which *The Prime Machine* offers based on insights from the theory of Lexical Priming. The concordance cards show paragraphing and other aspects of textual colligation for each concordance line, and the Collocations Tab is also an important resource. Furthermore, because of the strong positive reaction to the functions of the software for comparing different words and phrases or the same word or phrase across two different corpora, it is likely that being able to filter or compare results according to primings would not only help students understand what each of these features represents, but also give them new ways in which to interact with the data. Now that the filtering function has been fully tested in the development context, it would be interesting and useful to try introducing this newer feature to students.

In terms of future software development priorities, it is clear that the range of features may need to be revised or extended for English, and it is also obvious that the features which have been currently implemented will need to be reviewed if the software is used to analyse corpora of other languages. It is likely that features like position in text, paragraph or sentence are likely to be useful across other languages, but the concept of Theme-Rheme would need to be changed for many languages and certainly the importance of articles, modal verbs, voice and other features are likely to need further revision. Although it should be possible to implement some of these in the *SQL* scripts using the script generation application, the client software would need to be redesigned to be more flexible in the creation of menu items and dock icons for priming features.

One of the issues which also arose from the responses to the questionnaires presented in the previous chapter was that some students considered the software to be a little over complicated. Given that the input session only lasted about half an hour, it could be that more extended use of the software in a classroom setting would reduce these misgivings, but another consideration might be to try to use “hot” markers to draw attention to less used features of the software if and when they are likely to contain interesting results. For example, the Tags Tab and the Associates Tab on the main tab component look exactly the same no matter whether there are data available for these or not. It may be possible to determine empirically a cut-off for when tag information is likely to be sufficiently different from an “average” word, or perhaps more fine-tuning of the range of tags included should be made for each corpus, so that only labels which are likely to be useful are displayed. Similarly it may be possible to automatically determine whether the evidence available on

the Associates Tab is likely to be useful. In these circumstances these tabs could be highlighted with a “hot” indicator or perhaps greyed out when they are likely to be less interesting.

8.3 Implications for the software as a learning and teaching tool

As well as the areas which have been identified for further software development, *The Prime Machine* could also be further evaluated both as a tool for linguistic analysis and in terms of Computer Aided Language Learning (CALL). Some further investigation regarding the range of priming features and ranking methods has already been proposed as part of longer term software development goals. However, this project has also introduced some new methods and new applications of statistical methods, and empirical testing of these has not yet been devised. Taking the measure for collocation as an example, as well as looking at the impact of being able to present the word with the node in order, the measure could also be evaluated according to the way in which it changes the ranking of collocates compared with other methods. Wermtter and Hahn (2006) make the point that when comparing the power of statistical methods researchers often just look at the ranking without considering the discrimination the methods have in terms of both false positives and true negative problems. To evaluate other collocation measures, they split their lists in half, and then divided each half into a further 4 portions to investigate how each statistic changed the position of terms, whether up or down, compared with raw frequency. It would be worth comparing the collocation measure introduced in this thesis with the measure based on Delta P which has been proposed by Gries (2013) against a pure frequency baseline, to evaluate whether items which are promoted and demoted down the list according to each statistic improve the ranking from a language learning point of view.

From a CALL perspective, there are also a number of areas where more research will be needed. Chapelle provides a set of principles and a comprehensive framework by which CALL software can be evaluated both in terms of what she calls “judgemental analysis of appropriateness” and “empirical evaluation of CALL tasks” (Chapelle, 2001, pp. 59, 68). The purpose of this section is to consider each of these principles and to measure to what extent the work presented in this thesis has already met each aspect and what kinds of evaluation may be needed in the future. Table 8.1 below shows the summary of these principles.

Table 8.1: Summary of principles for evaluating CALL, quoted from Chapelle (2001, p. 52) but presented in a different order.

Principle	Implication
Evaluation of CALL is a situation-specific argument.	CALL developers need to be familiar with criteria for evaluation which should be applied relative to a particular context.
Criteria should be applied in view of the purpose of the task.	CALL tasks should have a clearly articulated purpose.
Criteria for CALL task quality should come from theory and research on instructed SLA [Second Language Acquisition].	CALL evaluators need to keep up with and make links to research on instructed SLA.
Language learning potential should be the central criterion in evaluation of CALL.	Language learning should be one aspect of the purpose of CALL tasks.
CALL should be evaluated through two perspectives: judgemental analysis of software and planned tasks, and empirical analysis of learners' performance.	Methodologies for both types of analyses are needed.

As explained in Chapter 2, while it is hoped that this software will be useful in a wide range of contexts, there are several reasons why Chinese language learners of English may benefit greatly from being presented with the kind of linguistic information which the software provides. Much of the motivation for developing different aspects of the software stemmed from many years teaching language learners in China. The evaluation presented in Chapter 7 was clearly situated in the context of Higher Education level English studies for students in China, with a writing task which would be relevant to many hundreds of thousands of English language test takers. This evaluation met the first and fourth principles given in table 8.1. However, future research could explore the use and performance of students using the software in different contexts and for different task types. As well as exploring attitudes of students from different geographical, cultural and educational backgrounds, the software could be trialled with low level students as well as post-graduates. It would be interesting to explore how the software would be used for the drafting of actual university assignments, and it would also be useful to see how teachers and students would respond to the kinds of feedback which were provided by the researcher during the evaluation. Other than as a reference tool during the writing process or as a resource for teacher feedback on written assignments, concordancing software can clearly be used to support many other language learning tasks. From the questionnaire responses, further exploration of its potential for use in peer editing tasks would seem

useful. It would also be interesting to see how well the software can support specific communication or language problems during a range of other activities. These evaluations could, for example, be focused on specific kinds of information that learners may look up when completing comprehension, translation, or vocabulary building activities. Evaluation of the software as a tool for producing materials for language teaching could also be carried out.

The purpose of this thesis has been to explain how linguistic theories and pedagogic considerations have directed the design of the whole software project. In this sense, the success of the project in terms of language learning potential should be clear. There are a number of principles from SLA which are directly relevant to the evaluation of the software, particularly in the approaches underlying the scenarios which were outlined at the beginning of this chapter. First and foremost, there is the SLA principle that learners should be exposed to target language in use (Krashen, 1989). The software leads language learners to read multiple examples from authentic texts. A second principle from SLA is that importance of attention and noticing. Schmidt argues that “intake is what learners consciously notice” (1990, p. 149). Tomlinson argues that an important objective in language learning should be for learners to discover for themselves language features which can be found in the authentic texts they encounter, so as to strengthen the positive effects of noticing and recognising a gap in their own language use (Bolitho et al., 2003; Tomlinson, 1994, 2008). It is hoped that *The Prime Machine* goes some way to providing a platform for these kinds of discovery as it has been designed specifically to facilitate noticing of patterns and tendencies. The Lines Tab and the Cards Tab provide different layouts of the concordance data, with the aim of making different aspects of the contextual environment more noticeable. The different concordance line ranking methods and the icons indicating strong tendencies have also been designed to draw attention to different aspects.

Other principles from SLA research which are relevant to the use of the software as a feedback tool, peer-editing and for on-going self-tutoring relate to the importance of autonomy and motivation. Dickinson (1995) argues that increasing a sense of autonomy in learners can increase motivation levels and therefore make learning more effective. Bernardini argues that through using concordancing activities the teacher can create a “supportive, non-authoritarian environment” (2004, p. 28). Since *The Prime Machine* provides opportunities to develop learner autonomy through its built in search support

Chapter 8: Conclusion

features, and since makes it more straightforward to incorporate corpus consultation into classroom activities, it would seem fair to suggest that it goes some way to addressing these SLA recommendations. Clearly, as new evidence and theory from language study and teaching practice arises, the software will need to be re-evaluated and adapted. Future research should be done to evaluate the links suggested between these SLA theories and the learning processes and experiences of students using the software.

The final principle emphasises the importance of both measuring attitudes and performance. The study presented in the previous chapter only considered attitudes and the question of whether or not the software led to better writing or improved understanding and retention of appropriate linguistic forms remains open. However, these are clearly areas which could be extended and explored. Another aspect which should be considered is how the optimal defaults for the software as it stands should be established. Evaluation through versioning would seem to be an important means of evaluating these defaults further, and the potential of using the user-settings for this has already been described in Section 7.1.1 of Chapter 7.

As well as providing important principles, Chapelle (Chapelle, 2001) also draws together and presents 6 qualities which should be evaluated for CALL software, providing suggestions on judgemental analysis of CALL software (p53-4), appropriateness of task (p59) and empirical evaluation of the tasks (p68). The qualities are:

- Language learning potential
- Learner fit
- Meaning focus
- Authenticity
- Impact
- Practicality

The first quality, “Language Learning Potential” when applied to this project might include a judgemental analysis of the level of interactivity and the suitability of the range of target forms the software can provide. It would seem fair to award the software highly in this area since its very design encourages students to look up words themselves and to interact with the different tabs of data which are presented, and it also supports a wide range of comparisons between words and collocations or between corpora. It is also clear that the software has great potential for providing students access to a very wide range of target

forms, both in terms of the level of analysis from individual word types, to similar words and collocations, and in terms of the range of text types from different disciplines and genres which are contained in the corpora which have so far been used. The question of whether target forms are acquired and retained, as has been mentioned above, is still one which needs to be explored, but the responses to the second questionnaire as presented in Chapter 7 suggest that students were able to identify the importance of the software in supporting language use and accuracy and as a means of obtaining information about language.

In terms of the second quality, "Learner fit", the software would also seem to stand up very well. As a tool for exploring words and phrases the software provides a great amount of control. The questionnaire responses indicating how students viewed exploration of words or phrases not directly related to their essay writing also provides evidence that the software has potential for incidental or less directed learning. As Bernardini points out, corpus exploration can provide learners with an appreciation "that discoveries are often made when least expected" (Bernardini, 2004, p. 23). To facilitate autonomy and unsupervised exploration, one of the main aims for the design of the software was to provide more adequate support, hints and guidance to learners, as compared with other leading concordancers. Within the context of higher education, the software seems to have been very well received by students of different levels. The evidence from the questionnaire on how students reported using the software, the variation in their preference for different tabs of information and also the different views on how it could be used in future suggest that it might cater well for different learners with different learner styles. Since students were overwhelmingly positive, but positive about different aspects, it could be claimed that there is some empirical evidence that the software has succeeded in this respect. However, clearly longer-term attitudes and measurements of change in performance over time would need to be considered.

A focus on meaning also seems to be evident both from a judgement of the software and task, as well as empirical evidence in the form of questionnaire responses. The high rating of the compare feature suggests that students were interested in understanding how different words were used. The reported use of the software as part of a writing task also provides some evidence that students could see how the software could be used to help communicate their meaning effectively in writing, although as was mentioned earlier the logs suggest that these attitudes were probably based on their ideas about how the

Chapter 8: Conclusion

software could be used, rather than based on their actual experience using the software. Clearly, a longer study with log data matching reported views would be desirable.

In terms of “authenticity”, the task design was highly relevant given the number of students who go on to take language tests such as IELTS as well as tests for their EAP modules, but it lacked the authenticity of being actually part of the degree programme itself. However, the learners clearly demonstrated a belief that the software would be useful in the classroom or for self-study, and the overwhelmingly positive indication that they would definitely or at least probably want to make use of the software again in the future is good evidence that the software has to some extent met its aims as being a tool suitable for classroom or home use. Its potential for other tasks, as has been mentioned above, would need to be explored more thoroughly. As a system which allows learners to explore examples and consider possible language choices as a tool for their own language comprehension and production, the concordancer also adds authenticity to use of corpus materials in the way which Bernardini (2004) describes.

