Linguistic Research 30(2), 141-161

A critical look at software tools in corpus linguistics*

Laurence Anthony
(Waseda University)

Anthony, Laurence. 2013. A critical look at software tools in corpus linguistics. *Linguistic Research* 30(2), 141-161. Corpora are often referred to as the 'tools' of corpus linguistics. However, it is important to recognize that corpora are simply linguistic data and that specialized software tools are required to view and analyze them. The functionality offered by software tools largely dictates what corpus linguistics research methods are available to a researcher, and hence, the design of tools will become an increasingly important factor as corpora become larger and the statistical analysis of linguistic data becomes increasingly complex. In this paper, I will first discuss how separating the data from the tools resolves various issues that are hotly debated within the field. Next, I will offer a critical look at the development of four generations of corpus tools, discussing their strengths and weaknesses. Then, I will discuss the role of programming in corpus linguistics tools creation and present a model for the development of future corpus tools. Finally, I will show a real-world example of a next-generation corpus tool that was developed for use in language learning. **(Waseda University)**

Keywords corpus linguistics, software tools, history, future, programming

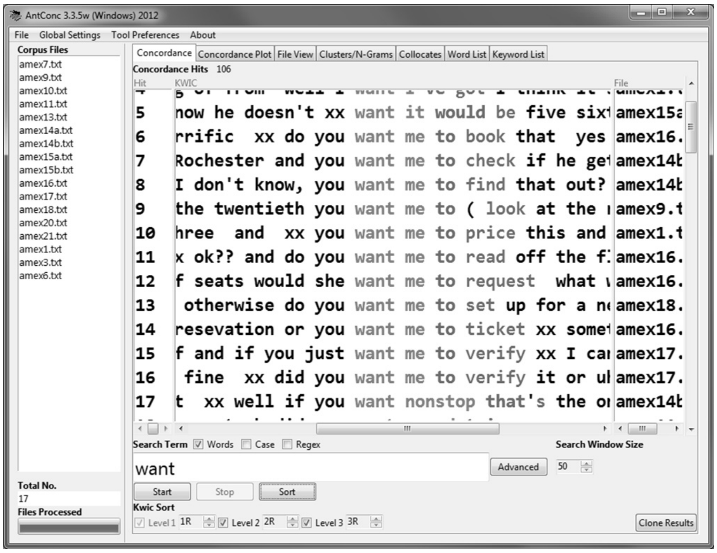
1. Introduction

Corpus linguistics is an applied linguistics approach that has become one of the dominant methods used to analyze language today. Biber et al. (1998) describe corpus linguistics as having four main features; 1) it is an empirical (experiment -based) approach in which patterns of language use that are observed in real language texts (spoken and written) are analyzed, 2) it uses a representative sample of the target language stored as an electronic database (a corpus) as the basis for the analysis, 3) it relies on computer software to count linguistics patterns as part of the

* This paper is an extended version of a plenary speech presented at the International Conference of Korea Association of Corpus Linguistics (KACL 2012), Pusan National University, Pusan, Korea. The author gratefully acknowledges the comments of the two reviewers.

analysis, and 4) it depends on both quantitative and qualitative analytical techniques to interpret the findings.

Within the community of corpus linguists, the above definition is well accepted and there is generally little disagreement about the nature of the approach. The main area for debate relates to the scope of corpus linguistics, with some researchers arguing that it is more than just a methodology and instead should be considered a new branch of linguistics (e.g. Tognini-Bonelli 2001). Another area for debate is how corpus experiments should proceed. One school of thought considers that direct observations of the corpus should be the starting point of analyses. This is usually termed a 'corpus-driven' approach (Tognini-Bonelli 2001), and it is often associated with the analyses of plain texts utilizing Key Word In Context (KWIC) concordance lines (see Figure 1). The rival school of thought argues that it is impossible to completely remove all pre-existing ideas about language before observing corpora, and thus all corpus analyses are essentially testing pre-existing linguistic theories (a model) against a representative sample of real language (the corpus data). This analysis subsequently leads to a refining of existing theories or perhaps the creation of new theories. This rival view of corpus linguistics methodology is usually referred to a 'corpus-based' approach (McEnery & Hardie 2012).



Concordance Hits 106

```
5   now he doesn't xx want it would be five sixt amex15a
6   rrific  xx do you want me to book that  yes  amex16.
7   Rochester and you want me to check if he get amex14b
8   I don't know, you want me to find that out?  amex14b
9   the twentieth you want me to ( look at the t amex9.t
10  hree  and  xx you want me to price this and  amex1.t
11  x ok?? and do you want me to read off the f: amex16.
12  f seats would she want me to request  what w amex16.
13   otherwise do you want me to set up for a ne amex18.
14  resevation or you want me to ticket xx somet amex16.
15  f and if you just want me to verify xx I can amex17.
16   fine  xx did you want me to verify it or ul amex17.
17  t  xx well if you want nonstop that's the or amex14b
```

Search Term ☑ Words □ Case □ Regex

want

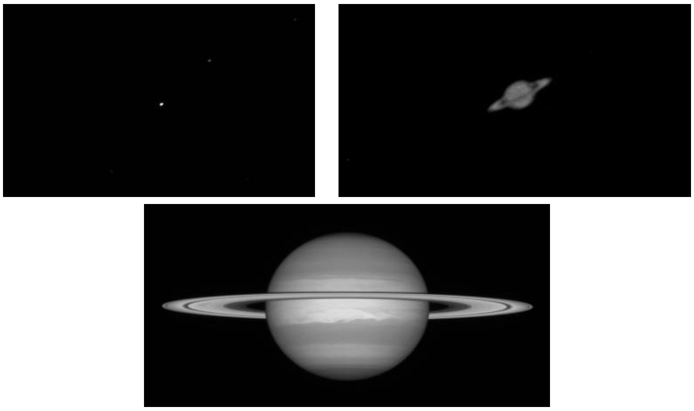
Search Window Size 50

Total No. 17

Kwic Sort ☑ Level 1 1R ☑ Level 2 2R ☑ Level 3 3R

Figure 1. KWIC Concordance View from AntConc 3.3.5 (Anthony, 2012)

