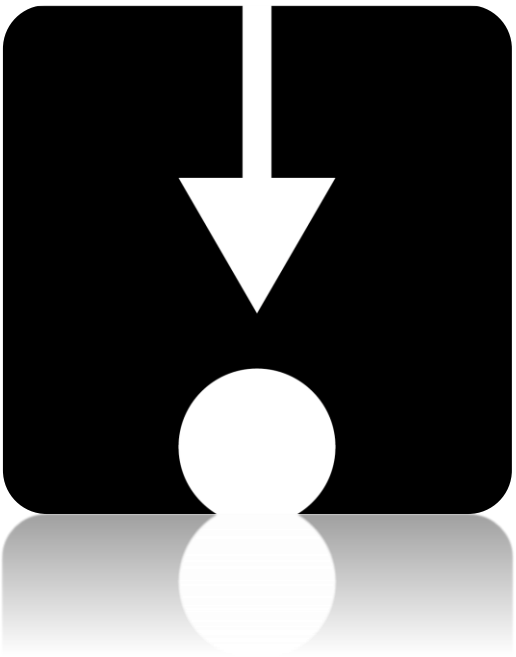
#LancsBox 4.5 manual



Citation for #LancsBox:

Brezina, V., McEnery, T. & Wattam, S. (2015). Collocations in context: A new perspective on collocation networks. *International Journal of Corpus Linguistics, 20*(2), 139-173

Brezina, V., Timperley, M., & McEnery, A. (2018). #LancsBox v. 4.x. [software package]

Lancaster University

Everyday
#LancsBox

Contents

1 Downloading and running #LancsBox version 4.5 4

2 Loading and importing data 6

2.1 Visual summary of Corpora tab 6

2.2 Load your corpora and wordlists 6

2.3 Supported file formats 7

2.4 Download #LancsBox corpora and wordlists 7

2.5 Working with corpora and wordlists 8

2.6 Saving corpora 8

2.7 Pre-processing of corpora (Advanced users). 8

3 Key functionalities 11

3.1 Mouse clicks 11

3.2 Shortcut Keys 11

3.3 Tools and Tabs 12

3.4 Split screen 12

3.5 Saving results 13

3.6 Copy/pasting selected results 13

4 KWIC tool (key word in context) 14

4.1 Visual summary of KWIC tab 14

4.2 Searching and displaying results 15

4.3 Settings and full text pop-up 15

4.4 Sorting, randomising and filtering 16

4.5 Statistical analysis 16

5 Whelk tool 18

5.1 Visual summary of Whelk tab 18

5.2 Top panel: KWIC 18

5.3 Bottom panel: Frequency distribution 18

5.4 Statistical analysis 19

6 GraphColl 20

6.1 Visual summary of GraphColl tab 20

6.2 Producing a collocation graph 20

6.3 Reading collocation table 21

6.4 Reading collocation graph 22

6.5 Extending graph to a collocation network 23

6.6 Problems with graphs: overpopulated graphs 24

6.7 Reporting collocates: CPN 25

7 Words tool 26

7.1 Visual summary 26

7.2 Producing frequency list 27

7.3 Visualizing frequency and dispersion 27

7.4 Producing keywords 28

7.5 Producing corpus statistics 28

8 Ngram tool 30

8.1 Visual summary 30

9 Text 32

9.1 Visual summary 32

9.2 Searching in Text 32

9.3 Settings 33

10 Searching in #LancsBox 34

11 Statistics in #LancsBox 36

11.1 Frequency measures 36

11.2 Dispersion measures 36

11.3 Keyword measures 36

11.4 Collocation measures 37

12 Glossary 38

13 Troubleshooting 41

14 Messages.Properties 42

#LancsBox v.4.5: License

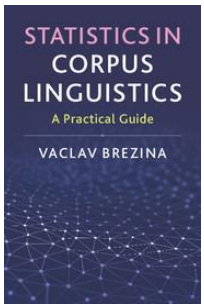
#LancsBox is licensed under BY-NC-ND Creative commons license. #LancsBox is free for non-commercial use. The full license is available from: http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode

#LancsBox uses the following third-party tools and libraries: Apache Tika, Gluegen, Groovy, JOGL, minlog, QuestDB, RSyntaxTextArea, smallseg, TreeTagger. Full credits are available http://corpora.lancs.ac.uk/lancsbox/credits.php

When you report research carried out using #LancsBox, please cite the following article:

- ☐ Brezina, V., McEnery, T. & Wattam, S. (2015). Collocations in context: A new perspective on collocation networks. *International Journal of Corpus Linguistics*, 20(2), 139-173
- ☐ Brezina, V., Timperley, M., & McEnery, A. (2018). #LancsBox v. 4.x. [software package].

Statistical help



Brezina, V. (2018). *Statistics for corpus linguistics: A practical guide*. Cambridge: Cambridge University Press.

If you are interested in finding out details about statistical procedures used in corpus linguistics, refer to Brezina (2018); visit also Lancaster Stats Tools online at http://corpora.lancs.ac.uk/stats

Further reading and materials

Brezina, V. (2016). Collocation Networks. In Baker, P. & Egbert, J. (eds.) *Triangulating Methodological Approaches in Corpus Linguistic Research*. Routledge: London.

Brezina, V. (2018). Statistical choices in corpus-based discourse analysis. In Taylor, Ch. & Marchi, A. (eds.) Corpus approaches to discourse: a critical review. Routledge: London.

Brezina, V. & Gablasova, D. (2017). The corpus method. In: Culpeper, J, Kerswill, P., Wodak, R., McEnery, T. & Katamba, F. (eds). *English Language (2nd edition).* Palgrave.

Brezina, V., McEnery, T. & Wattam, S. (2015). Collocations in context: A new perspective on collocation networks. *International Journal of Corpus Linguistics*, 20(2), 139-173.

Brezina, V., & Meyerhoff, M. (2014). Significant or random. *A critical review of sociolinguistic generalisations based on large corpora. International Journal of Corpus Linguistics*, *19*(1), 1-28.

Gablasova, D., Brezina, V., & McEnery, T. (2017). Collocations in corpus-based language learning research: Identifying, comparing, and interpreting the evidence. *Language Learning*, *67 (S1),* 155–179.

Gablasova, D., Brezina, V., & McEnery, T. (2017). Exploring learner language through corpora: comparing and interpreting corpus frequency information. *Language Learning, 67 (*S1), 130–154.

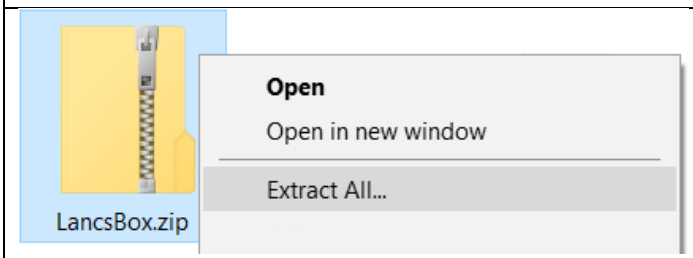
- More materials (video lectures, exercises, slides etc.) are available: on the #LancsBox website: http://corpora.lancs.ac.uk/lancsbox/materials.php

1 Downloading and running #LancsBox version 4.5

#LancsBox is a new-generation corpus analysis tool. Version 4.5 has been designed primarily for 64-bit operating systems (Windows 64-bit, Mac and Linux) that allow the tool's best performance. #LancsBox also operates on older 32-bit systems, but its performance is somewhat limited. Downloading and running it is very easy. It is done in three simple steps: 1) download, 2) extract and 3) run.

❶ Select and download: Select the version for your operating system and download to your computer.

❷ Extract (unzip) 'LancsBox'

Windows: This is usually done by right-clicking on the 'LancsBox.zip' file, which you have downloaded [note that the zip extension may be hidden on Windows], and by selecting 'Extract All'.	Mac: Double-click on LancsBox.app.tar
	

▶ **Note:** Make sure that 'LancsBox' is properly unzipped. If the zip file is only opened by double-clicking or selecting 'Open', #LancsBox won't run.

❸ Run #LancsBox: Depending on your operating system, do the following.

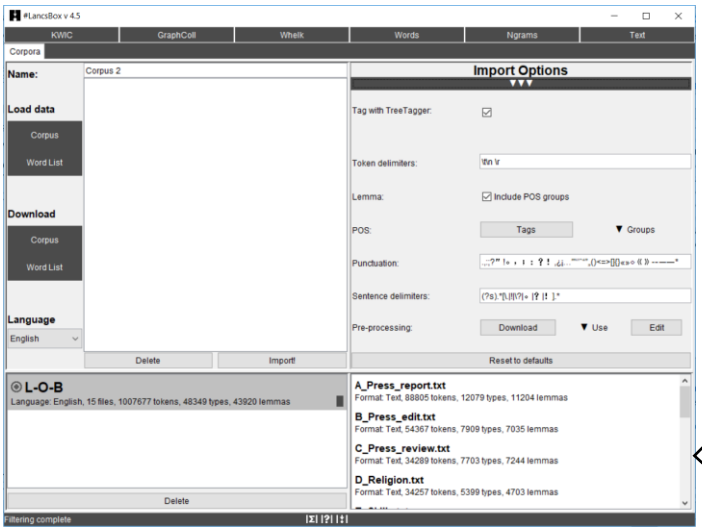
- Windows (any):
  >Double-click on 'LancsBox.bat' [note that the '.bat' extension may be hidden in Windows].
- Mac:
  >Copy LancsBox app from Downloads to Applications
  >Double-click on LancsBox App
  > Allow #LancsBox to run by giving the appropriate system security permissions.
  . Click on apple icon
  . Go to 'System Preferences' > 'Security & Privacy'
  . Allow #LancsBox to run

- Linux:
  > Make sure you have the oracleJDK / JVM installed (not the OpenJDK / JVM)
  > Set the permission to execute on Lancsbox.jar and all of the files in resources/tagger/bin
  > Run 'LancsBox.jar'

2 Loading and importing data

Data can be loaded and imported into #LancsBox on the 'Corpora' tab. This tab opens automatically when you run #LancsBox. #LancsBox works with corpora in different formats (.txt, .xml, .doc, .docx, .pdf, .odt, .xls, .xlsx, .zip etc.) and with wordlists (.csv). There are two options for loading corpora and wordlists: i) load (your own) data and ii) download corpora and wordlists that are distributed with #LancsBox.

2.1 Visual summary of Corpora tab



Top panel: Importing corpora and wordlists

You can:

- Select your corpus or wordlist to load.
- Download a corpora and wordlists distributed with #LancsBox.
- Select language.
- Review POS tags.
- Review punctuation marks and sentence delimiters.
- Set-up pre-processing via customisable scripts.

Bottom panel: Working with corpora and wordlists

You can:

- Activate or delete imported corpora or wordlists.
- Review corpus and text size (tokens, types, lemmas).
- Preview texts.
- Save processed corpora with pos-tags etc.

2.2 Load your corpora and wordlists

#LancsBox allows you to work easily with your own corpora and wordlists. These corpora are those stored on your computer or at a location accessible from your computer (memory stick, shared drive, dropbox, cloud etc.).

1. In the Corpora tab, left-click on 'Corpus' or 'Word List' under 'Load data', depending on whether you want to load a corpus or a wordlist.
2. This will open a window where you can navigate to the location (folder) where your corpus or wordlist is stored.
3. You can select a specific file, select multiple files by holding down Ctrl and left-clicking on your chosen files, or select all files in the folder by holding down Ctrl + A.
4. Left-click 'Open' to load your files.
5. Select the language of your corpus or wordlist. #LancsBox supports automatic lemmatisation and POS tagging in multiple languages. This is done using Tree Tagger. If your language is not listed, select 'Other'; in this case, automatic lemmatisation and POS tagging will be disabled.
6. [Optional: You can review/change the import options by left-clicking on a bar with three triangles (▲▲▲). In most cases, you can use the default options.]

7. Left-click 'Import!' to import your corpus into #LancsBox. By default, #LancsBox automatically adds POS tags to the corpus.

2.3 Supported file formats

#LancsBox supports different file formats (.txt, .xml, .doc, .docx, .pdf, .odt, .xls, .xlsx, .zip and many others) of corpus files. #LancsBox automatically extracts and processes text available in corpus files. For wordlists, #LancsBox assumes the comma-delimited file format (.csv).

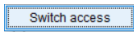
1. Corpus formats: .txt, .xml, .doc, .docx, .pdf, .odt, .xls, .xlsx, .zip – full list: Apache Tika.
2. Wordlist format: csv (see example below).

```
Corpus: BNC| Language: English| 4055 files| 96996843 tokens| 662414 types| 716618 lemmas|
"Type","Frequency: 01 - Freq","Dispersion: 01_CV"
"the","6054524.000000","0.286889"
"of","3049295.000000","0.400166"
"and","2622080.000000","0.263099"
"to","2599355.000000","0.223254"
"a","2168976.000000","0.221813"
"in","1945319.000000","0.333547"
```

2.4 Download #LancsBox corpora and wordlists

#LancsBox allows you to work with existing corpora that are freely distributed with #LancsBox under a specific license. Two modes for corpus sharing are available: i) open access and ii) restricted access. We are constantly adding more corpora to this list.

1. In the corpora tab, left-click on 'Corpus' or 'Word List' under 'Download'.
2. This will open a window where you can select corpora or wordlists distributed with #LancsBox. By left-clicking on a corpus, you will be shown additional information about the corpus or wordlist, including the language, date, text type, license etc.
3. Review and agree with the corpus license.
4. Left-click 'Download' to download the selected corpus or wordlist.
5. Left-click 'Import!' to import your corpus into #LancsBox. By default, #LancsBox automatically adds POS tags to the corpus.

▶ **Note:** To switch between open and restricted access corpora, use the 'Switch access' button in the bottom left corner (). Restricted access corpora are distributed as encrypted and have several display and usage restrictions. For example, they cannot be displayed in the Text tool or saved to the local computer.