The “impact” of the software could be measured in terms of the comprehensiveness of feedback and software logs. While the log data was a little disappointing in terms of quantity, the evaluation has demonstrated that the level of detail which can be provided about different actions made by users of this system does have great potential. The unexplored features related to pins and star ratings could also provide more detail on attitudes towards the concordance data itself during a longer study. It is certainly clear that students rated the experience of using the software as a positive experience and in this respect the evaluation so far has been highly successful.

The limited evidence of actual use of the software, especially after the main face-to-face part of the evaluation, points to a need for further research in order to ensure that the positive impact in terms of the perceptions of the students would also follow through to a positive impact on longer-term use. One of the main limitations of the evaluation in terms of its face validity was that although the participants were completing a writing task suited to their learning context, the essay was not part of their formal studies and was administered towards the end of the semester when other pressures such as assessed coursework and upcoming exams may have meant they were less inclined to put the usual amount of care and attention into it. In order to encourage greater use of the software so that attitudes would be based on more direct and prolonged exposure to the interface and results, participants could be given opportunities to access it over a longer period. The

software needs to be made available so students can access it as and when they encounter language learning needs. Even in a shorter term study, if permission could be gained for students to bring with them early drafts of assignments or materials from their classes, participants would be much more likely to look up more words and phrases than when writing for an additional essay which may not have any long or short term benefits beyond general improvement of their language abilities.

Given the learning background of learners in China, it would be unrealistic to expect a sudden shift in their understanding of effective language learning processes, but the highly positive response to the software suggests that providing students with a new way of looking at language can be very effective, especially when supported by the kind of evidence which *The Prime Machine* can readily provide.

Of course, a very important consideration with any kind of teaching software is whether or not teachers will be interested and willing to make use of it and to recommend it to their students. The design of the software was made by drawing on my own fairly considerable experience as a language teacher and as a manager of language teachers. However, as Krishnamurthy and Kosem point out, it would be important to get feedback from teachers in a pilot scheme in order to ensure teachers will want to use it. Scott's own reflections on perceptions of the user-friendliness of *WordSmith Tools* include an important point that teachers need to have confidence in their own abilities to use software, and what it should be used for, otherwise their fears for loss of face can be an inhibiting factor (Scott, 2008). Clearly, further exploration of the perceptions of teachers and input from them will be a key to making *The Prime Machine* a well-used tool as well as a useful tool for language learning.

The last quality is that of "practicality". An issue related to practicality has also been discussed earlier in this chapter, where the importance of developing the client software further to work with other operating systems was considered. In terms of the middle tier server software and database requirements, the rationale for the software architecture which was adopted has already been presented in Chapter 3. The fact that the evaluation ran smoothly with a single server which was actually a desktop machine purchased in 2011 and was located outside the university local area network suggests that the minimum requirements are reasonable. In recent years there has been an increase in the performance of "virtual servers", where server applications run in parallel on a large powerful server, and the system resources are allocated dynamically as and when each

application requires. The size of the databases for this project are somewhat larger than those typically envisaged for these systems, but further advances in data compression and server specifications are likely to mean that the installation of *The Prime Machine* server and database through virtualization would minimize hardware costs. Another aspect of “practicality”, however, is the amount of expertise which is required to set up new corpora. The pre-processing of corpus data in new formats would require more input and customization that would probably be possible for an average school or university IT department. However, with the range of sub-corpora from the *BNC* and resources such as the *Hindawi* academic corpora, it is likely that for academic English the corpora which have already been pre-processed are likely to be a good starting point for language learners. The templates which were described in Section 3.5.2 of Chapter 3 also go some way to making this process more smooth, but further consideration regarding how it can be made more flexible but easier to use will be needed in the future.

8.4 Beyond language learning

The main aim of this project was to develop concordancing software which would make information about language based on Hoey’s theory of *Lexical Priming* (2005) available to language learners in a convenient and easy-to-understand way. However, beyond this, by extending corpus linguistic approaches and providing new ways to hold information in a database, the project has also provided several methods which could be useful for a special kind of language learner and also for research. With regard to register analysis as part of university linguistics programmes, the visual design and some of the methods have great potential. For countries like China where university programmes for English majors include deeper analysis of these kinds of language features, the software could provide useful opportunities for direct manipulation of language data. As a tool for linguistic research, *The Prime Machine* also has some potential, but it is likely that the decisions made during the pre-processing stages would not suit the complete range of researcher needs. Needing to decide on settings for frequency cut-offs, the amount of data which would be stored and aspects such as window size and other features before data can be viewed is not appropriate for many kinds of corpus linguistic research. One way forward would be to adopt an on-the-fly approach so that researchers could make use of some of the scripts while adding much more customization. It seems reasonable to argue that speed is of great importance for language learners, but for researchers a wait of several seconds or

minutes is not likely to be a great obstruction. This would also increase the flexibility, allowing English majors or researchers to import their own texts, and further extensions to the *SQL* scripts could be made specifically for register analysis and research needs.

Another area which is undeveloped is the visualization of different kinds of corpus data. The Cards Tab provides an interesting alternative way of viewing written text data in paragraph form with headings. Further development is needed to make other kinds of corpus data visually rich. For example, the spoken sub-corpus of the *BNC* has been processed with POS tags and divided neatly into turns and sentence-like structures, but other spoken corpora could benefit from having a more customized visual design for concordance data. As Mauranen (2004) points out, language learners will make different use of spoken corpora compared to written corpora due to the clear limitations imposed by the need to produce language in real-time. However, to introduce students to the importance of the interactivity of spoken language, and to provide clear examples for students beginning work on register analysis, more could be done to highlight features which would not be so obvious from KWIC displays. Developing a visualization for showing back-channelling or overlapping speech from a spoken corpus like *MICASE* (*MICASE*, 2007) would seem helpful. Corpora like *BASE* (*BASE*, 2006) where utterances are only divided by pauses and turns, however, would be even more challenging. Other ways of displaying collocation, priming tendencies, key tags and key associates could also be developed.

Summary

This chapter has focussed on two aspects of the evaluation of *The Prime Machine*. It has introduced software design features which have already been implemented and could in the future be activated in order to carry out further empirical research. It has considered the software design implications of the small scale evaluation which took place over a short period of a few days and was presented in the previous chapter. It has also considered the scope of this evaluation within a wider framework. Despite being somewhat limited in size and duration, the questionnaire-based study has provided interesting insights into the acceptability of the software, face validity and student attitudes before and after and has also provided some concrete areas for future development. While the remaining ground drawing on frameworks from Computer Aided Language Learning for detailed evaluation of the software as a learning and teaching tool is wide, this initial evaluation has served to demonstrate confidence that the project meets its overall aims. While there is also much

Chapter 8: Conclusion

scope for detailed evaluation of specific features and mark-up processes, as well as opportunities for performance enhancement of the computer processes behind the software, the examples from earlier chapters along with the participants' enthusiasm suggests that the software is providing some meaningful data and provides at least face validity for the hidden processes. Having provided some initial answers to questions of how useful the software could be, and how well it might be received, the final chapter has also considered broader implications and identified limitations and areas for future research not only restricted to evaluation, but also looking at corpus linguistic approaches, visualization of different kinds of corpus data, and future extensions for register analysis and research tools.

This thesis makes several contributions to methods in corpus linguistics. It has set out a definition and a means for calculating KeyTags and also added to the number of ways in which concordance lines can be ranked and selected. The Cards design provides a new way for users to view concordance lines with a design offering more context than typically visible in KWIC displays and incorporating features of paragraphing and headings. The thesis has also extended the use of key word analysis for indicating strong tendencies of words and phrases to occur in specific environments on a much wider range of features associated with Lexical Priming than has previously been done. The thesis also makes a contribution to language learning and teaching through the application of Lexical Priming theory to second language learning situations.

It is hoped that this software will prove to be a valuable tool for both students and teachers as they gain access to corpus information in new and interesting ways, and as they take new opportunities to explore evidence for the wide variety of ways in which words and combinations of words are primed for experienced speakers of the language.

Appendix 1 Two word collocation measures

	Measure	Source	Asymmetrical
F4.1	$\rho = F_c / (Z - F_n)$ $E = \rho F_n S$ $z = (K - E) / \sqrt{E q}$ Where $q=1-p$	Significant collocations; Berry-Rogghe, 1973, cited in Oakes, 1998:163.	Yes
F4.2	$C = \frac{cfo(F, P) \sqrt{\sum(\frac{1}{d})}}{f(P)}$ F is the collocate; d is the number of items between	Strength of collocations; Geffroy et al., 1973, cited in Oakes, 1998:166.	Yes
Formulae based on mutual information		All given in Oakes, 1998:171;	
F4.3	$SMC = \frac{a + d}{a + b + c + d}$		No
F4.4	$KUC = \frac{a}{2} \left(\frac{1}{a + b} + \frac{1}{a + c} \right)$		No
F4.5	$OCH = \frac{a}{\sqrt{(a + b)(a + c)}}$		No
F4.6	$FAG = \frac{a}{\sqrt{(a+b)(a+c)}} - \frac{1}{2\sqrt{(a+b)}}$		Yes
F4.7	$YUL = \frac{ad - bc}{ad + bc}$		No
F4.8	$MCC = \frac{a^2 - bc}{(a + b)(a + c)}$		No
F4.9	$\phi^2 = \frac{(ad - bc)^2}{(a + b)(a + c)(b + c)(b + d)}$		No
F4.10	$MI = \log_2 \frac{a N}{(a + b)(a + c)}$		No
F4.11	<i>Cubic association measure</i> $MI3 = \log_2 \frac{a^3 N}{(a + b)(a + c)}$		No
4.12	$LL = 2 \times (a \log a + b \log b + c \log c + d \log d - (a + b) \log(a + b) - (a + c) \log(a + c) - (b + d) \log(b + d) - (c + d) \log(c + d) + (a + b + c + d) \log(a + b + c + d))$		No

Appendix 1 Two word collocation measures

	Measure	Source	Asymmetrical
	Formulae based on mutual information	From <i>The Sketch Engine</i> statistics (Lexical_Computing_Ltd., 2014);	
4.13	$T\text{-Score} = \frac{f_{AB} - \frac{f_A f_B}{N}}{\sqrt{f_{AB}}}$		No
4.14	Minimum sensitivity $= \min\left(\frac{f_{AB}}{f_B}, \frac{f_{AB}}{f_A}\right)$		No
4.15	MI-log-prod $= MI\text{-Score} \times \log(f_{AB} + 1)$		No
4.16	Relative frequency $= \frac{f_{AB}}{f_A} 100$		Yes
4.17	Dice $= \frac{2f_{AB}}{f_A + f_B}$		No
4.18a	Log Dice $= 14 + \log_2 \frac{2 \cdot \ w_1, R, w_2\ }{\ w_1, R, *\ + \ *, *, w_2\ }$		
4.18b	LogDice $= 14 + \log_2 \frac{2f_{AB}}{f_A + f_B}$	From (Rychlý, 2008); adapted from x, y notation to follow A, B of others.	No