However, one aspect of corpus linguistics that has been discussed far less to date is the importance of distinguishing between the corpus data and the corpus tools used to analyze that data. In any empirical field, be it physics, chemistry, biology, or corpus linguistics, it is essential that the researcher separates the actual data from the appearance of that data as seen through the observation tool. In astronomy, for example, observation tools can vary from the human eye and simple binoculars, to advanced reflector telescopes positioned in space. The data observed with these different tools may be the same, but the results of the observations will vary tremendously depending on which tool is selected (see Figure 2).



a) the human eye b) binoculars c) the Hubble telescope

Figure 2. Saturn observed through different observation tools

In corpus linguistics, on the other hand, researchers have tended to pay less attention to this separation. In fact, there is a continuing tendency within the field to ignore the tools of analysis and to consider the corpus data itself as an unchanging ‘tool’ that we use to directly observe new phenomenon in language. This view is revealed in quotes from some of the most prominent and well-respected members of the community.

> “...corpora [are becoming] more and more the normal tools of linguistic enquiry.”
>
> (Tognini-Bonelli 2001: 185)

> “Corpora offer an ideal instrument to observe and acquire socially

> established form/meaning pairings."
>
> (Bernadini 2004: 17)
>
> "A corpus is a powerful investigative tool for use in this revision."
>
> (Sinclair 2004a: 280)
>
> "Corpora have been likened to the invention of telescopes in the history of astronomy."
>
> (Hunston 2002: 20)

Interestingly, these same researchers have also shown an understanding for separating the two components, as illustrated in the following quotes:

> "...a corpus by itself can do nothing at all, being nothing more than a store of used language."
>
> (Hunston 2002: 20)
>
> "The essence of the corpus as against the text is that you do not observe it directly; instead you use tools of indirect observation, like query languages, concordancers, collocators, parsers, and aligners..."
>
> (Sinclair 2004b: 189)

One reason for blurring the separation between the data and tools of corpus linguistics is that the data itself can vary tremendously in quality and quantity depending on the research design. This has resulted in many researchers devoting much of their time and effort to collecting more data of a higher quality and then resigning themselves to using the available tools to observe this data. Another reason is that the tools used in corpus linguistics are software based, and thus, abstract in nature. To develop a tool for corpus linguistics requires an understanding of not only of human languages but also programming languages, computer algorithms, data storage methods, character encodings, and user-interface visual designs. Without a deep knowledge of these different aspects of software design and their impact on data analyses, it is easy to forget the crucial role that tools play.

In this paper, I will present a case for making a clear separation between corpus tools and linguistic data in corpus linguistics research. First, I will explain how a separation of the two components helps to resolve two long running debates in the field concerning the size and annotation of corpora and answers a common question about differences in the outputs of corpus linguistics tools. Next, I will present a brief account of the history of corpus tools development that will lead to an understanding of the strengths and weaknesses of popular tools used today. Finally, I will propose a model for future corpus tool design and development that does not rely on corpus linguists learning advanced programming techniques but can nevertheless lead to more powerful and flexible tools. Indeed, through the discussion, it will be made clear that such tools are becoming increasingly needed for cutting-edge corpus linguistics research.

2. Separating the data from the tool in corpus linguistics

2.1 Resolving the issue of corpus size

Corpus linguists have continually strived to build bigger and bigger corpora. The trend is most clearly seen in Figure 3, which shows the sizes in number of words of seven of the most influential corpora of the past 50 years. One of the most important reasons for this trend is that certain linguistic features are rare and will have a frequency of occurrence approaching zero in small-sized corpora. Clearly, in these cases, large corpora are unavoidable.

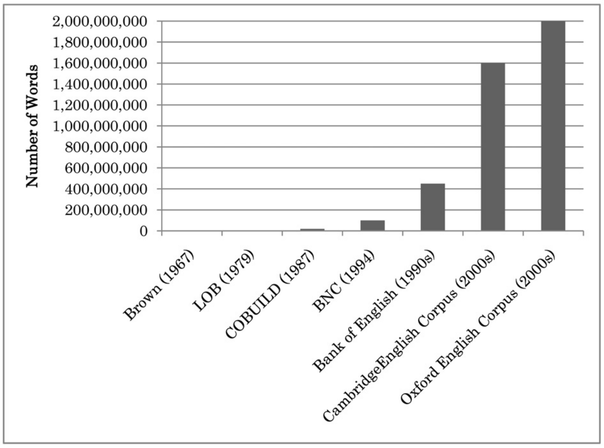


Figure 3. Growth in corpora sizes over 50 years

However, another reason stems from a view of corpus linguistics as being corpus-driven. Within this view, the corpus serves not to *test* a linguistic model but to *create* a linguistic model. As a result, if the corpus is small it can only provide a small window on the language phenomenon under investigation and hence, the results will only provide a partial picture of its 'true' complexity. On the other hand, a large corpus will provide a more complete view of the phenomenon and thus will always be superior to a smaller corpus. The argument is stated succinctly by Sinclair (2004b: 189), when he writes:

> *"There is no virtue in being small. Small is not beautiful; it is simply a limitation. If within the dimensions of a small corpus, using corpus techniques, you can get the results that you wish to get, then your methodology is above reproach - but the results will be extremely limited..."*

