AntConc – software overview

Anthony, L. (2020). AntConc (3.5.9) [Computer Software]. Tokyo, Japan: Waseda University. Available from https://www.laurenceanthony.net/software

1. AntConc can be downloaded from the website below, for the workshop we'll use version 3.5.9

https://laurenceanthony.net/software/antconc/

2. We'll begin by testing out the basic tools within AntConc, but first we need a sample dataset.

Go to:

https://tinyurl.com/CLWarsaw2024

And download the following:

COCA folder (this is a fragment of the Corpus of Contemporary American English)

antbnc_lemmas – this is a list that groups different word forms into lemmas

BNC wordlist – this is a frequency wordlist from the Brirish National Corpus

3. Concordancing – displaying "key words in context" (KWIC)

One of the basic tools of corpus linguistics is exploring linguistic items in context. A line of text that contains the linguistic item you're looking for is called a *concordance line*

Exercise 0
Test the search box. Type in any word.

Exercise 1
Consider the words: *mother, run, smart, quickly, in*
What are your expectations about the typical context in which these might appear?
How do your intuitions compare with the actual corpus results?

4. Collocate tool

Collocations are words that frequently appear near one another. In AntConc it's possible to specify the distance between the search term and its collocates.

Exercise 2
Choose 2-3 words from the following list (or come up with something of your own); note down your intuitions about possible collocates; verify your ideas

white, knife, day, time, happy, mother, smart, apple, student

dance, walk, eat, sleep, fight, love, talk, put, set, do

5. Word tool

The Word tool counts all the words in the corpus and orders the results in a list.

It's possible to lemmatize this list if needed.

Exercise 3
Before crating the list, what are your intuitions about the most frequent words in a corpus of American English?

6. Keyword tool

The AntConc keyword tool can tell you which words in your corpus are statistically more frequent than in a reference corpus. The tool is very useful for exploring thematic corpora, and general differences between datasets.

In the broadest sense a keyword is any word of interest to the researcher, and there are various ways of deciding what that means.

Exercise 4
Run the keyword tool on the American corpus with the BNC wordlist as reference. Comment on the results.

7 Advanced Querying

Discussion topic
So far we've queried specific words by entering the exact word forms. What are the possible limitations of this approach? What other things are worth looking for?

Wildcards
AntConc offers a number of "wildcards" in its search box

? – any one character
* – zero or more characters
+ – one or more characters
@ – zero or one word
# – any one word
| – 'OR' operator

Regular expressions (RegEx)

Regular expressions are special search queries commonly used in programming software. They can be used to find all expressions that match a specific pattern, which makes them useful in corpus linguistics as well. Below are some basic RegEx, along with examples. There are plenty of online guides for this as well:
https://www.regular-expressions.info
The implementation of RegEx differs from software to software. The examples below work best with AntConc. The changes mentioned by AntConc's creator are:
"With 'regex' option, each word-level regular expression needs to be separated by whitespace. To make regex expressions case-aware, select the 'Case' option."

Characters with special meanings.

. Matches any single character except newline.
* Matches 0 or more of the preceding element.
+ Matches 1 or more of the preceding element.
? Matches 0 or 1 of the preceding element.
\ Escapes a metacharacter.
^ : Matches the start of a line.
$: Matches the end of a line.
| : OR operator.
\w matches a word
\b matches a word boundary

For example:
\b.at\b is a query that matches all 3-character strings that end with -at, such as, *bat, cat, eat, fat...*

\bhit\b the \w
matches all 3-word expressions with "hit the", e.g. hit the water, hit the deck, hit the fan

[] : Matches any one of the characters inside the brackets. gr[ae]y matches both *gray* and *grey*

The brackets are also used for classes
[A-Z] – matches any uppercase letter
[a-z] – matches any lowercase letter

What does the following query do?
\b[A-Z][a-z]*\b \b[A-Z][a-z]*\b

Exercise 5
Create a regular expression that would match all expressions such as:

- *was damaged, was eaten*
- *the bigger the better, the higher they go the harder they fall*