Appendix 2: Collocation measures for more than two words

	Measure	Source and Notes
F4.19	$p(w_i str) = \frac{freq(w_i)}{freq(str)}$ $H(str) = \sum_{i=1}^n -p(w_i str) \log p(w_i str)$	Shimohata, Sugio & Nagata, 1999 Following the extraction, overlapping or adjoining strings are combined and shorter components are filtered out when a ratio threshold is satisfied.
4.20	$EMI(x_1, x_2, \dots, x_n) = (n-1)\alpha + \log_2 F - \sum_{i=1}^m \log_2(F_i - F) + (n-m)\beta$	Zhang et al. 2009
	Measurements tested by Petrović, Šnajder & Bašić 2009	
4.21a	$G_0(I, w_1 \dots w_n) = \log_2 \frac{P(w_1 \dots w_n)}{\prod_{i=1}^n P(w_i)}$	
4.21b	$G_0(DICE, w_1 \dots w_n) = \frac{nf(w_1 \dots w_n)}{\sum_{i=1}^n f(w_i)}$	
4.22	$G_1(g, w_1 \dots w_n) = \frac{g(w_1, w_2 \dots w_n) + g(w_1 \dots w_{n-1}, w_n)}{2}$	
4.23	$G_2(g, w_1 \dots w_n) = \frac{g(w_1 \dots w_{\lfloor \frac{n}{2} \rfloor}, w_{\lfloor \frac{n}{2} \rfloor + 1} \dots w_n) + g(w_1 \dots w_{\lfloor \frac{n}{2} \rfloor}, w_{\lfloor \frac{n}{2} \rfloor + 1})}{2}$	
4.24	$G_3(g, w_1 \dots w_n) = \frac{1}{n-1} \sum_{i=1}^{n-1} g(w_i, w_{i+1})$	
4.25	$G_4(g, w_1 \dots w_n) = g(w_1 \dots w_{n-1}, w_2 \dots w_n)$	

Appendix 2: Collocation measures for more than two words

	Measure	Source and Notes
4.26	$G_5(g, w_1 \cdots w_n) = \frac{1}{n-1} \sum_{i=1}^{n-1} g(w_1 \cdots w_i, w_{i+1} \cdots w_n)$	
4.27	$G_6(g, w_1 \cdots w_n) = G_0(g, (w_1 w_2, w_2 w_3 \cdots w_{n-1} w_n))$	
4.28	$H(g, w_1 w_2 w_3) = \begin{cases} \alpha_1 G_0^*(g, w_1 w_2 w_3, \{w_2\}) & \text{if } stop(w_2) \\ \alpha_2 G_{4,6}^*(g, w_1 w_2 w_3, \emptyset) & \text{otherwise} \end{cases}$	Three variations of this, with different treatment of non-stop words.

Appendix 3: List of source files

Note on the source code

The suite of software programs which was developed for this degree of Doctor in Philosophy comprise over 49,500 lines of *Delphi* source code and over 12,750 lines of *SQL* scripts (some of which are much longer than the standard of 80 characters). A further 3,408 lines of *SQL* script were written in the form of a template used by the *Lexical Priming Script Generator* application to create an additional 12,883 lines of *SQL* scripts.

The *Delphi* code draws on standard programming libraries included in the *Delphi Enterprise 2010* edition, as well as several libraries which were purchased including *NativeXML*, *TMS Component Studio* and *TMS Advanced Charts*. The corpus refactoring application makes calls to *CLAWS*. As explained below, the *SQL* scripts draw on data from *WordNet*, *CCEDICT* and the NCR Emotion lexicon.

In developing *The Prime Machine*, I built up my knowledge of both *Delphi* and *SQL*, drawing on several programming textbooks which have been cited at various points in the thesis, and also through reading questions and responses posted by others on a host of online forums. In addition to the programming libraries which the source code draws on directly, there are a few sub-routines (approximately 335 lines in total) which were based on or inspired by freely available web posts, and these have been indicated in the list of source files and in the source code itself.

File formats for the source code and other files

The following lists provide information about the file formats and how they can be viewed on a computer which does not have any programming software installed.

1. *Delphi* files

- | | |
|------|--|
| .EXE | Fully compiled <i>Windows</i> applications |
| .DPR | <i>Delphi</i> project files are usually created automatically, but may contain some additional code. These can also be opened as plain text. |
| .PAS | <i>Delphi</i> source code files can also be opened as plain text. |
| .DFM | <i>Delphi</i> form files hold information about the visual design of the form, and are created automatically. They can be opened as plain text, but some images or other data may also be encoded within the file. |

Comments in *Delphi* are surrounded by curly brackets, bracket or asterisk pairs (* *). Comments can also be written on the rest of a single line of code, following double slashes //.

2. SQL scripts

- .TXT MySQL scripts stored as .TXT files can be opened as plain text. However, some lines are extremely long and line breaks may change if the file is opened with word-wrapping on.
- .SQL MySQL scripts stored as .SQL files can also be opened as plain text. Again, some lines are extremely long and line breaks may change if the file is opened with word-wrapping on. The .SQL extension has typically been used when the script was originally generated using *MySQL Workbench* or a custom written *Delphi* file. However, these files may have been edited using a text editor to add additional code or comments.

Comments in *MySQL* scripts are made by using two hyphens -- at the beginning of a line.

3. Other files

- .TXT Plain text files.
- .RTF Rich text files can be opened by word processors (e.g. *Microsoft Word*).
- .PDF Should be compatible with any standard PDF reader.
- .TMSGrid This format can only be opened using the *TMS TAdvStringGrid* component. This is used for some of the resources which are embedded in the application files. The grids are also available as .XLS workbooks.
- .XLS This format for *Microsoft Excel 97-2003* workbooks can be opened by spreadsheet applications including *Microsoft Excel*.
- .tPMR These template files were created for this project and can only be opened properly by the Refactoring Application. The actual format is the *Delphi TClientDataSet* component binary file.

List of source files for the Refactoring Application

Main *Delphi* application

- ❖ RefactorXMLCorpus2.exe (fully compiled executable file for *Windows*)
- ❖ RefactorXMLCorpus2.dpr (Project file)

Appendix 3: List of source files

Requires:

- Standard *Delphi 2010 Enterprise* libraries, tested with *RAD Studio 2010 Enterprise Edition*, Copyright Embarcadero Technologies, Inc.
 - *NativeXML*, tested with Version 4.07, Copyright Simdesign BV. Some small enhancements were made to this component; details are provided in RefactorXMLCorpusUnit1b.pas.
 - *TMS Component Pack*, tested with Version 6.9.3.0, Copyright TMS Software
 - *CLAWS4* (to be installed in C:\Other Programs\WinCLAWS), tested with Version 22, Copyright UCREL, University of Lancaster
- ❖ RefactorXMLCorpusUnit1b.pas (Source code for main form)
 - ❖ RefactorXMLCorpusUnit1b.dfm (Form file for main form)
 - RefactorXMLCorpusUnit2b.pas

Subfolder AdvGridDefaults

Default tables of data are stored as resources in the application, using the *AdvStringGrid* file format. They are listed here in both *TMS AdvStringGrid* and *Microsoft Excel* file formats.

- ❖ CLAWS Text Replacements.TMSGrid (CLAWS Text Replacements.xls)
- ❖ ClawsC7Tags.TMSGrid (ClawsC7Tags.xls)
- ❖ PrimingTags.TMSGrid (PrimingTags.xls)

Subfolder Rules

This is a list of files containing rules for refactoring corpora.

- ❖ BAWE
 - PreProcess BAWE 18.xls
 - Rules for BAWE v23.xls
- ❖ BNC
 - Rules for BNC 19.xls
 - Rules for BNC 19 subcorpora no lookup.xls
 - Rules for BNC 19 subcorpora with lookup.xls
- ❖ Financial Times
 - Rules for FT 7.xls
- ❖ Guardian
 - PreProcess Guardian.xls
 - Rules for Guardian 4.xls
- ❖ Hindawi Corpora

Appendix 3: List of source files

- Rules for Hindawi v23.xls
- ❖ Springer Open
 - PreProcess Springer Open Supp 2.xls
 - Rules for Springer Open 25.xls
- ❖ WECCL
 - PreProcess WECCL 6.xls
 - Rules for WECCL v5.xls

Subfolder Lookup Tables

This is a list of files containing lookup information for text categories. Details of how these lists were created is provided in Chapter 6.

- ❖ BNC
 - BNC_SubLookup_Newspapers.xls
 - BNC_SubLookup_Spoken.xls
 - BNC_SubLookup_Unpublished.xls
- ❖ Hindawi Corpora
 - Biological Sciences Hindawi Lookup Table October 2014.xls
 - Chemistry Hindawi Lookup Table October 2014.xls
 - Computer Science Hindawi Lookup Table October 2014.xls
 - Earth Sciences Hindawi Lookup Table October 2014.xls
 - Engineering Hindawi Lookup Table October 2014.xls
 - Mathematics Hindawi Lookup Table October 2014.xls
 - Physics Hindawi Lookup Table October 2014.xls
 - Social Sciences Hindawi Lookup Table October 2014.xls
- ❖ Springer Open
 - Springer Open Categories December 2014.xls

Subfolder Templates

The application has its own file format for templates, enabling easy retrieval of previously stored corpus refactoring settings. This is a list of template files.

- ❖ BAWE.tPMR
- ❖ BNC Complete.tPMR
- ❖ BNC Newspapers.tPMR
- ❖ BNC Spoken.tPMR
- ❖ BNC Subcorpus No lookups.tPMR

Appendix 3: List of source files

- ❖ BNC Unpublished.tPMR
- ❖ FT.tPMR
- ❖ Guardian.tPMR
- ❖ Hindawi Biological Sciences.tPMR
- ❖ Hindawi Chemistry.tPMR
- ❖ Hindawi Computer Science.tPMR
- ❖ Hindawi Earth and Environment.tPMR
- ❖ Hindawi Engineering.tPMR
- ❖ Hindawi Maths.tPMR
- ❖ Hindawi Physics.tPMR
- ❖ Hindawi Social Sciences.tPMR
- ❖ SpringerOpen.tPMR
- ❖ WECCL.tPMR

List of source files for the Lexical Priming Script Generator

Main *Delphi* application

- ❖ LexicalPrimingScriptGenerator.exe (fully compiled executable file for *Windows*)
- ❖ LexicalPrimingScriptGenerator.dpr (Project file)
 - Requires:
 - Standard *Delphi 2010 Enterprise* libraries, tested with *RAD Studio 2010 Enterprise Edition*, Copyright Embarcadero Technologies, Inc.
 - *TMS Component Pack*, tested with Version 6.9.3.0, Copyright TMS Software
- ❖ LexicalPrimingScriptGeneratorUnit1.pas (Source code for main form)
- ❖ LexicalPrimingScriptGeneratorUnit1.dfm (Form file for main form)
- ❖ LexicalPrimingScriptGeneratorTexts.txt (*SQL* templates saved in plain text format, also stored inside the main form).

Tables of rules

This is a list of files which contain rules for collocations, extensions and priming features.