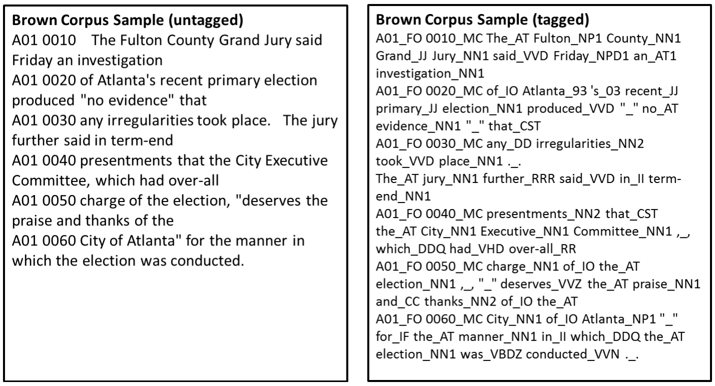
Others, however, have argued that small corpora can also be useful. McEnery & Wilson (2001), for example, give examples of interesting corpus studies on critical discourse that use small corpora, and Scott and Tribble (2006: 179) analyze the first story from Samuel Beckett's "Texts for nothing" series to reveal patterns about the author's writing. Anthony (2009) explains the importance of small-corpora studies using the analogy of astronomy. In astronomy, some researchers may be interested in studying galaxies and from these analyses create models of the universe and how it was born. In contrast, other researchers may be interested in studying a single star, such as our sun, and understanding its life-cycle, solar-flare seasons, and radiation emittance patterns. Few would doubt the importance of studying our own sun, and similarly, there is value in studying a small corpus, such as a story of Samuel Beckett, the works of J. K. Rowling, or a volume of research articles in biochemistry.

The value of a corpus is clearly dependent not on its size but on what kind of information we can extract from it. Therein lies the importance of corpus tools; we need to have tools that can provide us with the information that we desire. For example, if we are interested in observing which characters *Harry Potter* interacts with through the J. K. Rowling *Harry Potter* series, we need tools that can record and visualize these interactions in some way. Unfortunately, mainstream corpus toolkits do not provide such a feature. It can be conjectured that more progress would be made in this regard if researchers focused less on debating the merits and weaknesses of corpora sizes and more of the validity, interest, and value of the analyzes they carry out within the scope of the tools available to them.

2.2 Resolving the issue of corpus annotation

Another debate in corpus linguistics regards the value of annotating corpus data with Part-Of-Speech (POS) tags, semantic tags, header markup, or other relevant information (see Figure 4). The strongest argument against annotation is that it 'contaminates' the original data making it more difficult to see language patterns. This view has been voiced most loudly by researchers adopting the corpus-driven approach, who see the corpus itself as the starting point for analysis. For this group of researchers, adding linguistic tags and markers to the data essentially overlays the data with a pre-existing linguistic model. As the aim of these researchers is to observe new linguistics patterns within the corpus, annotation serves no purpose and can be considered as noise. In the words of Sinclair (2004b: 191),

> *"The interspersing of tags in a language texts is a perilous activity, because the text thereby loses its integrity, and no matter how careful one is the original text cannot be retrieved...In corpus-driven linguistics you do not use pre-tagged text, but you process the raw text directly and then the patterns of this uncontaminated text are able to be observed."*



Brown Corpus Sample (untagged)

```
A01 0010   The Fulton County Grand Jury said
Friday an investigation
A01 0020 of Atlanta's recent primary election
produced "no evidence" that
A01 0030 any irregularities took place.   The jury
further said in term-end
A01 0040 presentments that the City Executive
Committee, which had over-all
A01 0050 charge of the election, "deserves the
praise and thanks of the
A01 0060 City of Atlanta" for the manner in
which the election was conducted.
```

Brown Corpus Sample (tagged)

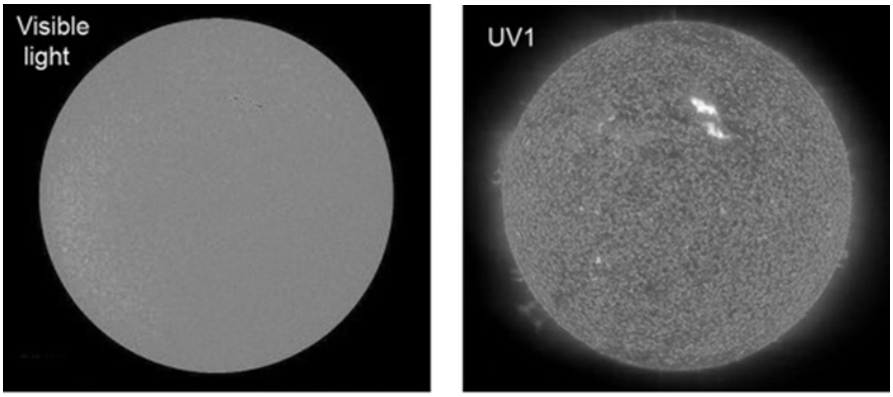
```
A01_FO 0010_MC The_AT Fulton_NP1 County_NN1
Grand_JJ Jury_NN1 said_VVD Friday_NPD1 an_AT1
investigation_NN1
A01_FO 0020_MC of_IO Atlanta_93 's_03 recent_JJ
primary_JJ election_NN1 produced_VVD "_" no_AT
evidence_NN1 "_" that_CST
A01_FO 0030_MC any_DD irregularities_NN2
took_VVD place_NN1 ._.
The_AT jury_NN1 further_RRR said_VVD in_II term-
end_NN1
A01_FO 0040_MC presentments_NN2 that_CST
the_AT City_NN1 Executive_NN1 Committee_NN1 ,_,
which_DDQ had_VHD over-all_RR
A01_FO 0050_MC charge_NN1 of_IO the_AT
election_NN1 ,_, "_" deserves_VVZ the_AT praise_NN1
and_CC thanks_NN2 of_IO the_AT
A01_FO 0060_MC City_NN1 of_IO Atlanta_NP1 "_"
for_IF the_AT manner_NN1 in_II which_DDQ the_AT
election_NN1 was_VBDZ conducted_VVN ._.
```

Figure 4. A sample of the Brown Corpus with a) no annotation and b) added Part-Of-Speech (POS) tags

Supporters of corpus annotation, on the other hand, usually regard the addition of annotation as a necessary step in order to test a particular linguistic theory. For example, in order to test whether a certain genre (e.g. newspaper articles) is written more commonly in the present tense than past tense, it is necessary to count the occurrence of the two tenses in a representative sample of target texts. If all verbs in the corpus are tagged for tense, it then becomes a simple task to confirm or reject the hypothesis.