2.5 Working with corpora and wordlists

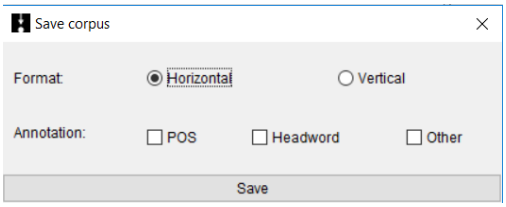
All corpora and wordlists that have been imported into #LancsBox are displayed in the bottom panel on the 'Corpora' tab. This panel allows reviewing corpora, previewing files and fast reloading of corpora and wordlists when #LancsBox is closed and re-opened.

1. If you have imported a corpus ( ) or wordlist ( ) it will appear in the bottom panel, alongside any other corpora or wordlist you have already imported. These can be removed by left-clicking 'delete'. In the bottom-right section, you can view the corpus structure: the individual text files that the corpus is composed of.
2. In the bottom panel (bottom left window), the default corpus can also be specified. The default corpus is a corpus that #LancsBox offers as a default choice in the individual modules. The default corpus can be specified by left-double-clicking on the name of the corpus; a filled rectangle ( ) will appear next to the name of the default corpus.
3. If #LancsBox is closed, the corpora and wordlists will remain imported but will be unloaded. To activate (reload) the corpora or wordlists for use, left-double-click on the corpora or wordlists.
4. You can also preview the files by right-clicking on them. They will appear in the Text tool (see Section 8). The list of files (including the info about their size) can also be copied (Ctrl/Coomand+C) and pasted (Ctrl/Command+V) into a spreadsheet or text document.
5. Corpora are now ready to be analysed using five modules: KWIC, Whelk, GraphColl, Words and Text. Wordlists can be used in the Words tool.

2.6 Saving corpora

#LancsBox saves corpora in the horizontal or the vertical format.

1. Right-click on the corpus which you wish to save.
2. Select appropriate options.



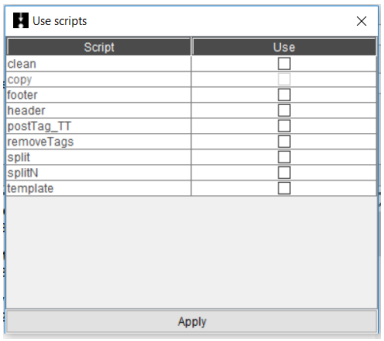
3. Click 'Save'.

2.7 Pre-processing of corpora (Advanced users).

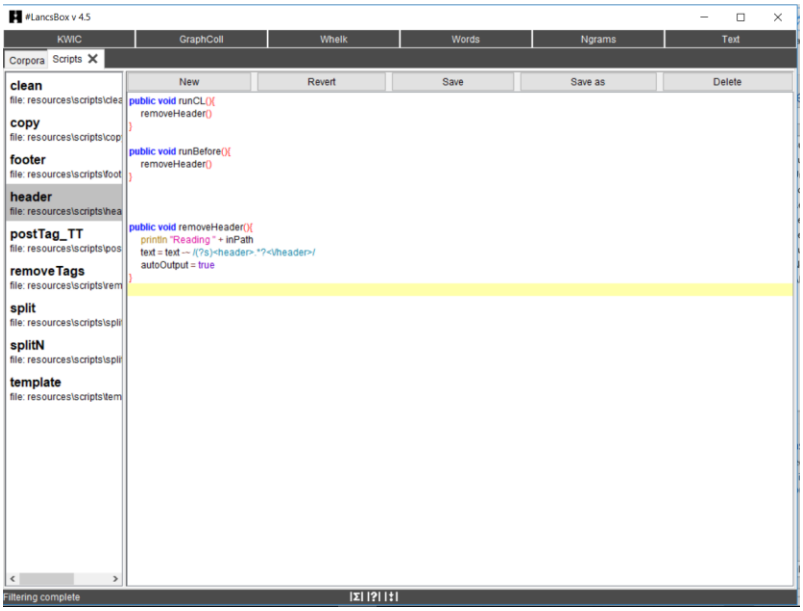
#LancsBox allows pre-processing data as part of the import procedure. This is set up in the 'Import options' under 'Pre-processing'. Data can be modified in different ways using a variety of Groovy scripts, which are fully customisable.

1. Under 'Pre-processing' three options are available:

2. ‘Download’ allows the user to download scripts and their newest versions available from the #LancsBox website.
3. ‘Use’ displays a list of currently available scripts and a checkbox next to each script for the user to indicate, which scripts will be used in the pre-processing stage.



4. ‘Edit’ displays the scripts in a script editor, which allows modifying existing scripts and creating new scripts.



5. The structure of a script is as follows. More information about the Groovy scripting language can be found at http://groovy-lang.org.

Script	Comments

public void runCL(){
println "Ran on the command line."
}
 	
public void runBefore(){
println "Ran as a pre-process script."
}
	Scripts run via the command line.
	
	Scripts run when the files are being loaded. This allows splitting files, deleting or changing texts or structuring elements e.g. xml tags.

public void runAfter(Token token){
println "Ran after the tagging step."
}
 	
public void removeHeader(){
println "Reading " + inPath
text = text -~ /(?s)<header>.*?<\/header>/
autoOutput = true
}
	Scripts run after part-of-speech tagging. This allows modifying the output of the Tree Tagger, e.g. correcting tagging errors.
	
	An example of a simple script deleting header indicated by <header></header> tags in text.

▶ Did you know?

The Brown corpus and the LOB (Lancaster-Oslo/Bergen) corpus are one of the first modern corpora stored and processed on computers. Each consists of one million running words (tokens), a size that was very ambitious at the time of their compilation. Brown was compiled in the 1960s by Henry Kučera and W. Nelson Francis at Brown University (US). It was originally stored and processed on IBM punch cards. In the early 1970s, a British counterpart to the Brown corpus was compiled as a collaboration between Lancaster University (UK) and two Norwegian universities: Oslo and Bergen. The project was initiated by Geoffrey Leech from Lancaster University.

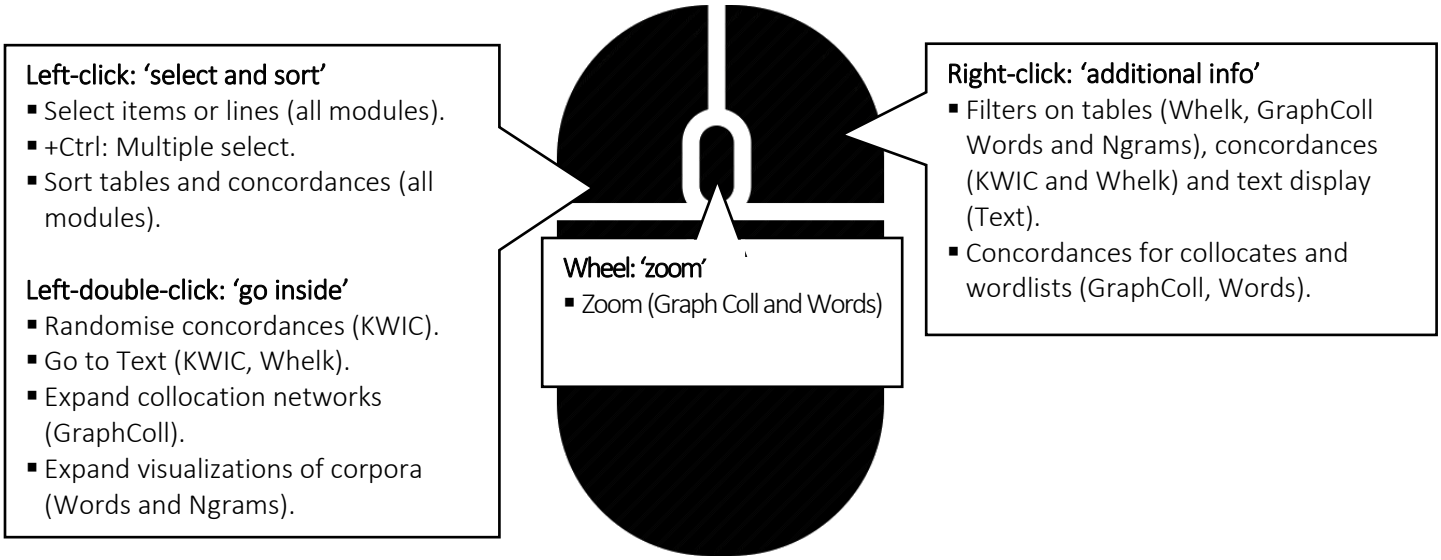
3 Key functionalities

This section reviews key functionalities of #LancsBox that are common to multiple #LancsBox modules.

3.1 Mouse clicks

#LancsBox doesn't use drop-down menus. Instead, all commands are literally just one mouse click away.

Hover with the mouse pointer for tooltips (brief contextual explanation of key functionalities/terms) to appear.



▶ **Note:** Mac users need to review their specific setup of the mouse clicks. By default, right-click is defined as Control + click. Alternatively, a standard two-button mouse with a wheel can be connected to a Mac machine.

3.2 Shortcut Keys

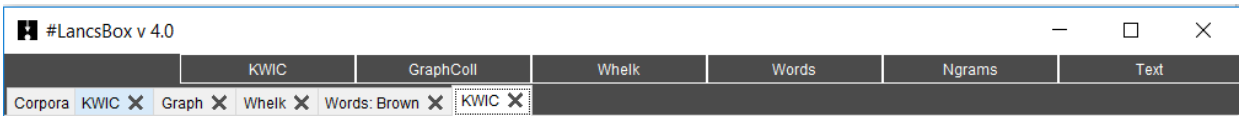
#LancsBox allows changing the size of the text for easy readability. This works both in graphs and tables.

Make all text bigger	Ctrl and +
Make all text smaller	Ctrl and -

3.3 Tools and Tabs

#LancsBox supports multiple simultaneous analyses and multiple corpora. #LancsBox has five main modules (tools): KWIC, Whelk, GraphColl, Words and Text. Each tool can be called multiple times on separate tabs. The modules in #LancsBox are interconnected: they can be launched as pop-ups inside a module.

1. The figure below show the top bar in #LancsBox with buttons for individual modules and multiple tabs open.



2. The modules in #LancsBox have the following functionalities:

   KWIC produces concordances.
   Whelk shows distribution of the search term in corpus files.
   GraphColl identifies and visualizes collocations.
   Words produces wordlists and identifies and visualizes keywords.
   Ngrams produces lists of ngrams and identifies and visualizes key ngrams.
   Text displays a full context of a search term.

3.4 Split screen

#LancsBox supports split-screen comparisons that allow displaying two separate analyses, one in the top and one in the bottom panel.

3. To use split screen, left-click on a bar with three triangles: ▲▲▲. This brings up the bottom panel.
4. To activate the bottom (or the top) panel in the split-screen view, left-click on the panel. An active panel is indicated by a light blue border ( ).
5. To close the split-screen view, left-click on the bar with three triangles: ▼▼▼. This will hide the bottom panel but will not clear the results, so the bottom panel can be brought back later, if needed.

3.5 Saving results

#LancsBox supports easy saving of results. It saves concordances, wordlists, tables and graphics.

1. To save the results that #LancsBox produces, left-click on the save icon ( ) in the top right-hand corner.
2. Select the location where you wish to save the results.
3. Click 'Save'.

3.6 Copy/pasting selected results

#LancsBox supports easy copy/pasting of selected results.

1. Select results which you wish to copy/paste by left-clicking on them; the results will be highlighted. To select discontinuous results, hold down Ctrl while selecting. To select all results, press Ctrl + A [Mac: Command + A].

Index	File	Left	Node	Right
1	A_Press_rep	and Juliet" was the irresponsibility of young	love	pushed into tragedy by Shakespeare." Othello" is
2	A_Press_rep	a cultivated, brave man who comes to	love	too late, and does not know what
3	A_Press_rep	not to know what to do with	love."	Zeffirelli does not mention the colour of
4	A_Press_rep	Logue writes fierce, noisy poems about war,	love,	and Logue. Son of a Southampton civil
5	A_Press_rep	go up in flames one day. In	love,	he wrote:—" I can not see Smiles
6	C_Press_rev	Byron's leaving him, the scandal of his	love	affair with his half-sister, Augusta Leigh, the
7	C_Press_revi	him to a point that looks like	love,	had fanned the enthusiasm which had sent

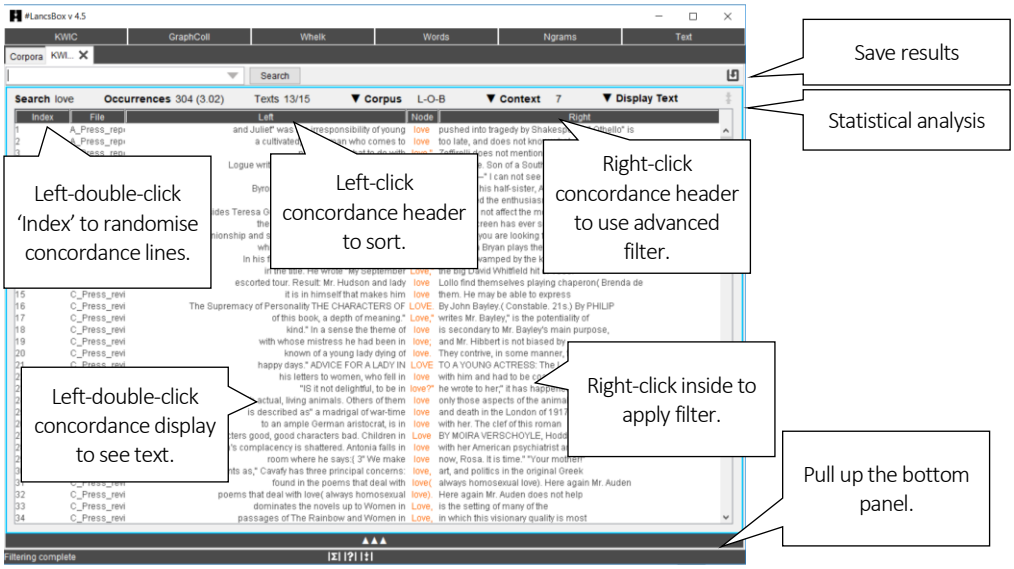
2. Press Ctrl + C [Mac: Command + C].
3. In the new location (e.g. text file, spreadsheet) press Ctrl + V [Mac: Command + V].