- ❖ CollocationPatterns5.xls
- ❖ ExtensionPatterns3.xls
- ❖ PrimingFeatures7.xls

List of source files for the database compression and processing script

This is a list of the SQL Scripts for *The Prime Machine* database server. They have been tested with *MySQL 5.6 Command Line Client*. For information about *MySQL* see

<https://www.mysql.com/>

Filenames and processing order for corpora with more than one category

- ❖ CreateCorpusTables.sql
- ❖ disablethekeys.txt
- ❖ *Corpus Manager imports data from Refactoring Application*
- ❖ All_the_way_many_categories_20150130.txt
 - First_Few_20140917.txt
 - enablethekeys.txt
 - Compress_Corpus_Words_20140917.txt
 - RemoveDuplicateSectionAndAuthorMetadata20140914.txt
 - alter_cb_info_for_cutoffs_20140106.txt
 - Various_Procedures_as_Stored_Procedures_20140106.txt
 - Priming_Script_20140106.SQL (Priming_Script_20140106_NOTE.TXT)
 - Metadata_Primings_With_Authors_20140917.txt
 - Mark_Duplicates_20140912.txt
 - Next_few_20140816.txt
 - Simple_MI_Collocations_20140816.txt
 - MWU_Script_20140109.SQL (MWU_Script_20140109_NOTE.TXT)
 - Mark_Extensions_20140107.txt
 - Cull_Collocations_20140107.txt
 - Concordance_Ranking_20140312.txt
 - Setup_Similar_Meaning_Tables_20140107.txt
 - Last_few_20150130_many_categories.txt
 - Check_Lexicon_For_Keyword_20140208.txt
 - Cull_Duplicate_Author_Metadata_20150130.txt
 - Key_Associates_Many_Categories_20150122.txt
 - Cull_Associate_MWUs5MWU_20150122.txt
 - Cull_Associate_MWUs4MWU_20150122.txt
 - Cull_Associate_MWUs3MWU_20150122.txt
 - Cull_Associate_MWUs2MWU_20150122.txt

Appendix 3: List of source files

- Cull_Associate_MWUs_20150122.txt
- Delete_Unused_Cols_and_Add_Index_20140326.txt
- CreateConcPairSummaryTables20140326.txt
- Storetop100s_20140326.txt
- ❖ *Corpus Manager updates Grants for database and adds to list of available corpora using Corpus Manager Application.*

Filenames and processing order for corpora with only one category

- ❖ CreateCorpusTables.sql
- ❖ disablethekeys.txt
- ❖ *Corpus Manager imports data from Refactoring Application*
- ❖ All_the_way_one_category_20150130.txt
 - First_Few_20140917.txt
 - enablethekeys.txt
 - Compress_Corpus_Words_20140917.txt
 - RemoveDuplicateSectionAndAuthorMetadata20140914.txt
 - alter_cb_info_for_cutoffs_20140106.txt
 - Various_Procedures_as_Stored_Procedures_20140106.txt
 - Priming_Script_20140106.SQL (Priming_Script_20140106_NOTE.TXT)
 - Metadata_Primings_With_Authors_20140917.txt
 - Mark_Duplicates_20140912.txt
 - Next_few_20140816.txt
 - Simple_MI_Collocations_20140816.txt
 - MWU_Script_20140109.SQL (MWU_Script_20140109_NOTE.TXT)
 - Mark_Extensions_20140107.txt
 - Cull_Collocations_20140107.txt
 - Concordance_Ranking_20140312.txt
 - Setup_Similar_Meaning_Tables_20140107.txt
 - Last_few_20150130_many_categories.txt
 - Check_Lexicon_For_Keyword_20140208.txt
 - Cull_Duplicate_Author_Metadata_20150130.txt
 - Key_Associates_Based_on_Ref_Corpus_20150122.txt
 - Cull_Associate_MWUs5MWU_20150122.txt
 - Cull_Associate_MWUs4MWU_20150122.txt
 - Cull_Associate_MWUs3MWU_20150122.txt

Appendix 3: List of source files

- Cull_Associate_MWUs2MWU_20150122.txt
 - Cull_Associate_MWUs_20150122.txt
 - Delete_Unused_Cols_and_Add_Index_20140326.txt
 - CreateConcPairSummaryTables20140326.txt
 - Storetop100s_20140326.txt
- ❖ *Corpus Manager updates Grants for database and adds to list of available corpora using Corpus Manager Application.*

List of source files for the one time setup and other metadata updates

This is a list of additional SQL Scripts for *The Prime Machine* database server. They have been tested with *MySQL 5.6 Command Line Client*. For information about *MySQL* see <https://www.mysql.com/>

Scripts to set up corpus administration database

- ❖ CreateCorpusAdminDatabase.sql
- ❖ createtableforsettings20140318.txt
- ❖ Grants_20140108.txt

Scripts to set up links to various resources

- ❖ createCCEDICTLinksbinary.txt

Requires additional `c:/sql_scripts/OneTimeOnly/justlinks.csv` file to be available. This CSV file was created by taking the raw data from the dictionary download, removing the column containing the main Chinese headwords and transforming it into this format. The dictionary is available from <http://www.mdbg.net/chindict/chindict.php?page=cedict>

- ❖ wordnet_to_binary.txt

This script should be run once for each server installation after the standard WordNet database has been installed, and before any corpora are refactored and processed. It has been tested with WordNet Release 2.0, Copyright 2003 by Princeton University.

- ❖ NCR_emotion_lexicon_binary_20140107.txt

Appendix 3: List of source files

The CSV file which is required for this script was created by taking the raw data from the NRC emotion download and transforming it into comma separated variables. The NRC emotion lexicon was created by Mohammad, S. M. and Turney, P. D. (2012).

❖ Create_Reference_Corpus_Script_20140107.txt

Requires a corpus to have already been fully processed to be used for the creation of a reference corpus.

Scripts to update citation data and metadata labels for specific corpora

❖ BAWE

➤ bawe_metadata_update_no_rename.txt

❖ BNC

➤ BNC_editing_of_citations_command.txt

➤ One of the following:

- bncmetadataachanges4.txt
- bncmetadataachanges4_Academic.txt
- bncmetadataachanges4_Fiction.txt
- bncmetadataachanges4_NonAcademic.txt
- bncmetadataachanges4_OtherPub.txt
- bncmetadataachanges4_SpokenNewsUnpub.txt

Script to improve speed after rebooting server machine

❖ Startup_20140511.txt

List of source files for the Corpus Management Application

Main *Delphi* application

❖ PrimeMachineManager.exe (fully compiled executable file for *Windows*)

❖ PrimeMachineManager.dpr (Project file)

Requires:

- Standard *Delphi 2010 Enterprise* libraries, tested with *RAD Studio 2010 Enterprise Edition*, Copyright Embarcadero Technologies, Inc.
- *TMS Component Pack*, tested with Version 6.9.3.0, Copyright TMS Software
- PrimeMachineClientIcons.pas from the Client Application folder

❖ AuthenticationManagerUnit1.pas (Source code for main form)

Appendix 3: List of source files

- ❖ AuthenticationManagerUnit1.dfm (Form file for main form)
- ❖ PrimeMachineManagerGetIPForm.pas (Source code for additional form)
 - Includes 8 lines of code for retrieving the IP address from a host name which were inspired by <http://stackoverflow.com/questions/18254209/how-to-get-the-ip-address-from-a-dns-for-a-host-name>
- ❖ PrimeMachineManagerGetIPForm.dfm (Form file for additional form)

Text for help screens providing explanations

This is a list of rich text files which contain explanations for each of the tabs in the software. These are embedded in the application as project resources.

- ❖ Authentication Options.rtf
- ❖ Corpora Available.rtf
- ❖ Tips.rtf

List of Source Files for the Server Application

Main Delphi application

- ❖ PrimeMachineDatanapinThreadServer.exe (fully compiled executable file for *Windows*)
- ❖ PrimeMachineDatanapinThreadServer.dpr (Project file)

Requires:

- Standard *Delphi 2010 Enterprise* libraries, tested with *RAD Studio 2010 Enterprise Edition*, Copyright Embarcadero Technologies, Inc. Two drivers are required for deployment:
 - The *MySQL* driver for dbExpress: dbxmys.dll (2009/11/19 6:05 283Kb)
 - ClientDataSet redistributable: midas.dll.res (2009/11/3 6:02 7Kb)
- The *MySQL* dynamic library file: libmysql.dll (2009/8/31 12:50 3,040Kb)
- ❖ PrimeMachineDatanapinThreadServerContainerUnit1.pas (Source code for DataSnap connections)
- ❖ PrimeMachineDatanapinThreadServerContainerUnit1.dfm (Form file for DataSnap connections)
- ❖ PrimeMachineDatanapinThreadServerMethodsUnit1.pas (Source code for server methods)
- ❖ PrimeMachineDatanapinThreadServerMethodsUnit1.dfm (Form file for server methods)
- ❖ SQLBasics1.pas (Source code for additional routines)

Other files

- ❖ The three redistributable resources listed under the requirements for this application.

List of source files for the Client Application

Core files for main *Delphi* application

- ❖ PrimeMachineDatasnapiThreadClient.exe (fully compiled executable file for *Windows*); this is usually renamed to tPM.exe when distributed.

- ❖ PrimeMachineDatasnapiThreadClient.dpr (Project file)

Requires:

- Standard *Delphi 2010 Enterprise* libraries, tested with *RAD Studio 2010 Enterprise Edition*, Copyright Embarcadero Technologies, Inc.
- *TMS Component Pack*, tested with Version 6.9.3.0, Copyright TMS Software. Some small enhancements were made to these components; details are provided in PrimeMachineMainSearchScreenUnit5.pas.
- *TMS Advanced Charts*, tested with Version 3.6.0.4, Copyright TMS Software. Some small enhancements were made to these components; details are provided in PrimeMachineMainSearchScreenUnit5.pas.
- *CheckPrevious* unit (129 lines) based on <http://delphi.about.com/od/windowshellapi/l/aa100703a.htm>
- ❖ PrimeMachineMainSearchScreenUnit5.pas (Source code for main form)
 - Includes approximately 40 lines of code based on suggestions from <http://www.scalabium.com/faq/dct0039.htm> and <http://www.delphigroups.info/2/3/322850.html> but actually implemented using my own code.
 - Includes approximately 150 lines of code based on advice on how to copy panels as jpeg from <http://www.delphigroups.info/2/2f/508061.html>
- ❖ PrimeMachineMainSearchScreenUnit5.dfm (Form file for main form)

Source code for other forms used in the *Delphi* application

- ❖ PrimeMachineClientGetIPForm.pas (Source code for Connect form)
 - Includes 8 lines of code for retrieving the IP address from a host name which were inspired by <http://stackoverflow.com/questions/18254209/how-to-get-the-ip-address-from-a-dns-for-a-host-name>
- ❖ PrimeMachineClientGetIPForm.dfm (Form file for Connect form)
- ❖ ClosingDialog.pas (Source code for Opening and Closing screen)

Appendix 3: List of source files

- ❖ ClosingDialog.dfm (Form file for Opening and Closing screen)
- ❖ EULAUnit1.pas (Source code for End User Licence Agreement screen)
- ❖ EULAUnit1.dfm (Form file for End User Licence Agreement screen)
- ❖ PrimeMachineAuthenticationPopup20130116Unit3.pas (Source code for pop-up browser for chosen authentication method)
 - Includes approximately 60 lines of code used to capture Enter key press on browser from <http://www.swissdelphicenter.ch/en/showcode.php?id=1055>
- ❖ PrimeMachineAuthenticationPopup20130116Unit3.dfm (Form file for pop-up browser for chosen authentication method)
- ❖ PrimeMachineClientIcons.pas (Source code for icon container and other images)
- ❖ PrimeMachineClientIcons.dfm (Source code for icon container and other images)
- ❖ PrimeMachineDatabaseConnections20130116Unit2.pas (Source code for local datasets)
- ❖ PrimeMachineDatabaseConnections20130116Unit2.dfm (Form file for local datasets)
- ❖ PrimeMachineDatasnapiinThreadClientUnit1.pas (Source code for form offering choice of authentication options)
- ❖ PrimeMachineDatasnapiinThreadClientUnit1.dfm (Form file for form offering choice of authentication options)
- ❖ PrimeMachineExtraDetailsUnit4b.pas (Source code for additional Life Ring pop-up grids)
- ❖ PrimeMachineExtraDetailsUnit4b.dfm (Form file for additional Life Ring pop-up grids)
- ❖ PrimeMachineSplashScreenUnit6 (Source code for splash screen shown while other forms are created)
- ❖ PrimeMachineSplashScreenUnit6.dfm (Form file for splash screen)