Again, the debate on the value of annotation can be easily resolved by refocusing the discussion on the tools used to analyze corpora. Modern corpus tools are easily able to show or hide different layers of annotation or markup of texts. If a researcher would like to analyze the raw texts, the various layers of annotation can be hidden. On the other hand, if a researcher needs to count verb tenses or any other linguistic feature and has tagged the corpus for them, the software can then utilize this additional information and provide the researcher with a result almost instantaneously.

The value of annotation can be understood further if it is considered as a way of marking features in texts that are not immediately observable when the raw text is seen with the naked eye. Once the annotation process is completed, these additional layers can be observed or ignored depending on the software tool used. The concept is similar to the way in which tools are used in other fields. In astronomy, for example, many astronomical objects, such as stars and planets, can be easily observed with the naked eye. However, by utilizing different tools, additional layers of information, such as ultraviolet emissions, can also be observed and analyzed to reveal new features of the target object (see Figure 5).



a) visible light b) ultraviolet light

Figure 5. The Sun as seen in different light regions (data obtained from NASA)

2.2 Resolving the issue of conflicting results in corpus linguistics software tools

Corpus linguists become most aware of the tools they use when their studies are replicated with a different tool and the results do not match. Surprisingly, this is a common situation, despite the fact that corpus linguistics is an empirical approach in which studies are supposed to be replicable and results verifiable. The reasons for disagreement between corpus studies are complex. However, one reason for a discrepancy is that the linguistic feature under investigation is not being counted in the same way by the tools. This can be illustrated using the example of counting word frequencies in a corpus. Word frequency is a linguistic phenomenon that many corpus researchers are interested in, whether it is to determine the complexity of a particular text in an English for Specific Purposes (ESP) study, the bias of a particular writer in a Critical Discourse Analysis (CDA) study, or any number of other linguistic research interests. When counting word frequencies, the corpus tool must first determine what a ‘word’ is. This is the job of the software tool developer and the word definition will be hardcoded into the computer program underlying the tool. One tool developer may code words as strings of letters A to Z. A second developer may include numbers 0 to 9 as part of a ‘word’, and a third developer may provide a default implementation (e.g. based on letters) but allow the user to change that implementation through the tool's user interface. The result is that all three tools may produce different word frequency counts for the same corpus. Table 1 shows a real-world example of the effect when counting the most frequent words in the US Presidential Inaugural Address of 2009 using three commonly used corpus tools.

Table 1. Top ten words appearing in the US Presidential Inaugural Address of 2009 according to *WordSmith Tools*, *Monoconc Pro*, and *AntConc*

	WordSmith Tools (Scott 2012)		*MonoConc Pro* (Barlow 2000)		*AntConc* (Anthony 2012)	
Rank		Frequency		Frequency		Frequency
1	the	136	the	136	the	136
2	and	111	and	111	and	111
3	of	82	of	82	of	82
4	to	71	to	71	to	71
5	our	68	our	68	our	68
6	we	60	we	60	we	62
7	that	50	that	50	that	50
8	a	46	a	46	a	46
9	is	36	is	36	is	36
10	in	26	in	26	in	26

Clearly, the results match closely. However, at rank 6, the word "we" appears 60 times according to both *WordSmith Tools* and *Moncone Pro* but 62 times according to *AntConc*. On first sight, this may suggest that *AntConc* contains a bug. However, on closer inspection of the tools, it can be found that both *WordSmith Tools* and *Moncone Pro* treat apostrophes (') as part of a word, whereas AntConc does not. As a result, the following two occurrences of "we" in the inaugural speech are not counted by the first two of these tools as they are considered to be different 'words' in the corpus:

...and bind us together. **We**'ll restore science...
...friends and former foes, **we**'ll work tirelessly to...

Rather surprisingly, *AntConc* is the only tool of the three that explicitly details the definition of 'words' that it uses. It is also unique in allowing the user to completely redefine the word definition to match that of other software tools, or implement a completely new definition, for example, when working with non-English texts, where the concepts of ‘letters’ and ‘numbers’ may not be easily transferable.

What is important to remember is that differences in the way tools are designed will have an impact on almost all corpus analyses. For example, differences in the definition of a word will impact not only on word frequency counts, but also on the values of type-token ratios, the strengths of collocation between words, and the N-grams and keywords produced for a corpus

3. Advances in software tools development for corpora analyses

3.1 Four generations of corpus tools

McEnery and Hardie (2012) describe four generations of software tools. The 1st generation appeared in the 1960s and 1970s and ran on mainframe computers. These tools were only able to process the ASCII (http://www.asciitable.com/) character set, which is essentially the letters A-Z, English punctuation, numbers, and a limited number of symbols, such as those for mathematical calculations. As a result, they were limited to processing only English corpora. Most of the tools were designed for a single function, such as counting the number of words in a text or producing KWIC concordance lines. Examples of these include *Concordance Generator* (Smith 1966), *Discon* (Clark 1966), *Drexel Concordance Program* (Price 1966), *Concordance* (Dearing 1966), and *CLOC* (Reed 1978). The last of these was used in the well-known COBUILD project at The University of Birmingham which was headed by John Sinclair. Interestingly, the majority of the tool concepts proposed in the 1960s still serve as the foundations of modern corpus tools, although of course, the tools of today run much faster. As an example, the *Discon* tool took approximately 4 min. to process 1000 lines of poetry (Clark 1966). Today’s tools would be able to perform the same task in a fraction of a second.