4 KWIC tool (key word in context)

The KWIC tool generates a list of all instances of a search term in a corpus in the form of a concordance. It can be used, for example, to:

- Find the frequency of a word or phrase in a corpus.
- Find frequencies of different word classes such as nouns, verbs, adjectives.
- Find complex linguistic structures such as the passives, split infinitives etc. using 'smart searches'.
- Sort, filter and randomise concordance lines.
- Perform statistical analysis comparing the use of a search term in two corpora.

4.1 Visual summary of KWIC tab



Simple search

You can:
Search for a word or phrase.
Search for number ranges, e.g. >1930&<=1945
Use * wildcards, e.g. new*
Use case sensitive regular expressions, e.g. /[abc].*/
Use case insensitive regular expressions, e.g. /dog|cat/i
Search for punctuation, e.g. /.*\./p
Use 'smart searches', e.g. PASSIVES, NOUNS

Advanced search

You can:
Search at different levels of annotation.
Combine search terms at various levels.
Use regular expressions, e.g. /N.*/
Define batch searches.

4.2 Searching and displaying results

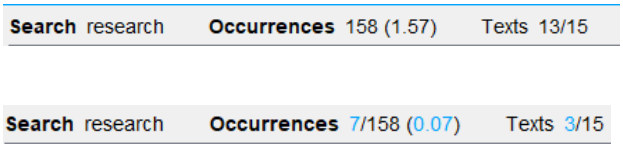
#LancsBox supports powerful searching of corpora. The search box can be used for simple as well as advanced searches at different levels of annotation.

1. Simple searches: type in the word or phrase of interest in the search box in the top left-hand corner and left-click 'Search'.
2. Advanced searches: click on the triangle inside the search box (▼) to activate advanced searches at different levels of corpus annotation. You can type search terms as separate constraints into one or more advanced search boxes. For example, the following advanced search is a search for the lemma 'go'.

Text level empty –> no constraint.

Headword is go.
AND
POS is any verbal use.

3. A concordance is generated. The search term, called the 'node', is positioned in the centre and highlighted (orange colour), with words displayed to the left and right of it.
4. KWIC displays basic information about the frequency of the search term and its distribution in texts; the second example shows an application of a filter (see Section 4.4):



Read: The search term 'research' occurs 158 times in the corpus with the relative frequency 1.57 per 10k words in 13 out of 15 texts.

Read: When a filter is applied (indicated by blue colour), the search term 'research' occurs 7 times out of 158 in the corpus with the relative frequency 0.07 per 10k words in 3 out of 15 texts.

4.3 Settings and full text pop-up

KWIC settings include Corpus, Context and Display options. KWIC also allows full-text pop-ups.

1. Corpus: this setting changes the corpus which is being searched. Note that different corpora can be searched in the top and bottom panel in split-screen view.
2. Context: this setting changes the number of words that are displayed in the concordance to the left and to the right of the node.
3. Display: this setting changes the display type. The 'Plain text' default can be changed to 'Text with POS', 'Lemmatized text' and 'All annotation'. The example below demonstrates these four display formats:

Plain text:	The new life looks promising for Mr. Noyce.
Text with POS:	The_DT new_JJ life_NN looks_VVZ promising_JJ for_IN Mr._NP Noyce._NP
Lemmatized text:	the_DT new_JJ life_NN look_VVZ promising_JJ for_IN Mr_NP Noyce_NP
All annotation:	[The{the}_DT] [new{new}_JJ] [life{life}_NN] [looks{look}_VVZ] [promising{promising}_JJ] [for{for}_IN] [Mr.{Mr}_NP] [Noyce.{Noyce}_NP]

4. Full text pop-up: Double left-click on a concordance line to display the entire text with the appropriate line highlighted.

4.4 Sorting, randomising and filtering

KWIC concordance can be sorted alphabetically, randomised and filtered.

1. Alphabetical sorting: Left-click the concordance header (any column) to sort the column alphabetically in the A-Z (ascending) order; click again to re-sort alphabetically in the Z-A (descending) order. The sorting is indicated by arrows: A-Z (▲) and Z-A (▼).
2. Randomising: Left-double-click the header of the 'Index' column to randomise the concordance lines. Randomisation is indicated by the tilde sign (~).
3. Simple filtering: Right-click anywhere inside the concordance to activate the simple filter on that column. Input a word or phrase or a regular expression enclosed in forward slashes (/ /) and click 'Apply'. Filtering is indicated by light blue colour of the filtered text. The filter also updates the results (Occurrences and Texts) in the top display panel (see Section 4.2, point 4).
4. Advanced filtering: Right-click any part of the concordance header to activate the advanced filter. Select an exact column or position for filtering (see below), enter value and click 'Add' and 'Apply'. Filtering is indicated by light blue colour on text and in the results display panel (Occurrences and Texts).
   An example of positions for advanced filtering:

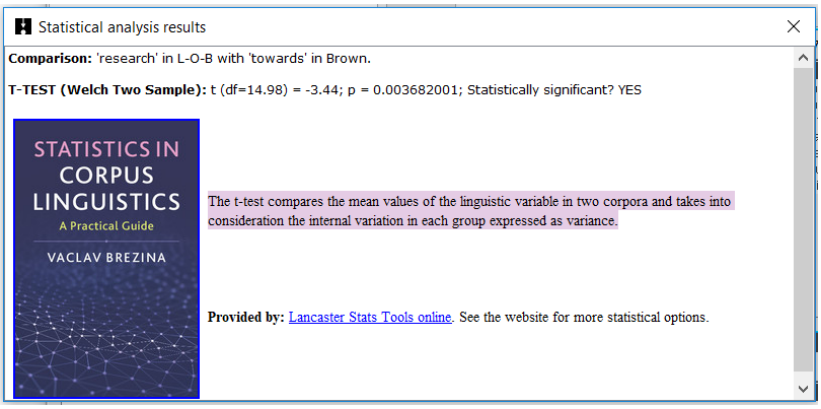
.

L5	L4	L3	L2	L1	Node	R1	R2	R3	R4	R5
is	Mr.	Robert	Weaver	of	New	York.	One	of	his	tasks

4.5 Statistical analysis

KWIC connects to Lancaster Stats Tools online to perform statistical analysis of the data in split panels.

When search results appear in both the top and the bottom panel in split-screen, these can be compared by clicking on the statistical analysis button ( ). The tool automatically connects to Lancaster Stats Tools online (Brezina 2018) and performs the t-test. The results are reported as follows:



▶ Did you know?

In 1992, when reviewing the state of the art in corpus linguistics, Leech (1992) considers a concordance program "[t]he simplest and the most widely-used tool for corpus-based research" (p. 114). 25 years later, a concordance program such as KWIC still belongs to the essential toolkit of a corpus linguist. The simple and direct access to data that a concordance program facilitates combined with more sophisticated functions such as sorting, filtering and randomising provides a powerful analytical technique.

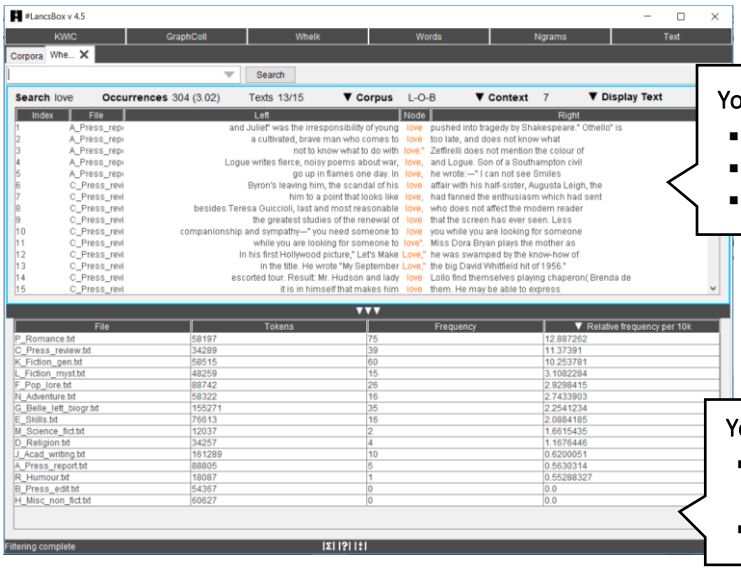
Leech, G. (1992). Corpora and theories of linguistic performance. In: *Directions in corpus linguistics*, 105-122.

5 Whelk tool

The Whelk tool provides information about how the search term is distributed across corpus files.
It can be used, for example, to:

- Find absolute and relative frequencies of the search term in corpus files.
- Filter the results according to different criteria.
- Sort files according to absolute and relative frequencies of the search term.

5.1 Visual summary of Whelk tab



Top panel: Searching corpora

You can:

- Search, sort and filter.
- Use simple and advanced searching functionality.
- Use 'smart' searches.

Bottom panel: Displaying distribution

You can:

- View the distribution of the search term in individual files.
- Sort, filter and copy/paste.

5.2 Top panel: KWIC

The top panel in Whelk has the same powerful search, sort and filter functionalities as the KWIC tool (see Section 4). It is directly connected to the bottom panel: any update in the top panel is immediately reflected in the bottom panel.

5.3 Bottom panel: Frequency distribution

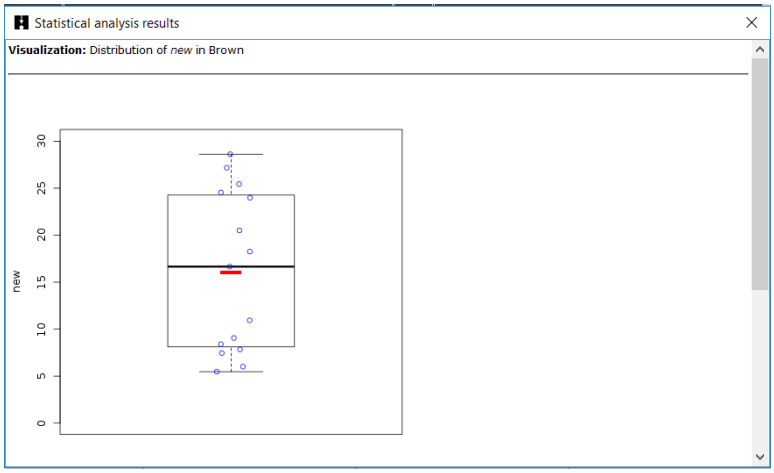
The bottom panel in Whelk provides detailed information about the distribution of the search term.

1. 'File' column lists the name of the individual files in the corpus.
2. 'Tokens' column provides the information about the size of each file in running words (tokens).
3. 'Frequency' column provides absolute frequencies of the search term i.e. refers to how many instances of the search term there are in each file.
4. 'Relative frequency per 10k' provides relative frequency normalised to the basis of 10,000 tokens; this value is comparable across files and corpora.

5.4 Statistical analysis

Whelk connects to Lancaster Stats Tools online to perform statistical analysis of the data.

When search results appear, these can be visualised using a boxplot by clicking on the statistical analysis button ( ). The tool automatically connects to Lancaster Stats Tools online (Brezina 2018) and displays the result:



▶ Did you know?

The Whelk tool (both the name and the functionality) is inspired by Kilgarriff's (1997: 138ff) notion of the 'whelks problem'. Imagine, says Kilgarriff, that you have a corpus which includes one text (a book) about whelks – small snail-like sea creatures ( ). In this text, the word *whelks* will appear many times and hence will appear as a frequent word in the entire corpus, although its use is limited to one specific context. To overcome the problem and present more accurate information about word distribution, the Whelk tool shows the frequency distribution of search terms in individual corpus files.

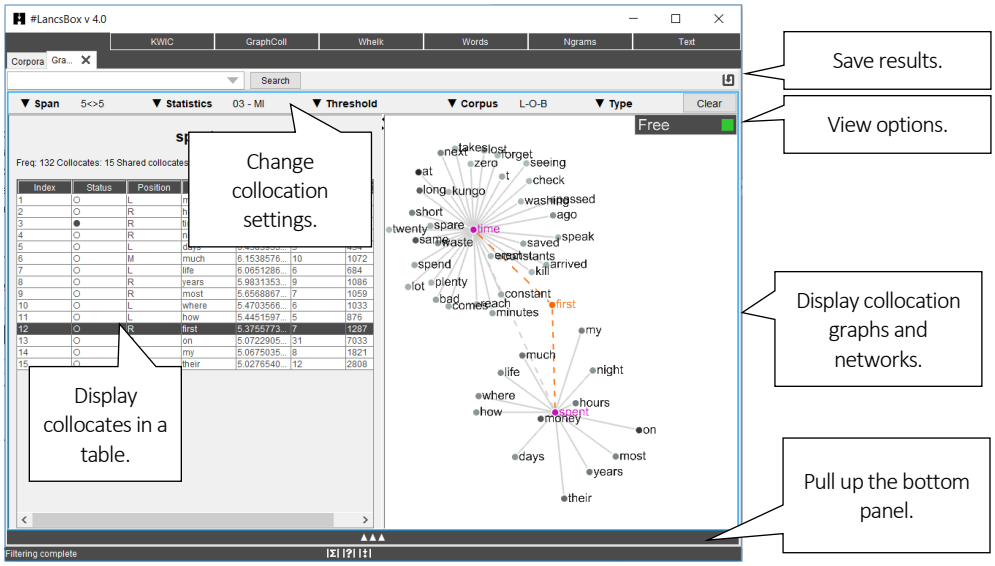
6 GraphColl

The GraphColl tool identifies collocations and displays them in a table and as a collocation graph or network.

It can be used, for example, to:

- Find the collocates of a word or phrase.
- Find colligations (co-occurrence of grammatical categories).
- Visualise collocations and colligations.
- Identify shared collocates of words or phrases.
- Summarise discourse in terms of its 'aboutness'.

6.1 Visual summary of GraphColl tab



6.2 Producing a collocation graph

GraphColl produces collocations graphs on the fly. After selecting the appropriate settings you can start searching for the node and its collocates.