Source code for threads used in the *Delphi* application

- ❖ PrimeMachineClientAuthenticationOptionsThreadUnit1.pas (Source code for thread that retrieves authentication options)
- ❖ PrimeMachineClientAutoCompleteThreadUnit1.pas (Source code for thread that handles auto-complete functionality for the main search screen)
- ❖ PrimeMachineClientCardViewBuilderUnit1.pas (Source code for thread that takes downloaded results and transforms them into cards and lines, storing and retrieving data from the local cache as required)
- ❖ PrimeMachineClientCheckforMWUThreadUnit1.pas (Source code for thread used to check whether a multi-word unit exists in a specific corpus)
- ❖ PrimeMachineClientGetUserSettingsThreadUnit1.pas (Source code for thread used to retrieve user settings)

Appendix 3: List of source files

- ❖ PrimeMachineClientLexiconLookupThreadUnit1.pas (Source code for thread used to look up a single word in a corpus)
- ❖ PrimeMachineClientLexiconSeveralLookupThreadUnit3.pas (Source code for thread used to look up a list of words in a corpus)
- ❖ PrimeMachineClientPrimingFreqsThreadUnit1.pas (Source code for thread used to retrieve basic information for a corpus)
- ❖ PrimeMachineClientSaveLogsUnit1.pas (Source code for thread to transfer logs to the server).
- ❖ PrimeMachineClientSaveNewHintLevelUnit1.pas (Source code for thread used to save new tip level to the user's settings)
- ❖ PrimeMachineClientSaveUserSettingsUnit1.pas (Source code for thread to save user settings)
- ❖ PrimeMachineClientSoundsLikeThreadUnit1.pas (Source code for thread to retrieve list of words which have a similar sound, and a list of other corpora containing a word)
- ❖ PrimeMachineClient_AutoTags_Thread.pas (Source code for auto-complete functionality on the Tag Search screen)
- ❖ PrimeMachineClient_Get_Info_For_Sentence_Thread.pas (Source code for thread used to retrieve metadata about the currently selected concordance line)
- ❖ PrimeMachineClient_TagKeyWords_Thread.pas (Source code for thread to retrieve table of results for key words)
- ❖ PrimeMachineClient_Get_All_Conc_Data_Thread.pas (Source code for thread used to request concordance line data including collocations, etc.)

Source code for other units

- ❖ PrimeMachineThreadConnectionErrorHandler.pas (Source code for routines designed to handle connection problems)
- ❖ Wordcloudunit5.pas (Source code for routines used to create the clouds and associated tables)
- ❖ PrimeMachineClient_Feature_Helping_Unit.pas (Source code for calculating totals for the priming environments of current sets of concordance lines; part of this code is automatically generated by the *Lexical Priming Script Generator*)
- ❖ PrimeMachineDataSnapClientClasses.pas (Source code for procedures used for DataSnap methods; this is automatically created by the DataSnap proxy generator built into *Delphi Enterprise*).

Other files

- ❖ EmptyCache3.cds (a ClientDataSet structure for the local cache)
- ❖ End User License Agreement.rtf (a license agreement based on text provided for the project by the University of Liverpool's IP partner)
- ❖ GraphsTabExtraDetailSheets folder containing several .xls and .TMSGrid files (copies of the tables provided in Chapter 5 which also appear if the user requests additional details from the Life Ring help screens on the Graphs Tab)
- ❖ tPM User Manual Version 2.pdf (User manual) .

Bibliography

- ActiveState_Software_Inc. (2012). PDK 6.0 Documentation. Retrieved 12 March, 2013, from http://docs.activestate.com/pdk/6.0/PerlApp_overview.html
- Alexander, O., Argent, S., & Spencer, J. (2008). *EAP Essentials: A Teacher's Guide to Principles and Practice*. Reading: Garnet.
- Andrade, M. S. (2006). International students in English-speaking universities: adjustment factors. *Journal of Research in International Education*, 5(2), 131-154.
- Anthony, L. (2004). AntConc: A learner and classroom friendly, multi-platform corpus analysis toolkit. Paper presented at the Interactive Workshop on Language e-Learning, Waseda University, Tokyo.
- Anthony, L. (2011). AntConc (Windows, Macintosh OS X, and Linux) Build 3.2.4 Readme file.
- Anthony, L. (2014a). AntConc (Windows, Macintosh OS X, and Linux) Build 3.4.3 Readme file.
- Anthony, L. (2014b). Laurence Anthony's Website: Software. Retrieved 23 September, 2014, from <http://www.laurenceanthony.net/software.html>
- Anthony, L., Chujo, K., & Oghigian, K. (2011). A freeware, open-source, web-based framework for distribution and analysis of single and parallel corpora. Paper presented at the Corpus Linguistics Conference, Birmingham.
- Axialis_Team. (2011). *Axialis IconWorkshop Professional Edition* (Version 6.62 for Microsoft Windows). Retrieved from <http://www.axialis.com/>
- Baker, P. (2004). Querying keywords: Questions of difference, frequency, and sense in keywords analysis. *Journal of English Linguistics*, 32(4), 346-359.
- Baroni, M., Kilgarriff, A., Pomikálek, J., & Rychlý, P. (2006). WebBootCaT: A web tool for instant corpora. Paper presented at the EuraLex Conference, Torini, Italy.
- BASE. (2006). British Academic Spoken English Corpus: <http://www.ota.ox.ac.uk/desc/2525>.

- BAWE. (2007). British Academic Written English Corpus:
<http://ota.ahds.ac.uk/headers/2539.xml>.
- Beighley, L. (2007). *Head First SQL*. Beijing: O'Reilly.
- Bernardini, S. (2004). Corpora in the classroom: An overview and some reflections on future developments. In J. M. Sinclair (Ed.), *How to Use Corpora in Language Teaching* (pp. 15-36). Amsterdam: John Benjamins.
- Berry, M., & Halliday, M. A. K. (1996). *Meaning and Form: Systemic Functional Interpretations*. Norwood, NJ: Ablex.
- Biber, D., Connor, U., & Upton, T. A. (2007). *Discourse on the Move: Using Corpus Analysis to Describe Discourse Structure*. Amsterdam: John Benjamins.
- Biber, D., & Conrad, S. M. (2009). *Register, Genre, and Style*. Cambridge: Cambridge University Press.
- Biber, D., Johansson, S., Leech, G., Conrad, S., & Finegan, E. (1999). *Longman Grammar of Spoken and Written English*. Harlow: Longman.
- BNC. (2007). The British National Corpus (Version 3 BNC XML ed.): Oxford University Computing Services on behalf of the BNC Consortium. URL:
<http://www.natcorp.ox.ac.uk/>.
- BNC_Webmaster. (2007). BNC XML Edition (2007-02-08). Retrieved 8 March, 2013, from
<http://www.natcorp.ox.ac.uk/XMLedition/>
- Bolitho, R., Carter, R., Hughes, R., Ivanič, R., Masuhara, H., & Tomlinson, B. (2003). Ten questions about Language Awareness. *ELT Journal*, 57(3), 251-259.
- Boulton, A. (2010). Data-driven learning: Taking the computer out of the equation. *Language Learning*, 60(3), 534-572.
- Breiteneder, A., Klimpfinger, T., Majewski, S., & Pitzl, M.-L. (2009). The Vienna-Oxford International Corpus of English (VOICE) - A linguistic resource for exploring English as a lingua franca. *ÖGAI-Journal*, 28(1), 21-26.

- Brown, K., & Hood, S. (2002). *Academic Encounters: Reading, Study Skills, and Writing*. Cambridge: Cambridge University Press.
- Burnard, L. (2007a). BNC User Reference Guide: Design of the corpus. Retrieved 23 March, 2013, from <http://www.natcorp.ox.ac.uk/docs/URG/BNCdes.html#BNCcompo>
- Burnard, L. (2007b). BNC User Reference Guide: List of Sources. Retrieved 12 August, 2013, from <http://sara.natcorp.ox.ac.uk/docs/URG/bibliog.html>
- Burnard, L., & Baumann, S. (2013). TEI P5: Guidelines for Electronic Text Encoding and Interchange, 2.3.0: TEI Consortium.
- Chambers, A. (2007). Popularising corpus consultation by language learners and teachers. *Language & Computers*, 61(1), 3-16.
- Chambers, A., Farr, F., & O'Riordan, S. (2011). Language teachers with corpora in mind: From starting steps to walking tall. *Language Learning Journal*, 39(1), 85-104.
- Chambers, A., & O'Sullivan, Í. (2004). Corpus consultation and advanced learners' writing skills in French. *ReCALL*, 16(01), 158-172.
- Chambers, A., & Wynne, M. (2008). Corpora and Language Learning. In B. Barber & F. Zhang (Eds.), *Handbook of Research on Computer-Enhanced Language Acquisition and Learning* (pp. 438-452). Hershey: IGI Global.
- Chan, A. Y. W. (2011). Bilingualised or monolingual dictionaries? Preferences and practices of advanced ESL learners in Hong Kong. *Language, Culture and Curriculum*, 24(1), 1-21.
- Chapelle, C. (2001). *Computer Applications in Second Language Acquisition: Foundations for Teaching, Testing and Research*. Cambridge: Cambridge University Press.
- Charles, M. (2007). Reconciling top-down and bottom-up approaches to graduate writing: Using a corpus to teach rhetorical functions. *Journal of English for Academic Purposes*, 6(4), 289-302.
- Charles, M. (2012). Student corpus use: Giving up or keeping on? Paper presented at the TaLC10 Conference, Warsaw.

- Chau, M., Xiao, F., & Yang, C. C. (2007). Web searching in Chinese : A study of a search engine in Hong Kong. *Journal of the American Society for Information Science and Technology*, 58(7), 1044-1054.
- Church, K. W., & Gale, W. A. (1995). Poisson mixtures. *Natural Language Engineering*, 1(02), 163-190.
- Church, K. W., & Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1), 22-29.
- Cobb, T. (1997). Is there any measurable learning from hands-on concordancing? *System*, 25(3), 301-315.
- Cobb, T. (1999). Giving learners something to do with concordance output. Paper presented at the ITMELT '99 Conference, Hong Kong.
- Cobb, T. (2000). The Compleat Lexical Tutor, from <http://www.lextutor.ca>
- Collier, A. (1994). A system for automating concordance line selection. Paper presented at the NeMLaP Conference, Manchester.
- Collier, A. (1999). The Automatic Selection of Concordance Lines. Unpublished Ph.D. dissertation, University of Liverpool.
- Collins COBUILD Advanced Dictionary of English*. (2009). Glasgow: HarperCollins.
- Coniam, D. (1997). A practical introduction to corpora in a teacher training language awareness programme. *Language Awareness*, 6(4), 199-207.
- Conzett, J. (2000). Integrating collocation into a reading and writing course. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 70-87). Hove: Language Teaching Publications.
- Cotton, D., Falvey, D., & Kent, S. (2006). *Market Leader Upper Intermediate Course Book* (New ed.). Harlow: Longman.
- Cowie, A. P. (1999). *English Dictionaries for Foreign Learners: A History*. Oxford: Clarendon.
- Cox, K., & Hill, D. (2011). *EAP Now! : English for Academic Purposes* (2nd ed.). London: Pearson Longman.