The 2nd generation of corpus tools in McEnery and Hardie’s (2012) review cover the tools that were introduced in the 1980s and 1990s. These were again limited to processing ASCII and had limited functionality. However, their advantage was that they could run on the early personal computers, allowing researchers to carry out small-scale studies. They also allowed teachers to introduce corpora analyses into the language learning classroom in what Johns (2002) described as a Data-Driven Learning (DDL) approach. Examples of software from this generation include the *Oxford Concordance Program* (Hockey 1988), *Longman Mini-Concordancer* (Chandler 1989), *Kaye concordancer* (Kaye 1990), and *MicroConcord* (Scott & Johns 1993).

Most of the current tools used by corpus linguists are classed by McEnery and Hardie (2012) as third generation tools. Early versions of these tools began appearing in the late 1990s but many are continuing to be developed and improved today. The main advantages of these tools over earlier ones are that they offer a multitude of functions, include common statistical methods, have improved scalability to work with larger corpora, offer some degree of multi-language support by processing characters outside of the ASCII character set, and include user-friendly interfaces that are more suitable for users with little computer experience. Examples of third generation tools include *WordSmith Tools* (Scott 1996-2012), *MonoConc Pro* (Barlow 2000), and *AntConc* (Anthony 2004-2012) that were mentioned earlier in this paper.

The biggest limitation with third generation corpus tools is that they struggle to handle very large corpora of over 100 million words. Today, an increasing number of corpora are being released that are automatically compiled by scraping data from Internet sites. These corpora can be several billion words in length, and the architecture of third generation tools is not appropriate to process them. Another limitation is that publishers are becoming increasingly sensitive about allowing their data to be used for research purposes. As a result, collections of texts can no longer be compiled and distributed for analysis with corpus tools on a personal computer. The response to these two problems has been the creation of fourth generation tools, such as *corpus.byu.edu* (Davies 2013), *CQPweb* (Hardie 2013), *SketchEngine* (Kilgariff 2013), and *Wmatrix* (Rayson 2013). These tools offer better scalability by storing the corpus in a Web server database and pre-indexing the data to allow for fast searches. They also offer protection from copyright issues by preventing users from viewing the complete corpus. Rather, users must access the corpus through a user-interface that presents only a small frame of the corpus data at a single time. The interface does, however, usually allow users to search the entire corpus and generate standard results from the entire corpus, such as KWIC concordance lines and word frequency lists.

Despite the above advantages of fourth generation tools, they also have a number of limitations. First, they may be considered as 'overkill' if a user wants to compile a small corpus and perform a simple analysis on it. Fourth generation tools require the data to be cleaned, processed, reformatted, indexed, and finally uploaded to a server before the analysis can begin. Also, to access the server in the first place, the user may have to register for the service, agree to various licenses, and possibly pay a monthly subscription charge. The alternative is to setup the tool on a personalized server, but this would require the user purchasing a server computer, setting up the server, installing the corpus tool software, and then maintaining the server for the duration of the project. Another problem is that many of these fourth generation tools are inappropriate for the analysis of corpus data of a sensitive nature, such as internal business meeting minutes, university entrance exams, and personal diaries, as they require the data to be uploaded to an external server. A third problem relates to the fact that some fourth generation tools are linked directly to particular corpora (usually copyright protected) and offer no option to be used to analyze other corpus data. A fourth problem that relates to the third is that when a new (copyright protected) corpus is created, it is inevitably released via a new corpus/tool setup, resulting in an explosion of one-off, web-based, single corpus interfaces, each with idiosyncratic positioning of controls and operation features. A final problem is that fourth generation tools blur the boundaries between the corpus data and the tool used to observe it. Due to the way these tools store the corpus data in an indexed form on an external server, users have no way to observe the raw data directly with their own eyes. All interactions must be through the tool (usually a web browser user interface). When analyzing corpora with these tools, it is easy for researchers to forget the filtering effect of the tool and begin using it in an unquestioning manner.

Based on the above historical account of tools development, it is clear both third and fourth generation tools have strengths and weaknesses. This is reflected in Figure 6, which shows the results of a recent survey of corpus linguists asking them which computer tools they most often use for research (Tribble, 2006). Clearly, third generation tools, such as *AntConc* and *WordSmith Tools*, continue to be popular amongst researchers. However, the figure also shows the popularity of fourth generation tools, with the *corpus.byu.edu* site ranked first in the list and *Sketch*

Engine and *Wmatrix* ranked 4th and 8th. It should be noted, however, that *corpus.byu.edu* is the only tool listed that is *not* a general purpose tool. The high ranking of the tool is likely to be related to the fact that one of the corpora it gives access to is the largest corpus of contemporary American English.