1. Select the appropriate settings for the collocation search:
   i) Span: how many words to the left (L) and to the right (R) of the node (search term) are being considered when searching for collocates [default: 5L, 5R].
   ii) Statistics: the association measure used to compute the strength of collocation [default: frequency – no association measure is preferred because the choice depends on the research question].

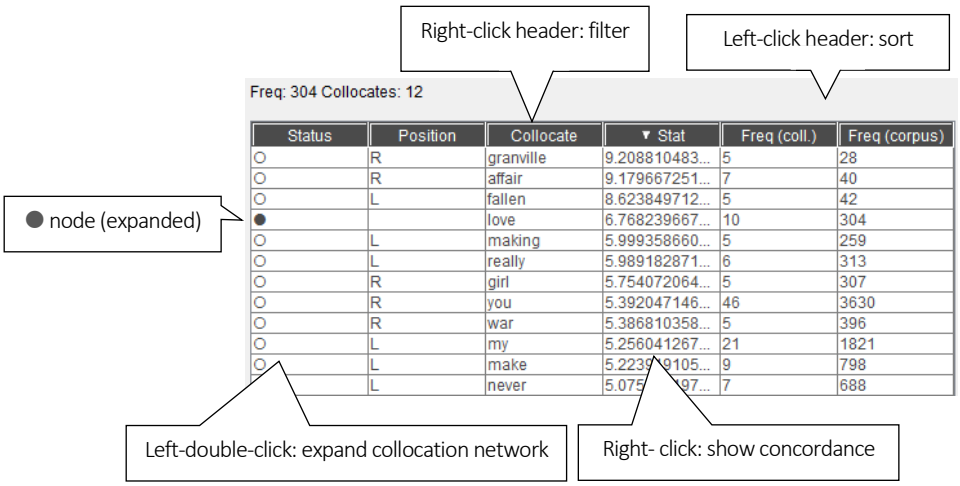
iii) Threshold: The minimum frequency and statistics cut-off values for an item (word, lemma, POS) to be considered a collocate.
iv) Corpus: The corpus that is being searched.
v) Unit: The unit (type, lemma, part of speech [POS] tag) used for collocates.

2. Type the search term into the search box (top left) and left-click 'Search'.
3. This will produce a colocation table (left) and a collocation graph (right).

6.3 Reading collocation table

A collocation table is a traditional way of displaying collocates. In GraphColl, the table shows the following pieces of information for each collocate: i) status, ii) position, iii) stat, iv) collocation frequency and v) frequency of the collocate anywhere in the corpus. By default, the table is sorted according to the selected collocation statistic (largest-smallest).

1. The following is a visual description of the collocation table.



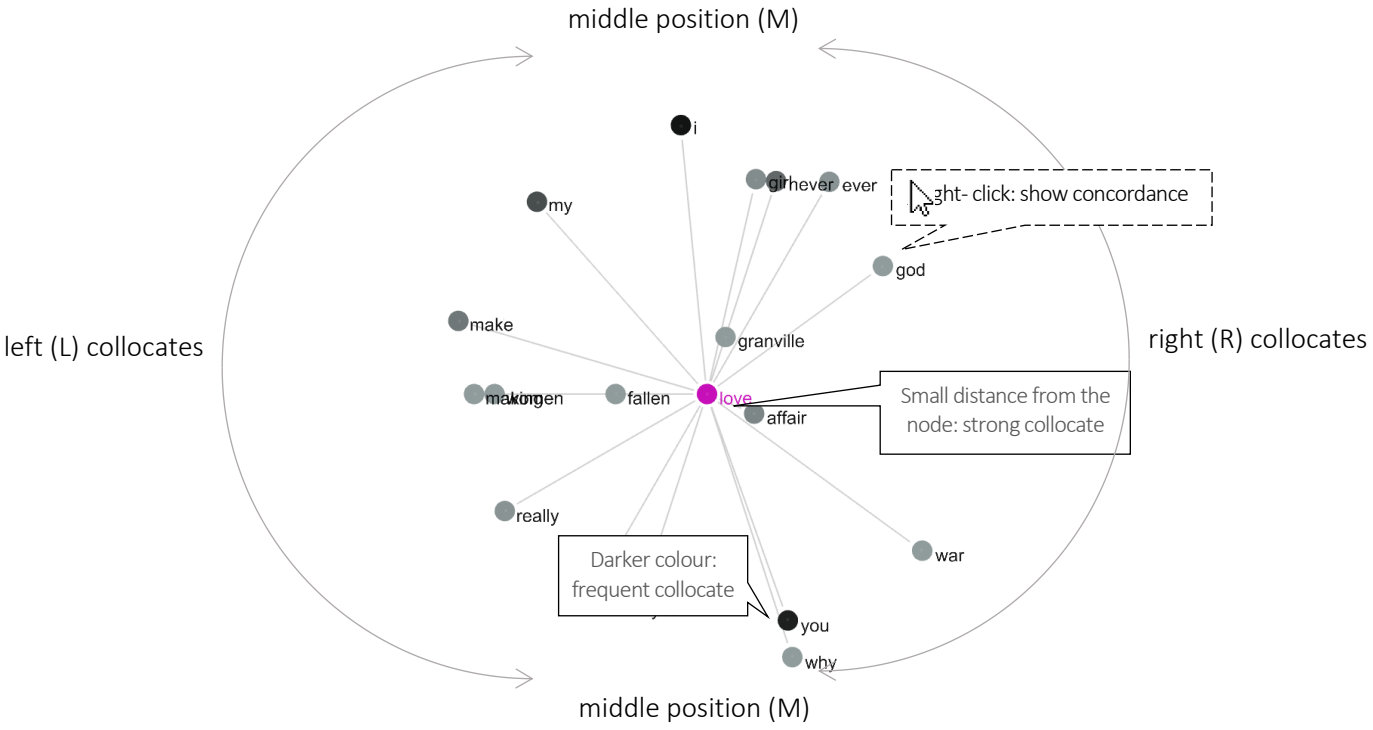
Status	Position	Collocate	▼ Stat	Freq (coll.)	Freq (corpus)
○	R	granville	9.208810483...	5	28
○	R	affair	9.179667251...	7	40
○	L	fallen	8.623849712...	5	42
●		love	6.768239667...	10	304
○	L	making	5.999358660...	5	259
○	L	really	5.989182871...	6	313
○	R	girl	5.754072064...	5	307
○	R	you	5.392047146...	46	3630
○	R	war	5.386810358...	5	396
○	L	my	5.256041267...	21	1821
○	L	make	5.223[illegible]9105...	9	798
	L	never	5.075[illegible]97...	7	688

2. The meaning of the individual columns is:
   i) Status: shows whether the collocate has been expanded; ○ indicates a non-expanded collocate, while ● indicates expanded collocate (node) in a collocation network.
   ii) Position: shows textual position of the collocate, which can be either left (L) of the node, right (R) of the node or middle (M), i.e. with equal frequency L and R.
   iii) Collocate: shows the collocate in question.
   iv) Stat: displays the value of the selected association measure.
   v) Freq (coll): displays the frequency of the collocation (combination of node + collocate).
   vi) Freq (corpus): displays the frequency of the collocate anywhere in the corpus.

6.4 Reading collocation graph

The graph displays three dimensions: i) strength of collocation, ii) collocation frequency and iii) position of collocates. To find out more about a collocate, right-click on it to obtain concordance lines (KWIC), in which the collocates co-occurs with the node.

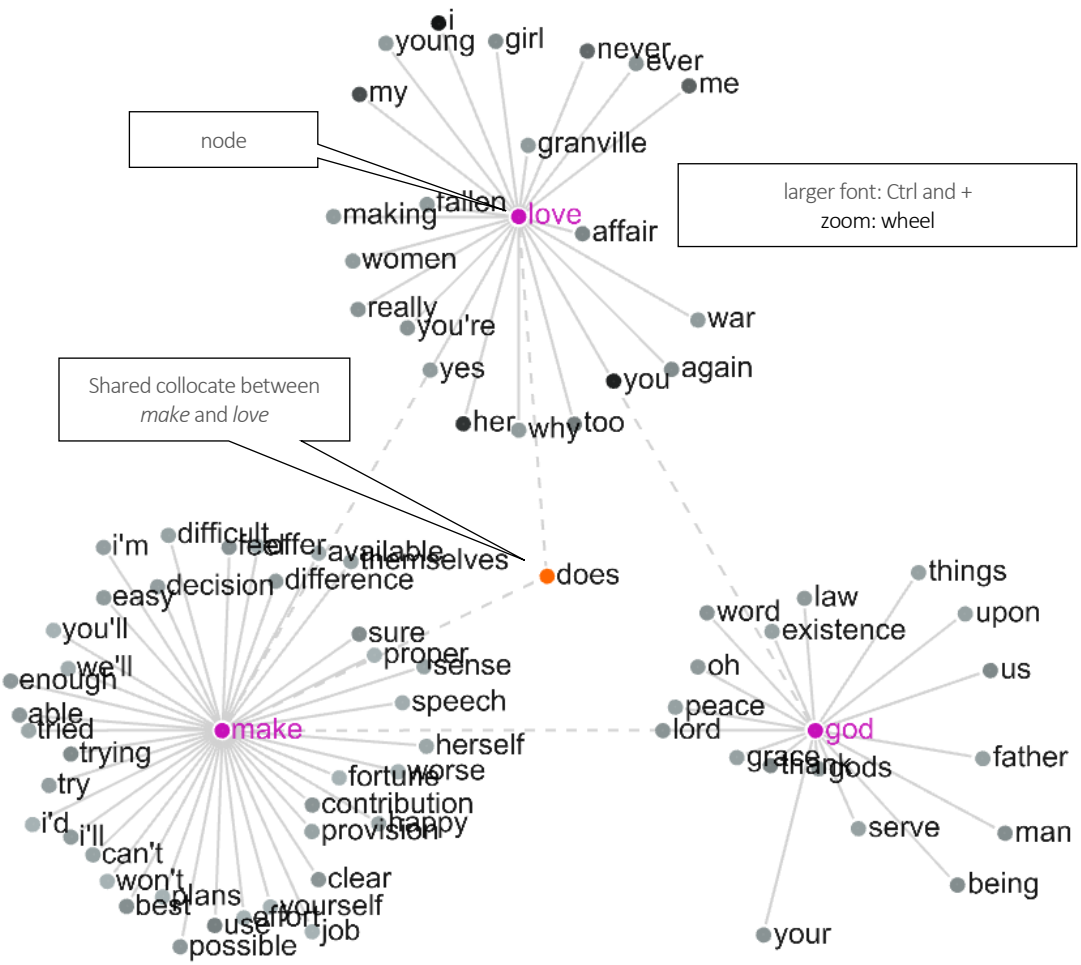
1. Strength: The strength of collocation as measured by the association measure is indicated by the distance (length of line) between the node and the collocates. The closer the collocate is to the node, the stronger the association between the node and the collocate ('magnet effect').
2. Frequency: Collocation frequency is indicated by the intensity of the colour of the collocate. The darker the shade of colour, the more frequent the collocation is.
3. Position: The position of collocates around the node in the graph reflects the exact position of the collocates in text: some collocates appear (predominantly) to the left of the node, others to the right; others still appear sometimes left and sometimes right (middle position in the graph). For the ease of display (if multiple collocates appear in a similar position and hence overlap), the tool allows 'spreading out' collocates evenly around the node. This is done by clicking on the 'Spread out' button (top right). When this is done, the collocates are dispersed evenly around the node with a 'L' or 'R' index displayed above the collocate circle indicating their original position to the left and to the right respectively.



6.5 Extending graph to a collocation network

A collocation network is an extended collocation graph that shows i) shared collocates and ii) cross-associations between several nodes.

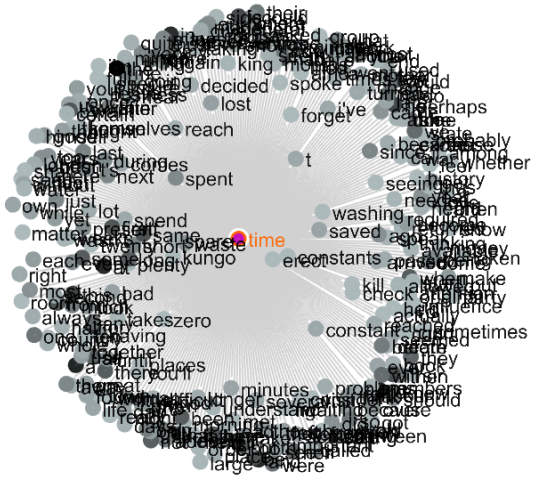
1. To expand a simple collocation graph (see above) into a collocation network, either search for more nodes or left-double-click on a collocate in either the table or the graph.
2. A collocation network displays nodes with unique collocates (outer rim of the graph) and shared collocates (middle of the graph). The links between nodes and shared collocates are indicated by a dash-dot line (–·–·–).



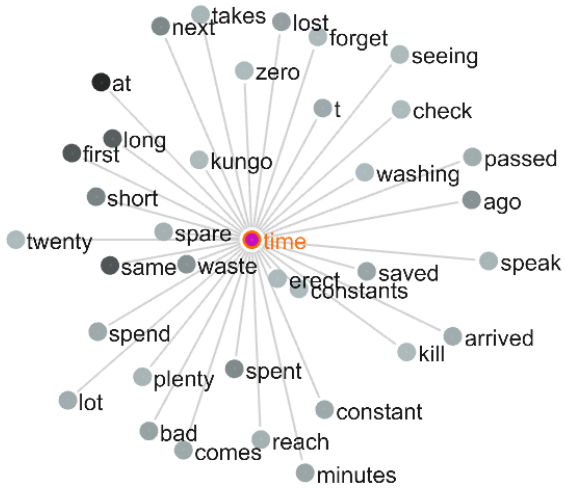
6.6 Problems with graphs: overpopulated graphs

If a collocation graph or network includes too many nodes and collocates, it becomes hard to interpret. We call this type of graph/network an overpopulated graph/network. The solution is either to change the graph's settings making the threshold values more restrictive (see Section 6.2) or filtering some of the results based on a clearly specified criterion (e.g. function words, top n words).