- Coxhead, A. (2000). A new academic word list. *TESOL Quarterly*, 34(2), 213-238.
- Croft, W. B., Metzler, D., & Strohman, T. (2010). *Search Engines: Information Retrieval in Practice*. Boston: Addison-Wesley.
- Cunningham, S., & Moor, P. (1999). *Cutting Edge*. Harlow: Longman.
- Danielsson, P. (2007). What constitutes a unit of analysis in language? *Linguistik Online*, 31(2/07), 18.
- Delphi_Enterprise. (2010). *Delphi Enterprise*: Embarcadero Technologies. Retrieved from <http://www.embarcadero.com/products/delphi>
- Devitt, A., & Ahmad, K. (2013). Is there a language of sentiment? An analysis of lexical resources for sentiment analysis. *Language Resources & Evaluation*, 47(2), 475-511.
- Dickinson, L. (1995). Autonomy and motivation: a literature review. *System*, 23(2), 165-174.
- Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1), 61-74.
- Durrant, P. (2009). Investigating the viability of a collocation list for students of English for Academic Purposes. *English for Specific Purposes*, 28(3), 157-169.
- Durrant, P., & Schmitt, N. (2010). Adult learners' retention of collocations from exposure. *Second Language Research*, 26(2), 163-188.
- Eastman, C. M., & Jansen, B. J. (2003). Coverage, relevance, and ranking: The impact of query operators on web search engine results. *ACM Transactions on Information Systems (TOIS)*, 21(4), 383-411.
- Embarcadero. (2010). Rad Studio 2010 Help: Advantages of the multi-tiered database model. Retrieved 19 August, 2014, from http://docwiki.embarcadero.com/RADStudio/2010/en/Advantages_of_the_Multi-tiered_Database_Model
- Feinberg, J. (2009). Algorithm to implement a word cloud like Wordle. Retrieved 20 January, 2014, from <http://stackoverflow.com/questions/342687/algorithm-to-implement-a-word-cloud-like-wordle>

- Feinberg, J. (2013). Wordle. Retrieved 20 January, 2014, from <http://www.wordle.net/>
- Firth, J. R. (1957). Linguistic analysis as a study of meaning. In F. R. Palmer (Ed.), *Selected Papers of J R Firth 1952 - 59* (pp. 12-26). London: Longman.
- Firth, J. R. ([1951]1957). A synopsis of linguistic theory, 1930-1955. In F. R. Palmer (Ed.), *Selected Papers of J R Firth 1952-59* (pp. 168-205). London: Longman.
- Fletcher, W. H. (2004). Making the web more useful as a source for linguistic corpora. *Language and Computers*, 52(1), 191-205.
- Fligelstone, S. (1993). Some reflections on the question of teaching, from a corpus linguistics perspective. *ICAME*, 17, 97-109.
- Fontaine, L. (2013). *Analysing English Grammar: A Systemic-Functional Introduction*. Cambridge: Cambridge University Press.
- Frankenberg-Garcia, A. (2011). Beyond L1-L2 equivalents: Where do users of English as a Foreign Language turn for help? *International Journal of Lexicography*, 24(1), 97-123.
- Frankenberg-Garcia, A. (2012). Getting help from corpus examples. Paper presented at the TaLC10 Conference, Warsaw.
- Friedman, G. L. (2009). Learner-created lexical databases using web-based source material. *ELT Journal: English Language Teachers Journal*, 63(2), 126-135.
- Gabel, S. (2001). Over-indulgence and under-representation in interlanguage: Reflections on the utilization of concordancers in self-directed foreign language learning. *Computer Assisted Language Learning*, 14(3-4), 269-288.
- Gabrielatos, C., & Marchi, A. (2012). Keyness: Appropriate metrics and practical issues. Paper presented at the CADS International Conference 2012, University of Bologna, Italy. <http://repository.edgehill.ac.uk/4196/1/Gabrielatos%26Marchi-Keyness-CADS2012.pdf>
- Gao, X. (2010). To be or not to be "part of them": Micropolitical challenges in Mainland Chinese students' learning of English in a multilingual university. *TESOL Quarterly*, 44(2), 274-294.

- Garretson, G. (2007). What your words know: The theory of lexical priming. *International Journal of Corpus Linguistics*, 12(3), 445-452.
- Garretson, G. (2010). Corpus-Derived Profiles: A Framework for Studying Word Meaning in Text. Unpublished Ph.D. dissertation, Boston University.
- Garside, R., & Smith, N. (1997). A hybrid grammatical tagger: CLAWS4. In R. Garside, G. Leech & A. McEnery (Eds.), *Corpus Annotation: Linguistic Information from Computer Text Corpora* (pp. 102-121). London: Longman.
- Gaskell, D., & Cobb, T. (2004). Can learners use concordance feedback for writing errors? *System*, 32(3), 301-319.
- Gide, E., Wu, M., & Wang, X. (2010). The influence of internationalisation of higher education: A China's study. *Procedia - Social and Behavioral Sciences*, 2(2), 5675-5681.
- Greaves, C. (2009). ConcGram 1.0: User Manual. Retrieved 24 September, 2014, from <https://benjamins.com/series/cls/1/manual.pdf>
- Greaves, C., & Warren, M. (2007). Concgramming: A computer driven approach to learning the phraseology of English. *ReCALL*, 19(03), 287-306.
- Gries, S. T. (2013). 50-something years of work on collocations. *International Journal of Corpus Linguistics*, 18(1), 137-165.
- Gu, Y., & Johnson, R. K. (1996). Vocabulary learning strategies and language learning outcomes. *Language Learning*, 46(4), 643-679.
- Guardian_Corpus. (1990-1995). All the articles from the Guardian newspaper from 1st January 1990 to 31st January 1995.
- Hai, X. (2008). Exemplification policy in English learners' dictionaries. *International Journal of Lexicography*, 21(4), 395-417.
- Halliday, M. A. K., & Matthiessen, C. M. I. M. (2004). *An Introduction to Functional Grammar* (3rd ed.). London: Arnold.
- Hanks, P. (2013). *Lexical Analysis: Norms and Exploitations*. Cambridge: MIT Press.

- Harman, D., & Hoffman, D. (1996). Text Research Collection Volume 4; Financial Times Limited (1991, 1992, 1993, 1994): NIST.
- He, D., & Li, D. C. S. (2009). Language attitudes and linguistic features in the 'China English' debate. *World Englishes*, 28(1), 70-89.
- He, L., & Qi, L. (2010). Gui Shichun: Founding Father of Language Testing in China. *Language Assessment Quarterly*, 7(4), 359-371.
- Hemming, C., & Lassi, M. (2003). Copyright and the Web as Corpus. Paper presented at the Linguistic Resources Conference, Stockholm University.
<http://hemming.se/gslt/copyrightHemmingLassi.pdf>
- Henry, A. (2007). Evaluating language learners' response to web-based, data-driven, genre teaching materials. *English for Specific Purposes*, 26(4), 462-484.
- Hill, J. (2000). Revising priorities: from grammatical failure to collocational success. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 47-69). Hove: Language Teaching Publications.
- Hill, J., Lewis, M., & Lewis, M. (2000). Classroom strategies, activities and exercises. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 88-117). Hove: Language Teaching Publications.
- Hindawi. (2013). Hindawi's open access full-text corpus for text mining research. Retrieved 6 November, 2013, from <http://www.hindawi.com/corpus/>
- Hoey, M. (1991). *Patterns of Lexis in Text*. Oxford: Oxford University Press.
- Hoey, M. (2003). Why grammar is beyond belief. *Belgian Journal of English Language and Literatures* (special issue), 183-196.
- Hoey, M. (2005). *Lexical Priming: A New Theory of Words and Language*. London: Routledge.
- Hoey, M. (2014). Words and their neighbours. In J. R. Taylor (Ed.), *Oxford Handbook of the Word*. Oxford: Oxford University Press. Advance online publication. doi: 10.1093/oxfordhb/9780199641604.013.39.

- Hoey, M., & O'Donnell, M. B. (2008). Lexicography, grammar, and textual position. *International Journal of Lexicography*, 21(3), 293-293.
- Hogue, A. (1996). *First Steps in Academic Writing*. White Plains: Longman.
- Horst, M., Cobb, T., & Nicolae, I. (2005). Expanding academic vocabulary with an interactive on-line database. *Language Learning & Technology*, 9(2), 90-110.
- Hu, G. (2005). Contextual influences on instructional practices: A Chinese case for an ecological approach to ELT. *TESOL Quarterly*, 39(4), 635-660.
- Hu, G., & Alsagoff, L. (2010). A public policy perspective on English medium instruction in China. *Journal of Multilingual and Multicultural Development*, 31(4), 365-382.
- Huang, F. (2006). Internationalization of curricula in Higher Education institutions in comparative perspectives: Case studies of China, Japan and the Netherlands. *Higher Education*, 51(4), 521-539.
- Huang, G. (2002). Hallidayan linguistics in China. *World Englishes*, 21(2), 281-290.
- Hunston, S. (2002). *Corpora in Applied Linguistics*. Cambridge: Cambridge University Press.
- Hunston, S., & Francis, G. (2000). *Pattern Grammar: A Corpus-Driven Approach to the Lexical Grammar of English*. Amsterdam: John Benjamins.
- Internet_Engineering_Task_Force. (2012). The OAuth 2.0 Authorization Framework. Retrieved 2 July, 2014, from <http://tools.ietf.org/html/rfc6749>
- Jakeman, V., & McDowell, C. (2008). *New Insights into IELTS*. Cambridge: Cambridge University Press.
- Johns, T. (1986). Micro-concord: A language learner's research tool. *System*, 14(2), 151-162.
- Johns, T. (1988). Whence and whither classroom concordancing? In T. Bongaerts (Ed.), *Computer Applications in Language Learning* (pp. 9-27). Dordrecht: Foris.
- Johns, T. (1991). Should you be persuaded: Two samples of data-driven learning materials. In T. Johns & P. King (Eds.), *Classroom Concordancing* (Vol. 4, pp. 1-13). Birmingham: Centre for English Language Studies, University of Birmingham.

- Johns, T. (1994). From printout to hand out: Grammar and vocabulary teaching in the context of data-driven learning. In T. Odlin (Ed.), *Perspectives on Pedagogical Grammar* (pp. 293-313). Cambridge: Cambridge University Press.
- Johns, T. (2002). Data-driven Learning: The perpetual change. In B. Kettemann, G. Marko & T. McEnery (Eds.), *Teaching and Learning by Doing Corpus Analysis* (pp. 107-117). Amsterdam: Rodopi.
- Johnson, H. L., Baumgartner Jr, W. A., Krallinger, M., Cohen, K. B., & Hunter, L. (2007). Corpus refactoring: a feasibility study. *Journal of Biomedical Discovery & Collaboration*, 2, 4-14.
- Jurafsky, D., & Martin, J. H. (2010). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* (2nd ed. ed.). Beijing: Pearson Education Asia.
- Kaltenböck, G., & Mehlmauer-Larcher, B. (2005). Computer corpora and the language classroom: On the potential and limitations of computer corpora in language teaching. *ReCALL*, 17(01), 65-84.
- Kass, R. E., & Raftery, A. E. (1995). Bayes Factors. *Journal of the American Statistical Association* (430), 773.
- Kaszubski, P. (2007). Michael Hoey. Lexical priming: A new theory of words and language. *Functions of Language*, 14(2), 283-294.
- Katz, S. M. (1996). Distribution of content words and phrases in text and language modelling. *Natural Language Engineering*, 2(1), 15-59.
- Kaur, J., & Hegelheimer, V. (2005). ESL students' use of concordance in the transfer of academic word knowledge: An exploratory study. *Computer Assisted Language Learning*, 18(4), 287-310.
- Kennedy, C., & Miceli, T. (2010). Corpus-assisted creative writing: Introducing intermediate Italian learners to a corpus as a reference resource. *Language Learning & Technology*, 14(1), 28-44.
- Kennedy, G. D. (1998). *An Introduction to Corpus Linguistics*. London: Longman.