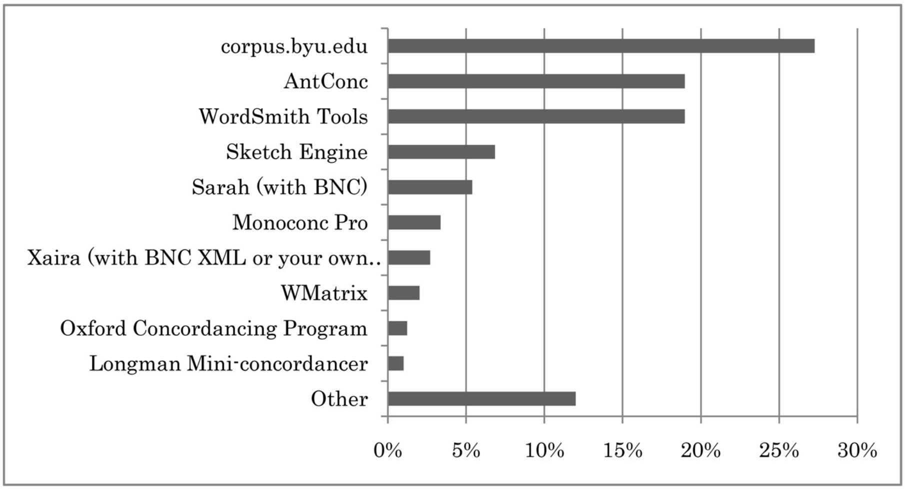


Figure 6. Most popular tools used for analyzing corpora (Tribble 2012)

3.2 Current issues in corpus tool development

In general, the current tools available to corpus linguists are fast and feature rich. They also offer researchers access to a wide range of functions to analyze KWIC concordancers, distribution plots, clusters and N-grams, collocates, word frequencies, and keywords. On the other hand, most of the tools are still English-centric in that they only allow access to English corpora. They are also generally researcher-centric in that they do not always lend themselves to easy use in the classroom with learners. An additional problem is that they generally do not explicate the settings they use, the most fundamental being their internal definition of words. Finally, they all offer a different user-experience, because each tool is created in isolation and thus offers a different user interface, control flow, and functionality.

One solution to the current problems with corpus tools has been suggested by researchers such as Biber et al. (1998), Gries (2009a), and Weisser (2009). They encourage corpus linguists to learn programming and develop their own analytical tools. Biber et al. (1998: 256) argue that if a corpus linguist can develop their own tools they can then do analyses not possible with concordancers, do analyses more quickly and more accurately, tailor the output to fit their own research needs, and analyze a corpus of any size. On a similar note, Gries (2009a: 11-12) argues that using a preexisting tool leaves researchers at the mercy of the company or individual selling them, whereas programming allows them to take control of their own research agenda.

There are clearly advantages in corpus linguists learning a programming language, not only in that it provides them with more flexibility to develop tools for a particular task, but also because it gives them an insight into the issues that all tool developers have to address when developing general purpose tools. On the other hand, as Anthony (2009: 5) explains,

> *"The reality for most corpus researchers, however, is that computer programming is in a completely different world...without extensive training in programming...it is likely that these tools would be more restrictive, slower, less accurate and only work with small corpora."*

In the following section, I will propose an alternative model for creating the next generation of corpus tools that overcomes the limitations of the current third and fourth generation tools, but also does not require corpus linguistics to learn advanced programming techniques.

4. A proposal for future corpus tool software design

Current trends in corpus linguistics research suggest that future studies will rely increasingly on large corpora, advanced functionality, and sophisticated statistical methods, such as those discussed by Bayern (2008), Gries (2009b), and others. The complexities in developing tools that can handle these requirements are a major challenge and certainly beyond the technical capabilities of corpus linguistics researchers who have completed an introductory course in programming. In astronomy, researchers face a similar dilemma in that they require increasingly sophisticated tools in order to look deeper into the university and collect more sophisticated measures of stars, planets, and other celestial objects. However, few researchers in astronomy have begun studying how to build advanced optical and radio telescopes. Rather, they form research teams that include members of the science and engineering community with backgrounds allowing them to build the desired tools. Similarly, I propose here that researchers in corpus linguistics should also work more closely with members of the science and engineering community, such as computer scientists and software engineers, in order to design and build the next generation of corpus tools. Within these teams, thought should be given to the needs of researchers, teachers, and learners so that the tools have maximum applicability. Researchers adopting a corpus-based approach, for example, need tools that can handle annotated corpora and allow access to sophisticated analytical functions and statistical measures. Researchers with a corpus-driven background, on the other hand, have less need for annotation and statistical measures and so the tools need to be able to hide these features and functions from the interface. Teachers generally have little need for a research tool. They need to be able to quickly and easily access a corpus, filter the results to show only those that are directly relevant, and be able to display, save, and perhaps print those results for use in teaching materials. Similarly, students in a DDL classroom do not need a research tool. They need a corpus tool that gives them access to a corpus in an easy and intuitive way. They also need a tool to show them results that are immediately applicable to a given learning task, such as finding a common collocation of a word or showing them a language pattern that is useful when writing a research paper. All these issues relate directly to tool design.

Finally, the complexity of future needs will almost certainly require the efforts of many people. This suggests that corpus tool development should be an open source initiative with tool components being developed in a modular fashion. By dividing tool components in this way, it becomes easier for tool functions and features to be extended, modified, or simplified depending on the need.

As an example of the above approach to corpus tool development, Anthony et al. (2011) have recently headed a team to develop a next generation corpus tool called *AntWebConc*. This tool has been built in an open-source and modular fashion, with input from researchers and teachers, and incorporating feedback from English foreign language learners at a Japanese university that have used a prototype of the tool in a DDL classroom setting. To avoid the danger of becoming just another web-based corpus tool, *AntWebConc* is designed to serve as a framework that can host a range of different single and parallel corpora. This is achieved by ensuring that the system is developed using a Model-View-Controller (MVC) architecture, as illustrated in Figure 7. Although *AntWebConc* runs on a server, its components are almost completely cross-server compatible and portable. This means that a user wishing to use the framework can simple copy the framework files into a standard website location and immediately have access to their corpora and tools. An example of a user-derived implementation of the *AntWebConc* is the *WebParaNews* parallel concordancer used in DDL classes at Nihon University in Japan (Anthony 2013). This tool is shown in Figure 8.