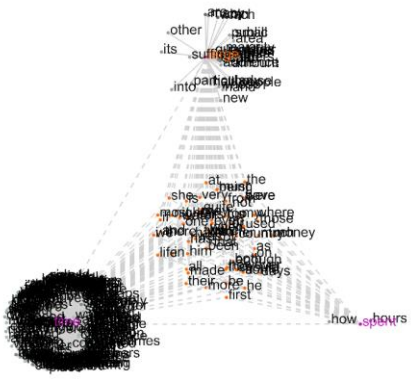
The following figure shows an overpopulated graph on the left and a graph that is more easily interpretable on the right. Note the difference in settings recorded in CPN (see Section 6.7)

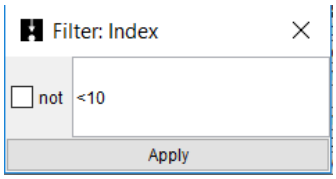


Collocation graph of 'time' 3a-MI(3), L5-R5, C5-NC5

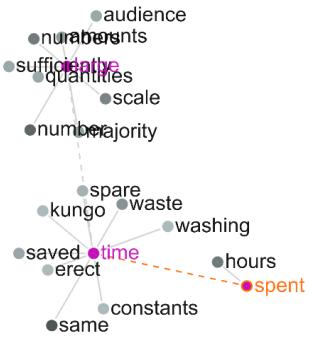


Collocation graph of 'time' 3a-MI(5), L5-R5, C5-NC5





To view n words in the whole graph, apply a filter (right-click) to the first column (Index).



6.7 Reporting collocates: CPN

It is important to realise that there is no one definite sets of collocates: different statistical procedures and threshold values highlight different sets of collocates. We therefore need to report the statistical choices involved in the identification of collocations using standard notation called Collocation Parameters Notation (CPN). When saving the results, GraphColl saves the settings in the form of CPN.

Brezina et al. (2015) propose CPN as a specific notation to be used for accurate description of collocation procedure and replication of the results. The following parameters are reported.

Statistic ID	Statistic name	Statistic cut-off value	L and R span	Minimum collocate freq. (C)	Minimum collocation freq. (NC)	Filter
4b	MI2	3	L5-R5	5	1	function words removed
4b-MI2(3), L5-R5, C5-NC1; function words removed						

▶ Did you know?

The name GraphColl is an acronym for *graph*ical *coll*ocations tool. GraphColl was the first module in #LancsBox (v.1.0) with the other tools being added at a later stage. Graphical display of collocations and collocation networks is inspired by the work of Phillips (1985), who demonstrated the concept of lexical networks (Phillip's term for 'collocation networks') with small specialised corpora. GraphColl takes this notion further, offering different statistical choices and producing collocation networks on the fly with both small and large corpora.

Phillips, M. (1985). *Aspects of text structure: An investigation of the lexical organisation of text*. Amsterdam: North-Holland.

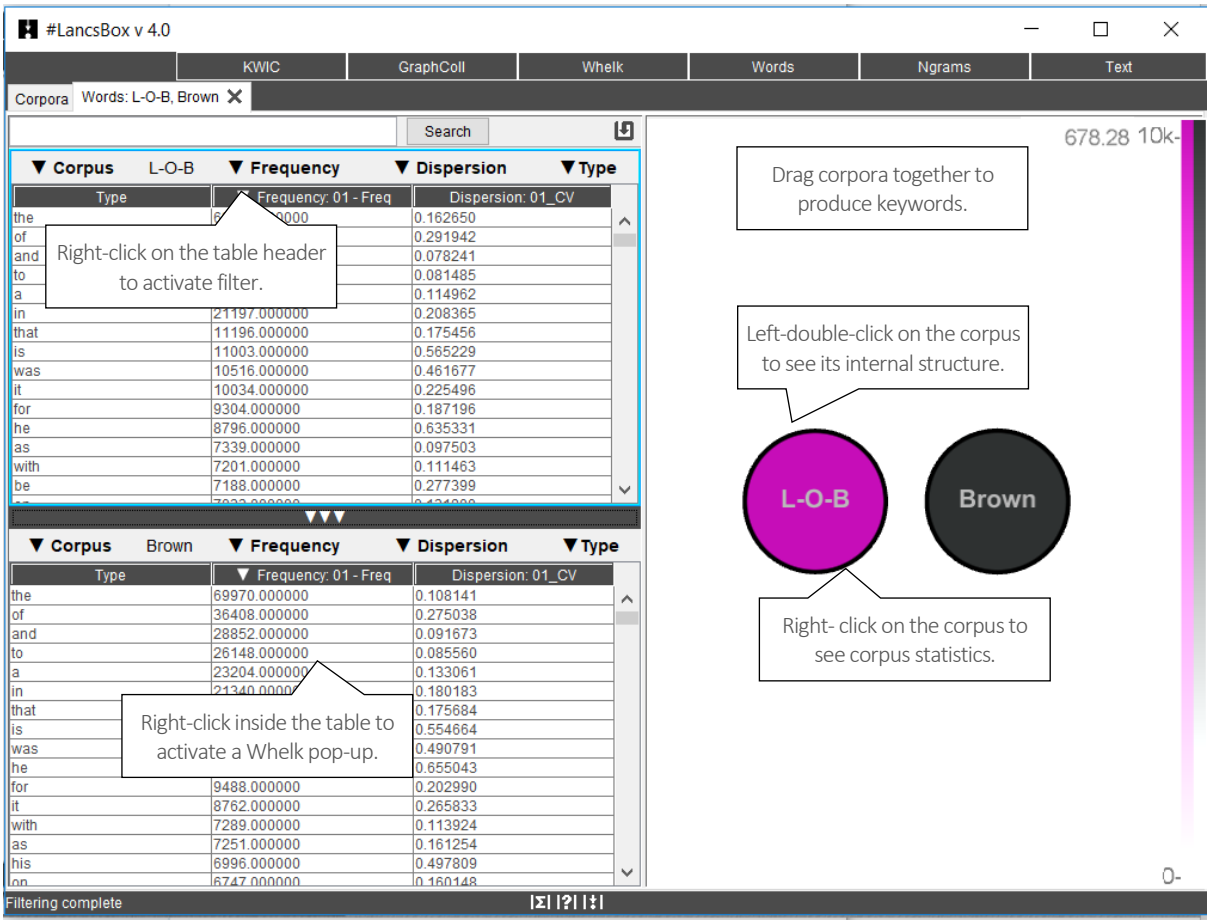
7 Words tool

The Words tool allows in-depth analysis of frequencies of types, lemmas and POS categories as well as comparison of corpora using the keywords technique.

It can be used, for example, to:

- Compute frequency and dispersion measures for types, lemmas and POS tags.
- Visualize frequency and dispersion in corpora.
- Compare corpora using the keyword technique.
- Visualize keywords.

7.1 Visual summary



Left: Creating frequency lists, computing dispersion and keywords.

Right: Visualizing frequencies, dispersions and keywords.

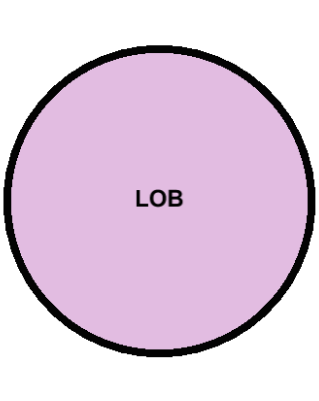
7.2 Producing frequency list

On start, Words produces a frequency list (table) based on the default corpus (see Section 2.5, point 2) and default settings. These settings can be changed and a different frequency list is produced.

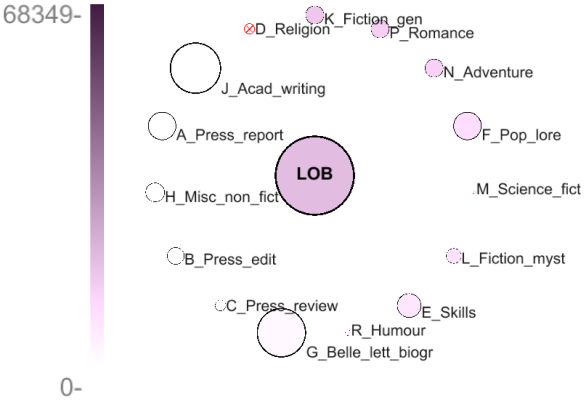
1. The following are the settings for frequency lists:
   i) Corpus: The corpus that is being used.
   ii) Frequency: Absolute or relative frequency [default: absolute frequency].
   iii) Dispersion: The dispersion statistic [default: coefficient of variation (CV)].
   iv) Unit: The unit used in the frequency list (type, lemma or part of speech tag).
2. Changing any of these settings triggers re-computing of the frequency list.
3. Frequency lists can be searched using the search box (top left).
4. Frequency lists can be sorted by left-clicking on the header.
5. Frequency lists can be filtered by right-clicking on the header and applying a filter.
6. Two different frequency lists can be computed in the split-screen view, which is triggered by left-clicking on a bar with three triangles: ▲▲▲. This brings up the bottom panel.

7.3 Visualizing frequency and dispersion

The Words module displays corpora and corpus files (when a corpus is left-double-clicked). It visualises frequency and dispersion of words using intensity of colour and position of individual files displayed as circles; the size of the circle indicates the relative size of the corpus/file.



Display of frequency in the whole corpus on the scale of 0 - 68,349 (most frequent item).



Display of frequency per file (when corpus is left-double-clicked).

1. To visualize frequency of an item in the table, left-click on the item in the frequency table. The shade of the colour of the corpus will change according to the frequency value of this item. The scale on the right offers a reference point for interpretation.
2. To visualize dispersion of an item in the table, left-double-click on the corpus (large circle). The corpus will expand to display individual files (small circles) of which the corpus consists. The size of each circles is proportional to the size of the corpus subpart. The shade of the colour of the

small circles will change according to the frequency value of the item in the frequency list. Crossed-out (⊗) circles indicate that the item does not occur in the given corpus file. In addition, the corpus files are ordered according to the relative frequency of the item with the file with the largest relative frequency of the item appearing at the 12-oclock position ( ) and the other files ordered clockwise according to decreasing relative frequency of the item ( ).

7.4 Producing keywords

The Words module computes a comparison of frequencies between two corpora/wordlists using a selected statistical measure. It identifies and visualizes positive keywords, negative keywords and lockwords.

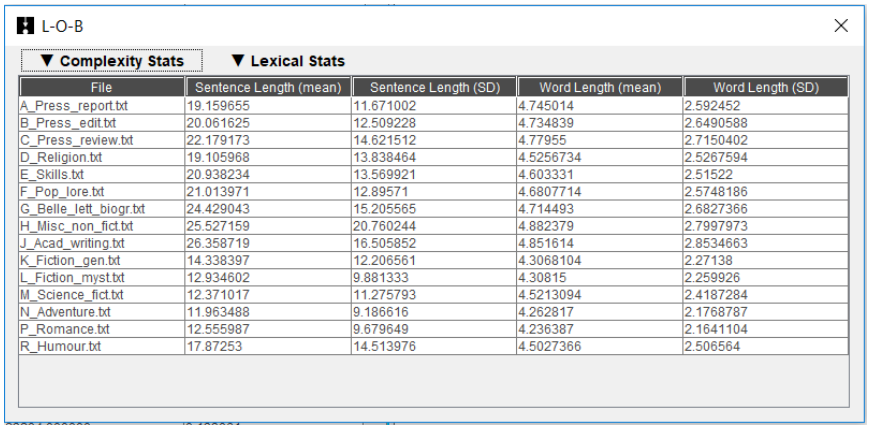
1. Left-click on ▲▲▲ to bring up the bottom panel.
2. In the bottom panel, select a comparison (reference) corpus, while in the top panel keep your corpus of interest.
3. In the visualisation panel (right), drag the circles that represent the two corpora together . Alternatively, press the space bar.
4. The resulting table will display frequency and dispersion info about the two corpora as well as the keyword statistic; the graphics will identify top 10 positive keywords, top 10 negative keywords and top 10 lockwords.
5. In the settings, you can change the i) keyword statistic and ii) threshold.
   Keyword statistic: This is a measure that compares two frequency lists [default: simple maths with constant k = 100].
   Threshold: Threshold values for the identification of positive keywords, negative keywords and (by implication) lockwords.