- Kenning, M.-M. (2000). Concordancing and comprehension: preliminary observations on using concordance output to predict pitfalls. *ReCALL*, 12(02), 157-169.
- Kettemann, B. (1995). On the use of concordancing in ELT. *TELL&CALL*, 4, 4-15.
- Kilgarriff, A. (2003). Thesauruses for natural language processing. Paper presented at the International Conference on Natural Language Processing & Knowledge Engineering, Beijing.
- Kilgarriff, A. (2007). Googleology is bad science. *Computational Linguistics*, 33(1), 147-151.
- Kilgarriff, A. (2009a). Corpora in the classroom without scaring the students. Paper presented at the 18th International Symposium on English Teaching, Taipei.
- Kilgarriff, A. (2009b). Simple maths for keywords. Paper presented at the Corpus Linguistics, Liverpool, UK.
- Kilgarriff, A., & Grefenstetter, G. (2003). Introduction to the special issue on the web as corpus. *Computational Linguistics*, 29(3), 333-347.
- Kilgarriff, A., Husak, M., McAdam, K., Rundell, M., & Rychlý, P. (2008). GDEX: Automatically finding good dictionary examples in a corpus. Paper presented at the Euralex, Barcelona.
- Kilgarriff, A., Rychlý, P., Kovár, V., & Baisa, V. (2012). Finding multiwords of more than two words. Paper presented at the 15th EURALEX International Congress, University of Oslo, Norway.
- Kilgarriff, A., Rychly, P., Smrz, P., & Tugwell, D. (2004). The Sketch Engine. Paper presented at the 2003 International Conference on Natural Language Processing and Knowledge Engineering, Beijing.
- Krashen, S. (1989). We acquire vocabulary and spelling by reading: additional evidence for the Input Hypothesis. *The Modern Language Journal*, 73(iv), 440-464.
- Kreyer, R. (2008). Corpora in the classroom and beyond. In B. Barber & F. Zhang (Eds.), *Handbook of Research on Computer-Enhanced Language Acquisition and Learning* (pp. 422-437).

- Krishnamurthy, R., & Kosem, I. (2007). Issues in creating a corpus for EAP pedagogy and research. *Journal of English for Academic Purposes*, 6(4), 356-373.
- Kyriacou, C., & Zhu, D. (2008). Shanghai pupils' motivation towards learning English and the perceived influence of important others. *Educational Studies (03055698)*, 34(2), 97-104.
- Lee, D. Y. W. (2001). Genres, registers, text types, domains, and styles: Clarifying the concepts and navigating a path through the BNC jungle. *Language Learning and Technology*, 5(3), 37-72.
- Lee, Y., Chen, A. N. K., & Ilie, V. (2012). Can online wait be managed? The effect of filler interfaces and presentation modes on perceived waiting time online. *MIS Quarterly*, 36(2), 365-394.
- Lewis, M. (2000a). Language in the lexical approach. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 126-154). Hove: Language Teaching Publications.
- Lewis, M. (2000b). Materials and resources for teaching collocation. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 186-204). Hove: Language Teaching Publications.
- Lewis, M. (2000). There is nothing as practical as a good theory. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 10-27). Hove: Language Teaching Publications.
- Lexical_Computing_Ltd. (2014). Statistics used in the Sketch Engine. Retrieved 26 September, 2014, from <http://trac.sketchengine.co.uk/raw-attachment/wiki/SkE/DocsIndex/ske-stat.pdf>
- Li, G., Chen, W., & Duanmu, J.-L. (2010). Determinants of international students' academic performance: A comparison between Chinese and other international students. *Journal of Studies in International Education*, 14(4), 389-405.
- Li, J., & Schmitt, N. (2009). The acquisition of lexical phrases in academic writing: A longitudinal case study. *Journal of Second Language Writing*, 18(2), 85-102.

- Li, Z., & Hao, H. (2009). *English Teaching in Chinese Context: Theory and Practice*. Hebei: Hebei Education Press.
- Liang, M. (2012). Patterned distribution of phraseologies within text: the case of academic English. Paper presented at the Corpus Technologies and Applied Linguistics, Xi'an Jiaotong-Liverpool University, Suzhou, China.
- Longman Dictionary of Contemporary English*. (2009). (5th ed.). Harlow: Pearson.
- Loucky, J. P. (2005). Combining the benefits of electronic and online dictionaries with CALL web sites to produce effective and enjoyable vocabulary and language learning lessons. *Computer Assisted Language Learning*, 18(5), 389-416.
- Macmillan English Dictionary for Advanced Learners*. (2007). (New ed.). Oxford: Macmillan.
- Madsen, R. E., Kauchak, D., & Elkan, C. (2005). Modeling word burstiness using the Dirichlet distribution. Paper presented at the 22nd International Conference on Machine Learning.
- Mair, C. (2002). Empowering non-native speakers: the hidden surplus value of corpora in Continental English departments. In B. Kettemann, G. Marko & T. McEnery (Eds.), *Teaching and Learning by Doing Corpus Analysis* (pp. 119-130). Amsterdam: Rodopi.
- Mascull, B., & Heitler, D. (2006). *Market Leader Upper Intermediate Teacher's Book* (New ed.). Harlow: Longman.
- Mauranen, A. (2004). Spoken corpus for an ordinary learner. In J. M. Sinclair (Ed.), *How to Use Corpora in Language Teaching* (pp. 89-105). Amsterdam: John Benjamins.
- McCarthy, M. (2004). *From Corpus to Course Book*. Retrieved from <http://www.cambridge.org/us/esl/touchstone/images/pdf/CorpusBooklet.pdf>
- McCarthy, M., McCarten, J., & Sandiford, H. (2006a). *Touchstone 3 Student's Book*. Cambridge: Cambridge University Press.
- McCarthy, M., McCarten, J., & Sandiford, H. (2006b). *Touchstone 4 Student's Book*. Cambridge: Cambridge University Press.

- MDBG. (2013). CC-CEDICT Download page. Retrieved 28 September, 2012, from <http://www.mdbg.net/chindict/chindict.php?page=cedict>
- Meyer, C. F. (2002). *English Corpus Linguistics: An Introduction*. Cambridge: Cambridge University Press.
- MICASE. (2007). Michigan Corpus of Academic Spoken English: <http://quod.lib.umich.edu/cgi/c/corpus/corpus?c=micase>.
- Miller, G. A. (1995). Word Net: A lexical database for English. *Communications of the ACM*, 38(11), 39-41.
- Mills, J. (1994). Learner autonomy through the use of a concordancer. Paper presented at the Meeting of EUROCALL, Karlsruhe, Germany.
- Mindt, D. (2002). A corpus-based grammar for ELT. In B. Kettemann, G. Marko & T. McEnery (Eds.), *Teaching and Learning by Doing Corpus Analysis* (pp. 91-105). Amsterdam: Rodopi.
- Mohammad, S. M., & Turney, P. D. (2012). Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 59.
- Mondria, J.-A., & Wit-de Boer, M. (1991). The effects of contextual richness on the guessability and the retention of words in a foreign language. *Applied Linguistics*, 12(3), 249-267.
- Nattinger, J. R., & DeCarrico, J. S. (1992). *Lexical Phrases and Language Teaching*. Oxford: Oxford University Press.
- Nesi, H. (1987). Do dictionaries help students write? Paper presented at the Annual Meeting of the British Association for Applied Linguistics, Reading, England.
- Niño, A. (2009). Internet and language teaching/learning: reflections on online emerging technologies and their impact on foreign-language instruction. In R. Oxford & J. Oxford (Eds.), *Second Language Teaching and Learning in the Net Generation* (pp. 23 - 31). Honolulu: National Foreign Language Resource Center, University of Hawaii.

- O'Donnell, M. B., Scott, M., Mahlberg, M., & Hoey, M. (2012). Exploring text-initial words, clusters and concgrams in a newspaper corpus. *Corpus Linguistics and Linguistic Theory*, 8(1), 73-101.
- O'Keeffe, A., McCarthy, M., & Carter, R. (2007). *From Corpus to Classroom: Language Use and Language Teaching*. Cambridge: Cambridge University Press.
- Oakes, M. P. (1998). *Statistics for Corpus Linguistics*. Edinburgh: Edinburgh University Press.
- Oshima, A., & Hogue, A. (1997). *Introduction to Academic Writing* (2nd ed.). White Plains: Longman.
- Oxford Collocations Dictionary for Students of English*. (2002). Oxford: Oxford University Press.
- Paice, C. (1977). *Information retrieval and the computer* (Vol. 26): Macdonald and Jane's Computer Monographs.
- Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A., & Booth, R. J. (2007). The development and psychometric properties of LIWC2007. Retrieved 18 June 2015, from http://homepage.psy.utexas.edu/homepage/faculty/pennebaker/reprints/liwc2007_languagemanual.pdf
- Pérez-Paredes, P., Sanchez-Tornel, M., Alcaraz Calero, J. M., & Jimenez, P. A. (2011). Tracking learners' actual uses of corpora: guided vs non-guided corpus consultation. *Computer Assisted Language Learning*, 24(3), 233-253.
- Petrović, S., Šnajder, J., & Bašić, B. D. (2010). Extending lexical association measures for collocation extraction. *Computer Speech & Language*, 24(2), 383-394.
- Phillips, M. A. (1985). *Aspects of Text Structure: An Investigation of the Lexical Organisation of Text*. Amsterdam: North-Holland.
- Ping-Fang, Y., & Jing-Chun, C. (2009). Semantic prosody: A new perspective on lexicography. *US-China Foreign Language*, 7(1), 20-25.
- Porter, M. (2007, 10 May). Guardian Unlimited - the new look explained. Retrieved from <http://www.theguardian.com/news/blog/2007/may/10/guardianunlimi12>

- Qiao, H. L., & Sussex, R. (1996). Using the Longman Mini-concordancer on tagged and parsed corpora, with special reference to their use as an aid to grammar learning. *System*, 24(1), 41-64.
- Raftery, A. E. (1986). A note on Bayes Factors for Log-Linear contingency table models with vague prior information. *Journal of the Royal Statistical Society. Series B (Methodological)*(2), 249.
- Ravelli, L. J. (1995). A dynamic perspective: Implications for metafunctional interaction and an understanding of Theme. In R. Hasan & P. H. Fries (Eds.), *On Subject and Theme: A Discourse Functional Perspective* (pp. 187-234). Amsterdam: John Benjamins.
- Rayson, P. (2002). Matrix: A Statistical Method and Software Tool for Linguistic Analysis through Corpus Comparison. Unpublished Ph.D. dissertation, Lancaster University.
- Rayson, P. (2008). From key words to key semantic domains. *International Journal of Corpus Linguistics*, 13(4), 519-549.
- Rayson, P., Archer, D., Piao, S., & McEnery, T. (2004). The UCREL semantic analysis system. Paper presented at the Beyond Named Entity Recognition Semantic Labeling for NLP Tasks Workshop, Lisbon, Portugal.
- Rayson, P., Berridge, D., & Francis, B. (2004). Extending the Cochran rule for the comparison of word frequencies between corpora. Paper presented at the 7th International Conference on Statistical Analysis of Textual Data, Louvain-la-Neuve, Belgium.
- Rayson, P., & Garside, R. (2000). Comparing corpora using frequency profiling. Paper presented at the Workshop on Comparing Corpora, Hong Kong University of Science and Technology, Hong Kong.
- Read, T. R. C., & Cressie, N. A. C. (1988). *Goodness-of-fit Statistics for Discrete Multivariate Data*. New York: Springer-Verlag.
- Renouf, A., Kehoe, A., & Mezquiriz, D. (2004). The accidental corpus: some issues in extracting linguistic information from the Web. In K. Aijmer & B. Altenberg (Eds.), *Advances in Corpus Linguistics: Papers from the 23rd International Conference on*

English Language Research on Computerized Corpora (ICAME 23) Goteborg 22-26 May 2002 (pp. 403-419). Amsterdam: Rodopi.