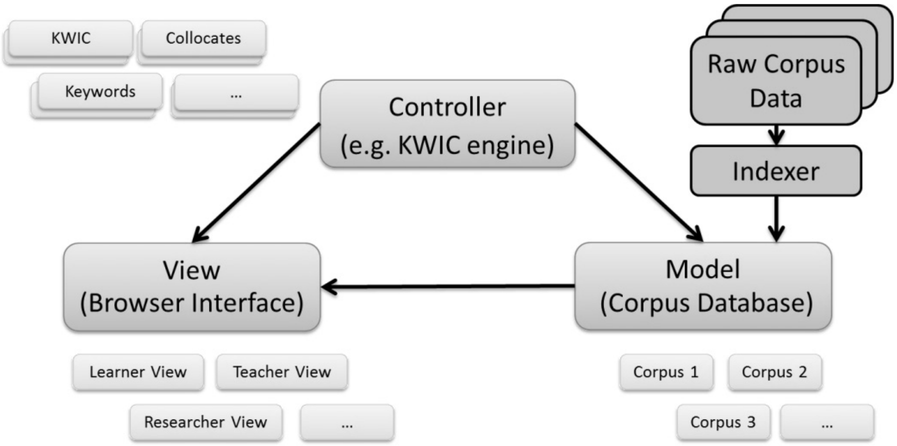
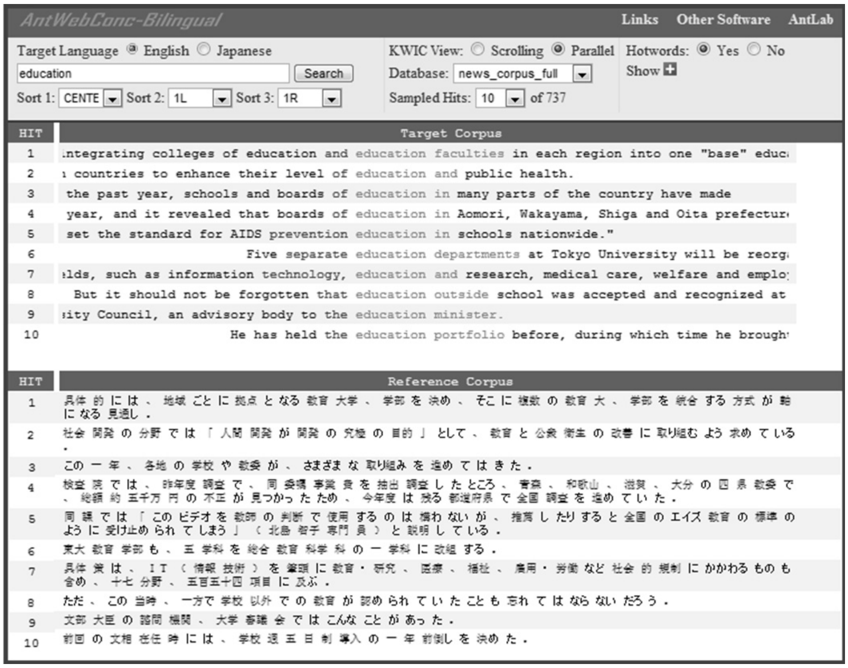


Figure 7. The Model-View-Controller (MVC) Architecture of *AntWebConc*



Target Language ● English ○ Japanese

education

Sort 1: CENTE Sort 2: 1L Sort 3: 1R

KWIC View: ○ Scrolling ● Parallel

Database: news_corpus_full

Sampled Hits: 10 of 737

Hotwords: ● Yes ○ No

HIT	Target Corpus
1	ıntegrating colleges of education and education faculties in each region into one "base" educ
2	ı countries to enhance their level of education and public health.
3	the past year, schools and boards of education in many parts of the country have made
4	year, and it revealed that boards of education in Aomori, Wakayama, Shiga and Oita prefectur
5	set the standard for AIDS prevention education in schools nationwide."
6	Five separate education departments at Tokyo University will be reorg
7	ılds, such as information technology, education and research, medical care, welfare and emplo
8	But it should not be forgotten that education outside school was accepted and recognized at
9	ıity Council, an advisory body to the education minister.
10	He has held the education portfolio before, during which time he brough

HIT	Reference Corpus
1	具体 的 に は 、 地域 ごと に 拠点 と なる 教育 大学 、 学部 を 決め 、 そこ に 複数 の 教育 大 、 学部 を 統合 する 方式 が 軸 に なる 見通し .
2	社会 開発 の 分野 で は 「 人間 開発 が 開発 の 究極 の 目的 」 として 、 教育 と 公衆 衛生 の 改善 に 取り組む よう 求め て いる .
3	この 一 年 、 各地 の 学校 や 教委 が 、 さまざま な 取り組み を 進め て は きた .
4	検査 院 で は 、 昨年度 調査 で 、 同 委嘱 事業 費 を 抽出 調査 し た ところ 、 青森 、 和歌山 、 滋賀 、 大分 の 四 県 教委 で 、 総額 約 五千万 円 の 不正 が 見つかっ た ため 、 今年度 は 残る 都道府県 で 全国 調査 を 進め て い た .
5	同 課 で は 「 この ビデオ を 教師 の 判断 で 使用 する の は 構わ ない が 、 推薦 し たり する と 全国 の エイズ 教育 の 標準 の よう に 受け止め られ て しまう 」 （ 北島 智子 専門 員 ） と 説明 し て いる .
6	東大 教育 学部 も 、 五 学科 を 総合 教育 科学 科 の 一 学科 に 改組 する .
7	具体 策 は 、 ＩＴ （ 情報 技術 ） を 筆頭 に 教育・研究 、 医療 、 福祉 、 雇用・労働 など 社会 的 規制 に かかわる もの も 含め 、 十七 分野 、 五百五十四 項目 に 及ぶ .
8	ただ 、 この 当時 、 一方で 学校 以外 で の 教育 が 認め られ て い た こと も 忘れ て は なら ない だろう .
9	文部 大臣 の 諮問 機関 、 大学 審議 会 で は こんな こと が あっ た .
10	前回 の 文相 在任 時 に は 、 学校 週 五 日 制 導入 の 一 年 前倒し を 決め た .