7.5 Producing corpus statistics

The Words module computes essential corpus statistics: i) Complexity stats and ii) Lexical stats

1. Right-click on corpus .
2. In the pop-up table toggle between Complexity stats and Lexical stats.

Mean sentence length and Standard deviation (SD)

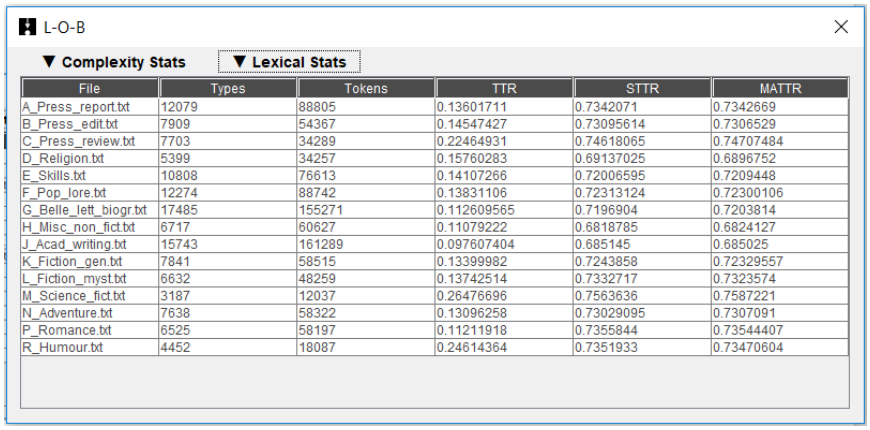


L-O-B

▼ Complexity Stats ▼ Lexical Stats

File	Sentence Length (mean)	Sentence Length (SD)	Word Length (mean)	Word Length (SD)
A_Press_report.txt	19.159655	11.671002	4.745014	2.592452
B_Press_edit.txt	20.061625	12.509228	4.734839	2.6490588
C_Press_review.txt	22.179173	14.621512	4.77955	2.7150402
D_Religion.txt	19.105968	13.838464	4.5256734	2.5267594
E_Skills.txt	20.938234	13.569921	4.603331	2.51522
F_Pop_lore.txt	21.013971	12.89571	4.6807714	2.5748186
G_Belle_lett_biogr.txt	24.429043	15.205565	4.714493	2.6827366
H_Misc_non_fict.txt	25.527159	20.760244	4.882379	2.7997973
J_Acad_writing.txt	26.358719	16.505852	4.851614	2.8534663
K_Fiction_gen.txt	14.338397	12.206561	4.3068104	2.27138
L_Fiction_myst.txt	12.934602	9.881333	4.30815	2.259926
M_Science_fict.txt	12.371017	11.275793	4.5213094	2.4187284
N_Adventure.txt	11.963488	9.186616	4.262817	2.1768787
P_Romance.txt	12.555987	9.679649	4.236387	2.1641104
R_Humour.txt	17.87253	14.513976	4.5027366	2.506564

Type-token ratio (TTR), Standardised type-token ratio (STTR), Moving average type-token ratio (MATTR)



L-O-B

▼ Complexity Stats ▼ Lexical Stats

File	Types	Tokens	TTR	STTR	MATTR
A_Press_report.txt	12079	88805	0.13601711	0.7342071	0.7342669
B_Press_edit.txt	7909	54367	0.14547427	0.73095614	0.7306529
C_Press_review.txt	7703	34289	0.22464931	0.74618065	0.74707484
D_Religion.txt	5399	34257	0.15760283	0.69137025	0.6896752
E_Skills.txt	10808	76613	0.14107266	0.72006595	0.7209448
F_Pop_lore.txt	12274	88742	0.13831106	0.72313124	0.72300106
G_Belle_lett_biogr.txt	17485	155271	0.112609565	0.7196904	0.7203814
H_Misc_non_fict.txt	6717	60627	0.11079222	0.6818785	0.6824127
J_Acad_writing.txt	15743	161289	0.097607404	0.685145	0.685025
K_Fiction_gen.txt	7841	58515	0.13399982	0.7243858	0.72329557
L_Fiction_myst.txt	6632	48259	0.13742514	0.7332717	0.7323574
M_Science_fict.txt	3187	12037	0.26476696	0.7563636	0.7587221
N_Adventure.txt	7638	58322	0.13096258	0.73029095	0.7307091
P_Romance.txt	6525	58197	0.11211918	0.7355844	0.73544407
R_Humour.txt	4452	18087	0.24614364	0.7351933	0.73470604

▶ Did you know?

The statistical technique of keyword analysis was originally developed by Mike Scott (1997) and it was implemented in WordSmith Tools. It relied on corpus comparison using the chi-squared test or the log-likelihood test. As Kilgarriff pointed out, the chi-squared test and the log-likelihood test are not entirely appropriate for this type of comparison. Kilgarriff's solution implemented in Sketch Engine was to compare corpora using a 'simple maths' procedure, a simple ratio between relative frequencies of words in the two corpora we compare. In addition to 'simple maths', #LancsBox offers also other types of solutions for corpus comparison.

Scott, M. (1997). PC analysis of key words—and key key words. System, 25(2), 233-245.
Kilgarriff, A. (2009, July). Simple maths for keywords. In *Proceedings of the Corpus Linguistics Conference. Liverpool, UK*.

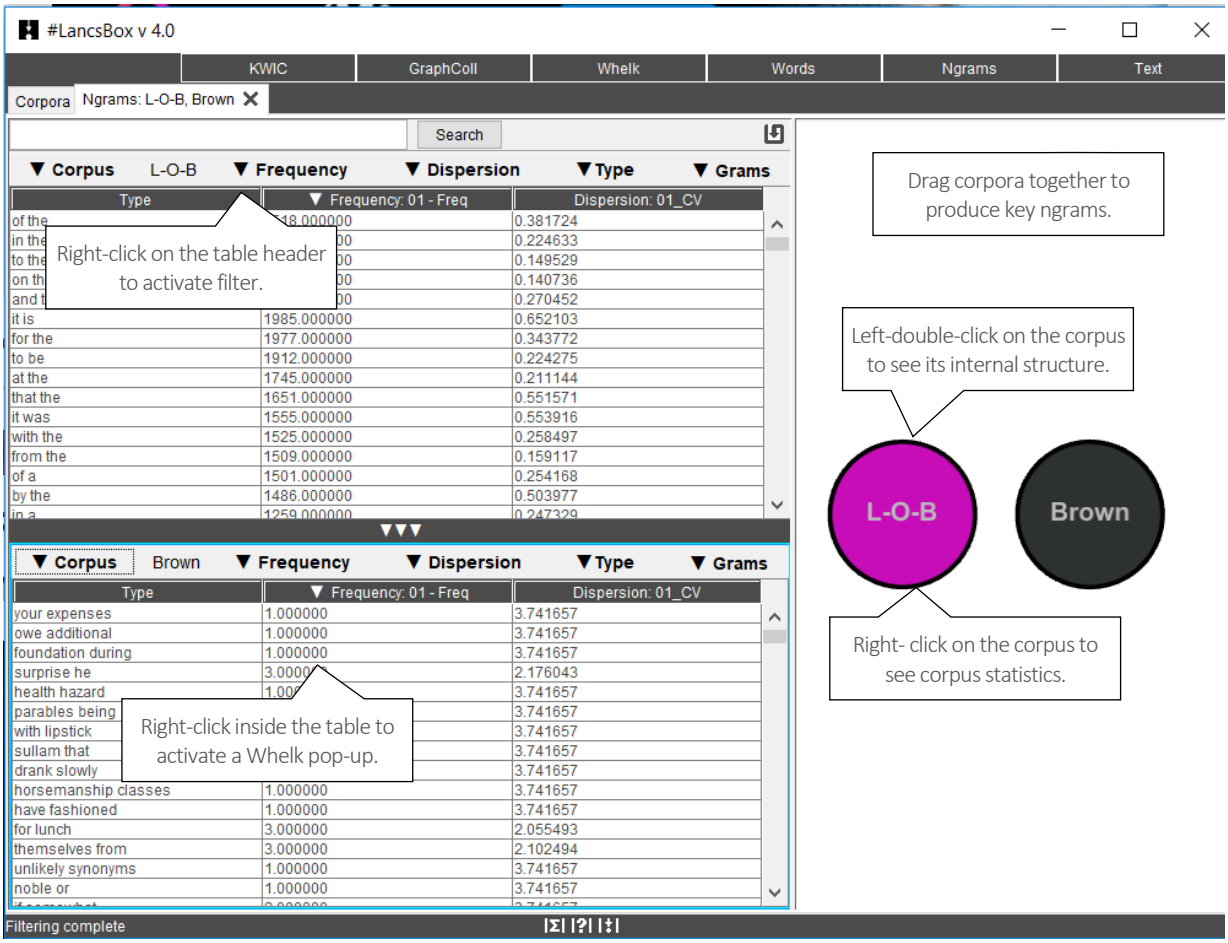
8 Ngram tool

The Ngram tool allows in-depth analysis of frequencies of n-grams (bigrams, trigrams etc.), which could be defined as contiguous combinations types, lemmas and POS. The tool also produces key ngrams by comparing two corpora using a technique similar to keywords.

It can be used, for example, to:

- Identify n-grams, lexical bundles and p-frames (also skip grams)
- Compute frequency and dispersion measures for ngram types, lemmas and POS tags.
- Visualize frequency and dispersion of ngrams in corpora.
- Compare ngrams in two corpora using the keyword technique.
- Visualize key ngrams.

8.1 Visual summary



Left: Creating frequency lists, computing dispersion and key ngrams.

Right: Visualizing frequencies, dispersions and key ngrams.

▶ Did you know?

Multi-word expressions are extremely important when describing language. There are different terms to describe multi-word expressions such as collocations (Brezina et al. 2015; Gablasova et al. 2017), n-grams, lexical bundles and p-frames. While collocations, which are identified in the GraphColl module, typically represent non-contiguous expressions, the n-gram type multi-word expressions represent contiguous lexico-grammatical patterns. They are defined as follows.

- n-gram: a sequence of n types, lemmas, POS from a text or corpus.
- lexical bundle: an ngram with certain frequency and distributional (dispersion) properties, e.g. relative freq. 10 per million and range > 5.
- p-frame (also skip gram): an n-gram that allows for variability at one or more positions such as *it would be * to.*

All these types of multi-word expressions can be identified using the Ngram tool in #LancsBox.

Brezina, V., McEnery, T. & Wattam, S. (2015). Collocations in context: A new perspective on collocation networks. *International Journal of Corpus Linguistics*, 20(2), 139-173.

Gablasova, D., Brezina, V., & McEnery, T. (2017). Collocations in corpus-based language learning research: Identifying, comparing, and interpreting the evidence. *Language Learning*, *67 (S1),* 155–179.

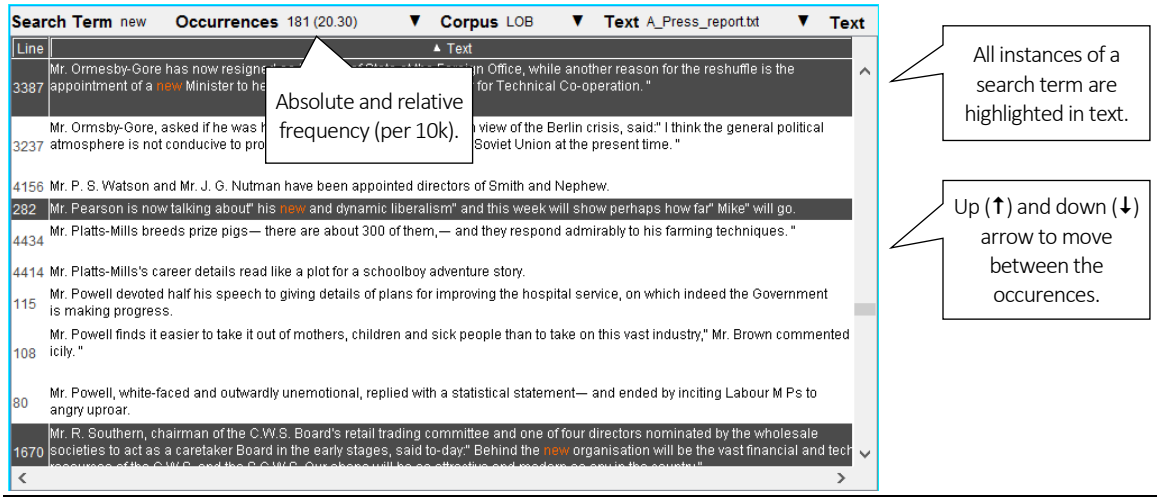
9 Text

The Text tool enables an in-depth insight into the context in which a word or phrase is used.

It can be used, for example, to:

- View a search term in full context.
- Preview a text.
- Preview a corpus as a run-on text.
- Check different levels of annotation of a text/corpus.

9.1 Visual summary



9.2 Searching in Text

Texts and corpora can be searched easily using a simple search box.

1. Type the search term into the search box (top left). Left-click 'Search'.
2. This will highlight all lines in the text where the search term appears in dark grey with the search term itself in red. To move between the highlighted lines up (↑) and down (↓) arrows can be used.
3. Frequency information (both an absolute and relative frequency per 10,000 tokens) will appear under 'Occurrences'.
4. A single line can be highlighted by left-clicking on the line. To highlight multiple lines, Ctrl (Command) + Left-click the desired lines.
5. Highlighted lines can be copied (Ctrl/Command+C) and pasted (Ctrl/Command+V) into a text editor.

9.3 Settings

The following settings are used in Text: i) Corpus, ii) Text and iii) Display.

1. Corpus: this setting allows changing the corpus which is being displayed and searched. Note that different corpora can be searched in the top and the bottom panel in the split-screen view.
2. Text: this setting allows changing the text that is being displayed and searched.
3. Display: this setting allows changing the display format. The 'Plain text' default can be changed to 'Text with POS', 'Lemmatized text' and 'All annotation'.

10 Searching in #LancsBox

Throughout the tool, #LancsBox offers powerful searches at different levels of corpus annotation using i) simple searches, ii) wildcard searches, iii) smart searches, iv) regex searches and v) batch searches.

1. Simple searches are literal searches for a particular word (*new*) or phrase (*New York Times*). Simple searches are case insensitive; this means that *new, New, NEW, NeW* etc. will return the same set of results.
2. Wildcard searches are searches including one of three special characters *, <, > and =.

Special character	Meaning	Example of use
*	0 or more characters	new* [*new, news, newly, newspaper…*]
	any word [with space]	new *[*new car, New York, new ideas…*]
>	larger than	
<	smaller than	
=	equals [combined with < and >]	

3. Smart searches are searches predefined in the tool to offer users easy access to complex searches; smart searches are unique to #LancsBox. These searches are used for searching for word classes (NOUNS, VERBS etc.), complex grammatical patterns (PASSIVES, SPLIT INFINITIVE etc.) and semantic categories (PLACE ADVERBS, HEDGES).
4. Regex searches are advanced searches that allow to search for any combination of characters. Any expression enclosed in forward slashes (//) is interpreted as regular expression. #LancsBox supports perl-compatible regular expressions.