- Robb, T. (2003). Google as a quick 'n dirty corpus tool. *TESOL-EJ*, 7(2). Retrieved from <http://www.tesol-ej.org/wordpress/issues/volume7/ej26/ej26int/>
- Römer, U. (2004). A corpus-driven approach to modal auxiliaries and their didactics. In J. M. Sinclair (Ed.), *How to Use Corpora in Language Teaching* (pp. 185-199). Amsterdam: John Benjamins.
- Römer, U. (2009). Corpus research and practice: What help do teachers need and what can we offer? In K. Aijmer (Ed.), *Corpora and Language Teaching* (pp. 83-98). Amsterdam: John Benjamins.
- Ruan, Y., & Jacob, W. J. (2009). The transformation of College English in China. *Frontiers of Education in China*, 4(3), 466.
- Rundell, M. (1999). Dictionary use in production. *International Journal of Lexicography*, 12(1), 35-54.
- Rundell, M. (n.d.). Macmillan English Dictionary - How it was Created. Retrieved 16 January, 2014, from <http://www.macmillandictionaries.com/features/how-dictionaries-are-written/med/>
- Rychlý, P. (2008). A lexicographer-friendly association score. Paper presented at the Recent Advances in Slavonic Natural Language Processing Conference, Masaryk University, Brno.
- Schmidt, R. W. (1990). The role of consciousness in second language learning. *Applied Linguistics*, 11(2), 129-158.
- Schwartz, B., Zaitsev, P., Tkachenko, V., Zawodny, J. D., Lentz, A., & Balling, D. J. (2008). *High Performance MySQL* (2nd ed.). Sebastopol: O'Reilly.
- Scott, M. (1997). PC analysis of key words -- and key key words. *System*, 25(2), 233-245.
- Scott, M. (2000). Mapping key words to problem and solution. In M. Scott & G. Thompson (Eds.), *Patterns of Text: In Honour of Michael Hoey* (pp. 109-127). Amsterdam: John Benjamins.

- Scott, M. (2008). Developing WordSmith. *International Journal of English Studies*, 8(1), 95-106.
- Scott, M. (2010a). *WordSmith Tools* (Version 5.0). Oxford: Oxford University Press.
- Scott, M. (2010b). WordSmith Tools online manual "KeyWords: calculation". Retrieved 10 February, 2014, from http://www.lexically.net/downloads/version6/HTML/keywords_calculate_info.htm
- Scott, M. (2010c). WordSmith Tools online manual "KeyWords: purpose". Retrieved 10 February, 2014, from http://www.lexically.net/downloads/version6/HTML/keywords_calculate_info.htm
- Scott, M., & Tribble, C. (2006). *Textual Patterns: Key Words and Corpus Analysis in Language Education*. Amsterdam: John Benjamins.
- Shaoqun, W., Franken, M., & Witten, I. H. (2009). Refining the use of the web (and web search) as a language teaching and learning resource. *Computer Assisted Language Learning*, 22(3), 249-268.
- Shimohata, S., Sugio, T., & Nagata, J. (1999). Retrieving domain-specific collocations by co-occurrences and word order constraints. *Computational Intelligence*, 15(2), 92.
- Shin, D., & Nation, P. (2008). Beyond single words: the most frequent collocations in spoken English. *ELT Journal*, 62(4), 339-348.
- Simpson-Vlach, R., & Ellis, N. C. (2010). An academic formulas list: New methods in phraseology research. *Applied Linguistics*, 31(4), 487-512.
- Simpson, R. C., Briggs, S. L., Ovens, J., & Swales, J. M. (2002). *The Michigan Corpus of Academic Spoken English*: Ann Arbor, MI: The Regents of the University of Michigan.
- Sinclair, J. M. (1966). Beginning the study of lexis. In C. E. Bazell, J. C. Catford, M. A. K. Halliday & R. H. Robins (Eds.), *In Memory of J. R. Firth* (pp. 410-430). London: Longmans, Green and Co. Ltd.
- Sinclair, J. M. (1991). *Corpus, Concordance, Collocation*. Oxford: Oxford University Press.

- Sinclair, J. M. (2004). *Trust the Text: Language, Corpus and Discourse*: London : Routledge, 2004.
- Siyanova, A., & Schmitt, N. (2008). L2 learner production and processing of collocation: A multi-study perspective. *Canadian Modern Language Review/La Revue Canadienne des Langues Vivantes*, 64(3), 429-458.
- Sketch_Engine. (2013). 2013 prices. Retrieved 23 March, 2013, from <https://sketchengine.co.uk/?page=Website/Prices>
- Sorell, J., & Shin, D. (2007). The next step in concordance-based language learning: Constructing an online language learning resource for high-frequency vocabulary and collocations. *International Journal of Learning*, 13(12), 217-221.
- SpringerOpen. (2011). SpringerOpen's open access full-text corpus for text mining research. Retrieved 6 July, 2011, from <http://www.springeropen.com/about/datamining/>
- Staiano, J., & Guerini, M. (2014). Depeche Mood: A lexicon for emotion analysis from crowd-annotated news. Paper presented at the 52nd Annual Meeting of the Association for Computational Linguistics (ACL-2014), Baltimore.
- Stevens, V. (1991). Concordance-based vocabulary exercises: A viable alternative to gap-fillers. In T. Johns & P. King (Eds.), *Classroom Concordancing* (Vol. 4, pp. 47-63). Birmingham: Centre for English Language Studies, University of Birmingham.
- Stevens, V. (1995). Concordancing with language learners: Why? When? What? *CAELL Journal*, 6(2), 2-10.
- Stewart, D. (2010). *Semantic Prosody: A Critical Evaluation*. New York: Routledge.
- Sun, Y.-C. (2003). Learning process, strategies and web-based concordancers: a case study. *British Journal of Educational Technology*, 34, 601-613.
- Swart, B. (2009). Delphi 2010 DataSnap: Your data - where you want it, how you want it. Retrieved from <http://update.codegear.com/forms/AMUSCA1002DataSnapWhitepaper>
- Tahaghoghi, S. M. M., & Williams, H. E. (2007). *Learning MySQL*. Beijing: O'Reilly.

- Thomas, J., & Short, M. H. (1996). *Using Corpora for Language Research: Studies in Honour of Geoffrey Leech*. London: Longman.
- Thompson, G. (2004). *Introducing Functional Grammar* (2nd ed.). London: Arnold.
- Thurstun, J. (1996). Teaching the vocabulary of academic English via concordances. Paper presented at the Annual Meeting of the Teachers of English to Speakers of Other Languages, Chicago.
- Tickoo, M. L. (Ed.). (1989). *Learners' Dictionaries: State of the art*. Singapore: SEAMEO Regional Language Centre.
- Tidwell, J. (2010). *Designing interfaces* (Second ed.). Sebastopol: O'Reilly.
- TMS_Software. (2011). *TMS Component Studio* (Version 1.0): tmssoftware.com. Retrieved from <http://www.tmssoftware.com/site/studio.asp>
- TMS_Software. (2013). *TMS Advanced Charts* (Version 3.6): tmssoftware.com. Retrieved from <http://www.tmssoftware.com/site/advchart.asp>
- Tomlinson, B. (1994). Pragmatic awareness activities. *Language Awareness*, 3(3-4), 119-129.
- Tomlinson, B. (2008). Language acquisition and language learning materials. In B. Tomlinson (Ed.), *English Language Learning Materials: A Critical Review* (pp. 3-13). London: Bloomsbury Publishing.
- Tsui, A. B. M. (2004). What teachers have always wanted to know - and how corpora can help. In J. M. Sinclair (Ed.), *How to Use Corpora in Language Teaching* (pp. 39-61). Amsterdam: John Benjamins.
- Vannestål, M. E., & Lindquist, H. (2007). Learning English grammar with a corpus: Experimenting with concordancing in a university grammar course. *ReCALL*, 19(03), 329-350.
- Varley, S. (2009). I'll just look that up in the concordancer: integrating corpus consultation into the language learning environment. *Computer Assisted Language Learning*, 22(2), 133-152.

- Wattenberg, M., & Viégas, F. B. (2008). The Word Tree, an interactive visual concordance. *IEEE Transactions on Visualization & Computer Graphics*, 14(6), 1221-1228.
- Wen, Q., Liang, M., Yan, X., & Zhu, B. (2008). Spoken and Written English Corpus of Chinese Learners (2.0 ed.). Beijing, China: Beijing Foreign Studies University, Foreign Language Teaching and Research Press.
- Wermter, J., & Hahn, U. (2006). You can't beat frequency (unless you use linguistic knowledge): A qualitative evaluation of association measures for collocation and term extraction. Paper presented at the Annual Meeting of the Association for Computational Linguistics, Sydney.
- Whistle, J. (1999). Concordancing with students using an 'off-the-Web' corpus. *ReCALL*, 11(02), 74-80.
- Wible, D., Kuo, C.-H., Chien, F.-y., & Wang, C. C. (2002). Toward automating a personalized concordancer for Data-Driven Learning: A lexical difficulty filter for language learners. In B. Kettemann, G. Marko & T. McEnery (Eds.), *Teaching and Learning by Doing Corpus Analysis* (pp. 147-154). Amsterdam: Rodopi.
- Wilson, A. (2013). Embracing Bayes Factors for key item analysis in corpus linguistics. In M. Bieswanger & A. Koll-Stobbe (Eds.), *New Approaches to the Study of Linguistic Variability*. (pp. 3-12). Frankfurt: Peter Lang.
- Woolard, G. (2000). Collocation - encouraging learner independence. In M. Lewis (Ed.), *Teaching Collocation: Further Developments in the Lexical Approach* (pp. 28-46). Hove: Language Teaching Publications.
- www.ielts.org. IELTS | Researchers - Band descriptors, reporting and interpretation. Retrieved 16 January, 2014, from http://www.ielts.org/researchers/score_processing_and_reporting.aspx
- Xiao, R., & McEnery, T. (2006). Collocation, semantic prosody, and near synonymy: A cross-linguistic perspective. *Applied Linguistics*, 27(1), 103-129.
- Xu, H. (2009). *Towards Prototypical Exemplification in English Dictionaries for Chinese EFL Learners*. Beijing: Beijing Science Press.

- Yeh, Y., Liou, H.-C., & Li, Y.-H. (2007). Online synonym materials and concordancing for EFL college writing. *Computer Assisted Language Learning, 20*(2), 131-152.
- Yoon, C. (2011). Concordancing in L2 writing class: An overview of research and issues. *Journal of English for Academic Purposes, 10*(3), 130-139.
- Yoon, H. (2008). More than a linguistic reference: The influence of corpus technology on L2 academic writing. *Language Learning & Technology, 12*(2), 31-48.
- Zhang, J. (2008). Response of Chinese Higher Education and SJTU to Globalization: An Overview. In L. E. Weber, J. J. Duderstadt & C. Glion (Eds.), *The Globalization of Higher Education* (pp. 119). London: Economica.
- Zhang, Y., & Mi, Y. (2010). Another look at the language difficulties of international students. *Journal of Studies in International Education, 14*(4), 371-388.
- Zheng, X., & Adamson, B. (2003). The pedagogy of a secondary school teacher of English in the People's Republic of China: Challenging the stereotypes. *RELC Journal, 34*(3), 323-337.
- Zheng, Y., & Cheng, L. (2008). College English Test (CET) in China. *Language Testing, 25*(3), 408-417.