Regex	Explanation	Regex	Explanation
Word	A string of characters (case sensitive)	a{3}	Exactly 3 of a
/word/i	A string of characters (case insensitive)	a{3,}	3 or more of a
/word\./p	Punctuation search: A string of characters followed by full stop (case sensitive)	a{3,6}	Between 3 and 6 of a
[abc]	A single character either a, b or c.	\d	Any digit
[^abc]	Any single character except: a, b, or c	\D	Any non-digit
[a-z]	Any single character in the range a-z	\w	Any word character (letter, number, underscore)
[a-zA-Z]	Any single character in the range a-z or A-Z	\W	Any non-word character
[0-9]	A single number in the range 0-9		
.	Any single character		
(a|b)	a or b		
a?	Zero or one of a		
a*	Zero or more of a		
a+	One or more of a		

5. Batch searches allow to search for multiple search terms recursively and saving the results automatically; #LancsBox supports both simple and complex batch searches. Batch searches can be used in KWIC, GraphColl and Whelk modules when the corpora are tagged. Here is how batch searches work.

a) Click on the down arrow in the search box to activate Advanced search options. The last option is a batch search. Click on 'Batch'.

b) Navigate to and load a text file with the appropriate search terms, one per line. Simple search terms include a list of word forms to be searched; complex search terms are defined via a combination of criteria such as word form, pos tag, headword etc… Consecutive criteria need to be present on the same line separated by tab (\t) in the following order: label – wordform – headword – pos – user tag. This is best achieved by creating the file with advanced batch search terms in Excel or Calc. Examples of simple and complex searches can be seen below.

Simple batch search: each search term on a separate line	Complex batch search: label – wordform – headword – pos – user tag (tab separated)
my cat go went	

	A	B	C	D	E
1	multiple-simple	/cat|dog|mouse/			
2	headword		go		
3	pos			N*	
4	user				Mytag
5	combibnation	/going|went/	go	/V.*/	Mytag

c) Once the file with search terms is loaded, click on the 'Search' button (Search) and navigate to the location where the results will be saved.

11 Statistics in #LancsBox

#LancsBox uses statistics for calculating measures of i) frequency, ii) dispersion, iii) keywords and iv) collocation. The equations of these measures can be reviewed and modified on the 'Stats' tab, which is called by clicking on the Σ button.

11.1 Frequency measures

1. absolute frequency = o11
2. relative frequency = (o11/r1) x 10,000

11.2 Dispersion measures

1. CV = SD/mean
2. SD = $\sqrt{\frac{\sum(x-mean)^2}{n}}$
3. Range = no of files where the search term occurs at least once
4. Range % = $\frac{\text{Range}}{\text{number of files}} \times 100$
5. D = $1 - \frac{\text{CV}}{\sqrt{\text{number of files}-1}}$
6. DP = $\frac{\text{Sum of absolute values of (observed}-\text{expected proportions)}}{2}$

11.3 Keyword measures

1. simple maths parameter = $\frac{\text{relative frequency of w in C + k}}{\text{relative frequency of w in R + k}}$
2. log likelihood $_{\text{short}}$ = $2 \times \left(O_{11} \times \log\frac{O_{11}}{E_{11}} + O_{21} \times \log\frac{O_{21}}{E_{21}}\right)$
3. % DIFF = $\frac{(\text{relative freq. in C}-\text{relative freq. in R}) \times 100}{\text{relative freq.in R}}$
4. Log Ratio = $\log 2 \left(\frac{\text{relative freq. in C}}{\text{relative freq. in R}}\right)$
5. Cohen's *d* = $\frac{\text{Mean}_{\text{in C}}-\text{Mean in R}}{\text{pooled SD}}$

11.4 Collocation measures

ID	Statistic	Equation	ID	Statistic	Equation
1	Freq. of co-occurrence	O_{11}	8	T-score	$\frac{O_{11} - E_{11}}{\sqrt{O_{11}}}$
2	MU	$\frac{O_{11}}{E_{11}}$	9	DICE	$\frac{2 \times O_{11}}{R_1 + C_1}$
3	MI (Mutual information)	$\log_2 \frac{O_{11}}{E_{11}}$	10	LOG DICE	$14 + \log_2 \frac{2 \times O_{11}}{R_1 + C_1}$
4	MI2	$\log_2 \frac{{O_{11}}^2}{E_{11}}$	11	LOG RATIO	$\log_2 \frac{O_{11} \times R_2}{O_{21} \times R_1}$
5	MI3	$\log_2 \frac{{O_{11}}^3}{E_{11}}$	12	MS (Minimum sensitivity)	$min\left(\frac{O_{11}}{C_1}, \frac{O_{11}}{R_1}\right)$
6	LL (Log likelihood)	$2 \times \left(\begin{matrix} O_{11} \times log\frac{O_{11}}{E_{11}} + O_{21} \times log\frac{O_{21}}{E_{21}} + \\ O_{12} \times log\frac{O_{12}}{E_{12}} + O_{22} \times log\frac{O_{22}}{E_{22}} \end{matrix} \right)$	13	DELTA P	$\frac{O_{11}}{R_1} - \frac{O_{21}}{R_2}; \frac{O_{11}}{C_1} - \frac{O_{12}}{C_2}$
7	Z-score[1]	$\frac{O_{11} - E_{11}}{\sqrt{E_{11}}}$	14	Cohen's *d*	$\frac{Mean_{in\ window} - Mean_{outside\ window}}{pooled\ SD}$

12 Glossary

Absolute (or raw) frequency – The simple frequency with which a search term occurs in a corpus or its part(s); a number of hits of a search term in a corpus.

Batch search – A batch search enables searching for multiple search terms recursively and saving the results automatically; #LancsBox supports both simple and complex (i.e. defined via a combination of criteria such as wordform, pos tag, headword etc.) searches.

Colligation – Systematic co-occurrence of grammatical categories (e.g. POS tags) in text identified statistically.

Collocate – A word that systematically occurs with the node (word or phrase of interest, search term).

Collocation – Systematic co-occurrence of words in text identified statistically.

Collocation graph is a visual display of the association between a node and its collocates. See GraphColl.

Collocation network is a visual display of complex associations (collocations) in language and discourse. It consists of multiple inter-connected collocation graphs. See GraphColl.

Concordance line – A single line in the KWIC display representing a node (search term) with the words before and after it (the right and left context).

Concordance is a typical form of display of examples of language use found in a corpus with the node (search term) centred in the middle and several words of context displayed left and right of the node. Concordance is sometimes also called a 'KWIC (display)'.

Corpus (pl. corpora) – A collection of language data that can be searched by a computer.

Dispersion – is the spread of values of a variable (e.g. relative frequencies of a search term) in a dataset (corpus). Dispersion is measured statistically using metrics such as standard deviation (*SD*), coefficient of variation (*CV*), range, Juilland's D, DP etc. See Words.

Frequency – The number of times a search term occurs in the corpus. A distinction is made between absolute (absolute number of hits) and relative frequency (proportional frequency per X number of tokens).

Frequency distribution – frequency distribution provides information about the frequencies of a word or phrase in different parts of the corpus. See Whelk.

GraphColl is a module n #LancsBox, which identifies collocations and builds collocation networks on the fly.

Import – In #LancsBox, processing of corpus data and making it available to all modules in the package.

KWIC is an abbreviation for 'keyword in context'. This is a typical form of display of examples found in a corpus with the node (word or phrase of interest) centred in the middle and several words of context displayed left and right of the node. KWIC is sometimes also called a 'concordance'. KWIC is also the name of a module in #LancsBox.

Left context – The words preceding a particular search term (node). Individual positions in the left-context are referred to as L1 (position immediately preceding), L2, L3 etc.

Lemma – All inflected forms belonging to one stem; in #LancsBox by default, a combination of a headword and a grammatical category (e.g. go + VERB). For example, a lemma 'go' includes the following word forms (types): 'go', 'goes', 'went', 'going' and 'gone'.

Lexical bundle – an n-gram with certain frequency and distributional (dispersion) properties, e.g. relative freq. 10 per million and range > 5.

Loaded – In #LancsBox, when a corpus is loaded it is available to be analysed. To re-load a corpus, double-left-click on the name of the corpus.

Module – A specific tool within #LancsBox offering particular analytical functionalities. #LancsBox includes five different modules: KWIC, Whelk, GraphColl, Words and Text.

N-gram – a sequence of n types, lemmas, POS from a text or corpus.

Node – The word, phrase or grammatical structure of interest. See Search term.

Part of speech (POS) – A grammatical category, a word class. Part-of-speech is usually assigned automatically using a process called part-of-speech tagging (see below). #LancsBox includes TreeTagger, which performs part-of-speech tagging for a range of languages.

Part-of-speech tagging (POS tagging) – A process of adding information about the grammatical category of each word in a text or corpus. For example, the following sentence was POS-tagged: Automatically_RB annotates_VBZ data_NNS for_IN part-of-speech_NN.

P-frame (also skip gram) – an n-gram that allows for variability at one or more positions such as it would be * to.

Regular expressions (regex) – A special meta-language that allows advanced users to search for any combination of strings. In #LancsBox, regex searches are enclosed in forward slashes e.g. /.*ions?/

Relative (or normalized) frequency (RF) is calculated as the proportion of the absolute frequency of a word we are interested in divided by the total number of words (tokens) in the corpus. This number is usually multiplied by an appropriate basis for normalization (e.g. 10,000).

Right context – The words following a particular search term (node). Individual positions in the right-context are referred to as R1 (position immediately following), R2, R3 etc.

Split screen – A comparison option in #LancsBox where the screen can be split into two panels; each panel can display a different type of analysis. #LancsBox allows second panel to be opened and minimised via left-clicking on three small triangles (▲▲▲/▼▼▼).

Tab – A further 'page' that can be opened in #LancsBox to run multiple analytical procedures simultaneously. Each module in #LancsBox can run on an unlimited number of tabs.

Tagging – The process of adding linguistic information to the words in a text or corpus, automatically or semi-automatically. See Part-of-speech tagging.

Text – A basic unit of a corpus; a corpus is a collection multiple texts. Text is also the name of a module in #LancsBox that displays and searches texts in corpora.

Threshold – Setting options in GraphColl and Words to display only relevant collocates or keywords respectively.

Token is a single occurrence of a word form in a text or corpus.

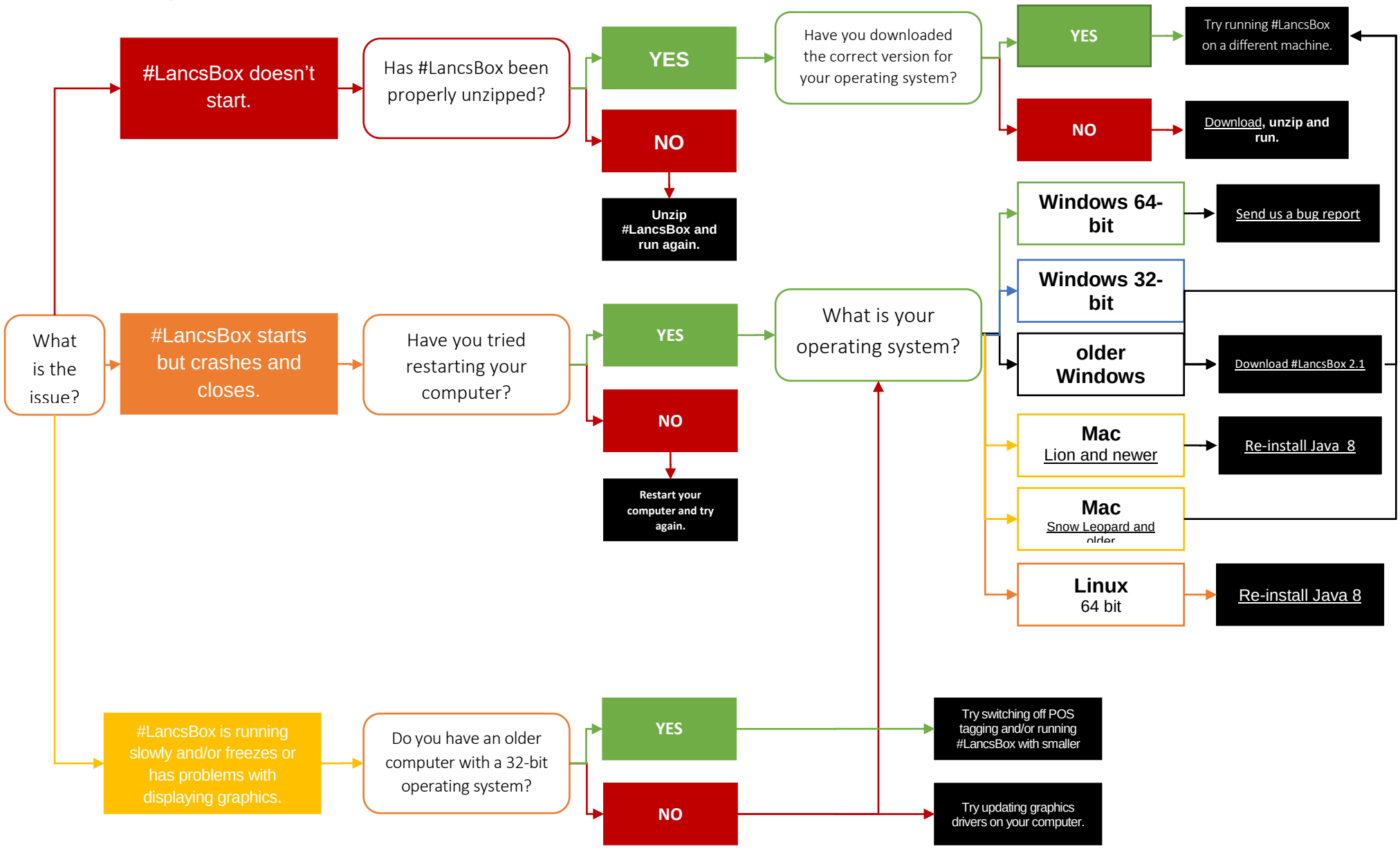
TreeTagger is a part-of-speech tagger developed by Helmut Schmid, which performs part-of-speech tagging for a range of languages.

Type is a unique word form in a text or corpus.

Whelk is a module in #LancsBox which provides information about how the search term is distributed across corpus files.

Words is a module in #LancsBox which allows in-depth analysis of frequencies of types, lemmas and POS categories as well as comparison of corpora using the keywords technique.

13 Troubleshooting



14 Messages.Properties

How to configure #LancsBox for advanced users

The Messages.properties file lets you customise #LancsBox. The things you change in here will change how the program operates and looks.

15 Making Changes

To change a setting in Messages.Properties: First, look for the setting you want. This will look something like: an.interesting.setting = value. We will call the first part (before the =) the key and the second part (after the =) the value.

Changing the value will change the setting. Each of the values has it's own type. This just means that when you change a colour it should be for another colour, not for a word. We will now introduce the value types used in Messages.properties.

- Path – This tells LancsBox where on your computer to look for something. This could be where to find an Icon, or where to find other information that LancsBox relies on. It will look like this: **/resources/path/to/a/file**. Any path starting with /resources refers to something within the resources folder.
- Integer (This just means a whole number)
  - As a number – Some settings want an actual number, like the default span for KWIC searches.
  - As a selector – Sometimes we use an integer to pick between several options. The options will be described in comments so you will know what your choices are.
  - As a true/false – Some settings turn things on or off in #LancsBox. If you want the setting on then the value should be **1**. **0** means turn it off.
- Colour – You can change many of the colours used in #LancsBox. These follow a particular format. We recommend using an online colour picker to find out what the value should be. It will begin with a **#**.
- Regular expression – These are like the regular expressions that you use within #LancsBox with two changes. Firstly, the //s are omitted, Secondly the options (like i) are in the Java format. These are in round brackets, have a question mark and precede the expression itself. like: **(?i) regular expression**
- Literal text – Whatever you type as the value here will be used directly. However, please note that the UI for #LancsBox uses an English font, this may limit the utility of these types of settings.
- Number Format – Java has it's own way of defining number formats. Changing these in Messages.Properties will change now #LancsBox displays numbers and can be useful in altering the number of decimal places in tables.

Message.properties file	Explanation

# This is the Messages.properties file 	
# tagger 	
tagger.dir = resources/tagger
tagger.langs = resources/tagger/models
	Tagger .dir and .langs both have path values and refer to parts of a Tree-Tagger installation. The root directory of the Tree-Tagger is defined by.dir and .langs is a folder containing the language specific files (.par extension).

```
# database
# < 1 - RAM database
#   0 - RAM database, basic persistence
#   1 - QuestDB (recommended) (64bit)
#   2 - MapDB (32bit)
#   3 - LMDB (64bit)
database.use = 1
database.dir = resources/corpora
database.cache.size = 2000000

#download locations
downloads.corpora       = resources/downloads/corpora
downloads.wordlists     = resources/downloads/wordlists



# language settings
langs.dir = resources/languages



# default tokenizer settings
defaults.punctuation
        =.,:;?!\u00e3\u0080\u0082\u00ef\u00bc\u008c\u00ef\u00bc\u009b\u00ef
\u00bc\u009a\u00ef\u00bc\u009f\u00ef\u00bc\u0081\u00e2\u0080\u009a\u00c
2\u00bf\u00c2\u00a1\u00e2\u0080\u00a6'\"\u00e2\u0080\u0098\u00e2\u0080\
u0099`\u00e2\u0080\u009c\u00e2\u0080\u009d\u00e2\u0080\u009e()<=>[]{}\u0
0e2\u0080\u00b9\u00e2\u0080\u00ba\u00e3\u0080\u008a\u00e3\u0080\u008b-
\u00e2\u0080\u0093\u00e2\u0080\u0094\u00e4\u00b8\u0080*
defaults.segmentation           =\\t\\n \\r
defaults.sentence_boundary      =(?s).*[\\.|!|\\?|。|？|！].*
```

Database

.dir lets you change where corpora will be stored and is a path value. The number of tokens held in RAM can be limited or expanded by changing .size, which is an integer value. A number of different databases can be used within #LancsBox and .use lets you change which one is being used. The integer value can be one of a number of options which are part of the comment above the setting. Note that you can't load a corpus using a database if it wasn't created by the same database.

Download locations

When you download wordlists and corpora they are saved in the downloads folder in resources prior to being imported into a corpus. Changing the path values of these settings lets you change where they get saved.

Language settings

The language-specific settings are stored in the languages folder in resources. Changing the .dir setting lets you change this location.

Default tokenizer settings

The tokenizer can be configured on the corpora panel in #LancsBox. The default values that appear on those boxes come from here. Making the change in Messages.Properties means that you only have to make the change once. Please note that .punctuation and .segmentation are literal values (which include an additional escape character - \) whereas .sentence_boundary is a regular expression. The sentence boundary is used for calculating average sentence length and similar metrics.

```
# Script directory
stats.dir          = resources/stats
stats.threshold    = resources/groovy/default_threshold.groovy
stats.dir.collocate = resources/stats/collocate
stats.dir.keyword.frequency   = resources/stats/keyword/frequency
stats.dir.keyword.dispersion   = resources/stats/keyword/dispersion
stats.dir.keyword.statistic   = resources/stats/keyword/statistic
shaders.dir          = resources/shaders

# Tool logo
icons.logo         = resources/images/logo.png

# Fonts
fonts.all.size                  = 12
fonts.table.size          = 12
fonts.2d.scale                  = 0.25
fonts.graph.size          = 84
fonts.graph.size.scale          = 0.125
fonts.keyword.size              = 84
fonts.keyword.size.scale= 0.4




# Select the fonts to use when there is no custom font installed.
# The custom font is the first .ttf file found in the resources/fonts folder.
# Java logical font options:
# 1 - Dialog
# 2 - DialogInput
# 3 - Monospaced
# 4 - Serif
# 5 - SansSerif
fonts.default.ui = 5
fonts.default.3d = 1
```

Script directory

#LancsBox calculates statistics using a number of external scripts, which you can also edit. Each of the groups of scrips lives in a different folder. These path values let you change where they are read from.

Tool logo

The path value can be changed to change the #LancsBox logo for another image.

Fonts

The font sizes used in #LancsBox can be altered here. The graph and keyword fonts have large sizes which use the .scale values to shrink them. This gives high resolution text at a good size. To increase the beauty of text in graphs and words tools make the appropriate .size values larger and the .scale values smaller.

Java uses what it calls logical fonts. The #LancsBox UI uses different logical fonts by default. You can change the default font options by changin the .ui and .3d integer values. A comment precedes the settings to inform you of the available options. The data font can be overwritten from this by placing a single .ttf file in the resouces/fonts folder.

```
# Misc
window.size.width = 1024
window.size.height = 768
slider.lock = 0
tokeniser.allowRtoL	=0
display.default.RtoL	=0
numbers.format.integer		=###,###,###,###,###
numbers.format.real		=####0.000000
numbers.format.real_short	=####0.00



# General program colours
colours.bar		=#4B4B4B
colours.highlight	=#00A4FF
colours.text_highlight	=#ff6600
colours.text	=#4B4B4B
colours.advanced_arrow		=#B3B1B0

# General UI paths and settings
icons.frame		= resources/images/icon.png
icons.tabs.attach	= resources/images/pin1.png
icons.tabs.close	= resources/images/cross.png
icons.generic.right_arrow	= resources/images/right-arrow.gif
icons.corpora		= resources/images/corpora.png
icons.save		= resources/images/save.png
icons.stats		= resources/images/stats.png
icons.about		= resources/images/about.png
icons.help		= resources/images/help.png
icons.kwic		= resources/images/kwic.png
```

Misc

.lock allows you to either lock or unlock (true/false) the slider which lets you resize the tables in graph and words.

Most right to left corpora are actually in left to right format in the files, but are displayed in reverse. If the actual data is stored as right to left (very unlikely) then .allowRtoL can be enabled and a new checkbox will appear in import options on the corpora tab. In the much more likely event that the data is left to right but should be displayed right to left then the default display direction of #LancsBox using the .RtoL setting. Both of these settings are also true/false values. The format of numbers in tables can be changed using the .integer, .real and .real_short settings.

General program colours

Some of the more widely used colours in #LancsBox can be changed here.

General UI paths and settings

The path values can be changed to load custom icons in the #LancsBox UI. The default message on the status bar can be changed by altering the .welcome_message value, which is a literal string.

```
icons.graph                    = resources/images/graph.png
icons.compare                  = resources/images/compare.png
icons.compare.disabled = resources/images/compareDisabled.png
statusbar.welcome_message   = Welcome to #LancsBox

# table icons
icons.sort.ascending           = resources/images/upArrow.png
icons.sort.descending          = resources/images/downArrow.png
icons.sort.ascending.filtered  = resources/images/upArrowSquare.png
icons.sort.descending.filtered = resources/images/downArrowSquare.png
icons.sort.filter          = resources/images/square.png
icons.sort.random              = resources/images/random.png



# The tooltips for various buttons
buttons.tooltip.corpora  = Corpora
buttons.tooltip.save           = Save
buttons.tooltip.graph          = Collocation graphs and networks tool
buttons.tooltip.kwic           = <html>Concordance tool</html>
buttons.tooltip.whelk          = Dispersion tool
buttons.tooltip.keywords       = Wordlists and keywords tool
buttons.tooltip.ngram          = N-Gram tool
buttons.tooltip.text           = Text tool
buttons.tooltip.help           = Help
buttons.tooltip.stats          = Statistics
buttons.tooltip.about          = About
buttons.popup.close      = Apply

# Generic button labels, reused throughout
buttons.generic.browse  = Load data
buttons.generic.delete  = Delete
buttons.generic.clear   = Clear
buttons.generic.run     = Run
```

Table Icons

Changing these path values lets you change the icons that appear in #LancsBox tables.

The tooltips for various buttons

The tooltips are customisable for the main buttons. These are the tool and status bar buttons that you first see when loading #LancsBox.

Generic button labels, reused throughout

The text of some buttons in the UI can be changed by altering these literal string values. This includes the apply button on some popups.

```
buttons.generic.new     = New
buttons.generic.load    = Load
buttons.generic.save    = Save
buttons.generic.close   = Close
#buttons.generic.stop   = Stop

# Load pane
labels.load.prompt_name = Name:
labels.load.corpus_name = Corpus
labels.load.case        = Clamp types to lowercase
labels.load.punctuation = Store punctuation
buttons.load.new        = Import!
buttons.load.reset      = Reset to defaults
icons.load.corpus           = resources/images/corpus.png
icons.load.wordlist         = resources/images/wordlist.png

# Stats pane text
tabs.name.stats             = Statistics
labels.stats.name       = Name:
buttons.stats.commit    = Save
buttons.stats.save      = Save as...
buttons.stats.load      = Open...
buttons.stats.remove    = Remove
buttons.stats.revert    = Revert

# n-gram settings
defaults.ngrams = 2


# keywords renderer
colours.keywords.corpus_name_dark     = #000000
colours.keywords.corpus_name_light    = #bababa
colours.keywords.text                 = #000000
```

Load pane

The main corpora pane uses some literal strings that can be changed here.

Stats pane text

The stats panel uses some literal strings that can be changed here.

N-gram settings

The n-grams tool defaults to being a bigram tool. This can be changed by changing this integer value.

Keywords renderer

The words / ngrams tool has a number of colours which can be changed. Those which have a corresponding .max value denote a colour range. The frequency colours will be interpolated using these ranges.

```
colours.keywords.target                 = #c60db8
colours.keywords.target.max             = #c60db8
colours.keywords.reference              = #2e3131
colours.keywords.reference.max                  = #2e3131
colours.keywords.highlight              = #ff6600
colours.keywords.table                  = #d1d1d1
colours.scroll                          = #5f5f5f
colours.no_scroll                       = #d1d1d1

# Graph pane text
buttons.graph.export    = Export
buttons.graph.export.dot = .dot File
buttons.graph.export.img = .png Image
buttons.graph.labels    = Labels
buttons.graph.run       = Search
buttons.graph.layout    = Layout
buttons.graph.kwic      = KWIC
buttons.graph.threshold  = Threshold
buttons.graph.stat      = Stat

# Graph Renderer
colours.graph.node                      =#c60db8
colours.graph.collocate_light           =#e6f7f9
colours.graph.collocate_dark            =#000000
colours.graph.highlight                 =#ff6600
colours.graph.edge              =#d1d1d1
colours.graph.text              =#000000
colours.graph.shared                            =#ff6600
colours.graph.shared_background                 =#583e82
renderer.screenshot.width                       =7680
renderer.screenshot.height                      =4320
renderer.default.sphere_size            =6
renderer.default.sphere_resolution      =50
```

Graph pane text

A number of string literals are given here for the GraphColl tool. The string literals can be changed here.

Graph renderer

The GraphColl tool has colours and colour ranges which can be changed. These are the colour values given here. Additionally the size of screenshots can be changed here (though they also apply to words) by changing the integer values of .width and .height. The number of sides a sphere has (all 3d tools) can be changed by altering the integer value of the .sphere_resolution setting. This can drastically speed up crowded graphs but not all numbers will work on all computers. The size of graph spheres can also be changed using .sphere_size. This gives you even greater control that just changing the font size.

```
colours.toggle.free             = #31c831
colours.toggle.hybrid           = #ffc200
colours.toggle.positional       = #ff0040
colours.toggle.word_class       = #cd00cd

# Whelk searches
whelk.window.span = 100


#KWIC pane colours
colours.kwic.node               =#ff6600
colours.kwic.highlight          =#00a4ff
colours.kwic.highlight_not      =#5e626b

# KWIC window size settings
kwic.left.min                  =3
kwic.left.def                 =5
kwic.left.max                  =20
kwic.right.min                 =3
kwic.right.def                 =5
kwic.right.max                  =20

# POS group colours
colours.group.1         = #0080ff
colours.group.2         = #ff0080
colours.group.3         = #00cd67
colours.group.4         = #ff6500
colours.group.5         = #cd00cd
colours.group.6         = #d5ff00
colours.group.7         = #a6a6a6
colours.group.8         = #00e6e6
colours.group.9         = #ff4dff
colours.group.10        = #006200
```

Whelk searches

STTR and MATTR searches can be performed in #LancsBox. These use a window size of a number of tokens. This number can be changed by setting the value of the .span setting to a different integer.

KWIC pane colours

The colours used in the KWIC tool can be changed by altering the colour values of these settings.

KWIC window size settings

The default span settings for KWIC searches can be set here. The integer values only define the defaults, you can still change them in the program.

POS group colours

The POS groups / aliases can be defined in the import options. The first ten of them will be assigned these colours when viewing lemma graphs in the word class mode.