Figure 8. The *WebParaNews* Parallel Concordancer Based Running on the *AntWebConc* framework

5. Summary and conclusions

In this paper, I have discussed the role of software tools in corpus linguistics research. I first explained that corpus tools are critical to the success of all corpus-based and corpus driven research projects, as well as Data-Driven Learning (DDL) approaches in the classroom. I also explained the need for researchers to clearly separate the corpus data from the corpus tool when addressing problems and issues in corpus linguistics. Many of the strongly debated issues and concerns in corpus linguistics can be addressed by simply understanding the role and position of corpus tools in a research project.

Next, I gave a brief summary of the history and development of corpus tools spanning almost 50 years. Four generations of tools were covered and the advantages and limitations of each generation of tools were discussed. This led to a proposal for developing the next generation of corpus tools that are built in an open-source and modular fashion, and are developed as a community effort, incorporating the skills and knowledge of corpus linguists, computer scientists, software engineers, as well as language researchers, teachers, and learners. A real-world example of such a tool was also presented.

Corpus linguistics is becoming one of the dominant approaches used in linguistics research and it is increasingly being used in the language learning classroom. The success of the approach is intrinsically related to the tools used to access, analyze, and display the results of corpus searches. It is hoped that this paper has provided a new perspective on corpus tools that will lead to continued growth of corpus linguistics tools and the field as a whole.

References

Anthony, L. 2009. Issues in the design and development of software tools for corpus studies: The case for collaboration. In P. Baker (ed.), *Contemporary corpus linguistics,* 87-104. London, UK: Continuum Press.

Anthony, L. 2012. *AntConc* (Version 3.3.5) [Computer Software]. Tokyo, Japan: Waseda University. Available from http://www.antlab.sci.waseda.ac.jp/.

Anthony, L. 2013. *WebParaNews* [Computer Software]. Tokyo, Japan: Waseda University. Available from http://www.antlab.sci.waseda.ac.jp/.

Anthony, L., Chujo, K, and Oghigian, K. 2011. A novel, web-based, parallel concordancer for use in the ESL/EFL classroom. In J. Newman, H. Baayen, & S. Rice (eds.), *Corpus-based Studies in Language Use, Language Learning, and Language Documentation,* 123-138. New York: Rodopi.

Baayen, R. H. 2008. *Analyzing linguistic data: A practical introduction to statistics using R.* Cambridge: Cambridge University Press.

Barlow, M. 2000. *MonoConc Pro (Version 2.2)* [Computer Software]. Available from http://www.athel.com/mono.html.

Bernardini, S. 2002. Corpora in the classroom. In J. McH. Sinclair (ed.) *How to use corpora in language teaching,* 15-36. Amsterdam: John Benjamins.

Biber, D, Conrad, S, and Reppen, R. 1998. *Corpus linguistics*. Cambridge: Cambridge University Press.

Chandler, B. 1989. *Longman Mini-Concordancer* [Computer Software]. Harlow, UK: Longman Press.

Clark, R. 1966. *Computers and the Humanities*. Vol. 1, Issue 3, p. 39.

Davies, M. 2013. *corpus.byu.edu*. Accessed on April 30, 2013.

Dearing, V. A. 1966. *Computers and the Humanities*. Vol. 1, Issue 3, p. 39-40.

Gries, S. Th. 2009a. What is corpus linguistics? *Language and Linguistics Compass*, 3, pp. 1 – 17.

Gries, S. Th. 2009b. *Quantitative corpus linguistics with R: A practical introduction*. New York: Routledge.

Hardie, A. 2013. *CQPweb* [Computer Software]. Available from http://cwb.sourceforge.net/cqpweb.php.

Hockey, S. 1988. *Oxford Concordance Program* [Computer Software]. Oxford, UK: Oxford University Press.

Hunston, S. 2002. *Corpora in applied linguistics*. Cambridge: Cambridge University Press.

Johns, T. 2002. Data-driven learning: The perpetual challenge. In Kettermann, B. and Marko, G. (eds.) *Teaching and learning by doing corpus linguistics*. Amsterdam: Rodopi, 107-117.

Kaye, G. 1990. *Kaye Concordancer* [Computer Software]. No longer available.

Kilgariff, A. 2013. *SketchEngine* [Computer Software]. Available from http://www.sketchengine.co.uk/.

McEnery, T., and Hardie, A. 2012. *Corpus linguistics: Method, theory and practice*. Cambridge: Cambridge University Press.

McEnery, T., and Wilson, A. 2001. *Corpus linguistics* (2nd edition). Edinburgh: Edinburgh University Press.

Price, K. 1966. *Computers and the Humanities*. Vol. 1, Issue 3, p. 39.

Rayson, P. 2013. *Wmatrix*. [Computer Software]. Available from http://ucrel.lancs.ac.uk/wmatrix/.

Reed, A. 1978. *CLOC* [Computer Software]. Birmingham, UK: University of Birmingham.

Scott, M., and Tribble C. 2006. *Textual patterns*. Amsterdam: John Benjamins.

Scott, M. 2012. *WordSmith Tools (Version 5.0)* [Computer Software]. Available from http://www.lexically.net/software/index.htm.

Scott, M., and Johns, T. 1993. *MicroConcord* [Computer Software]. Available from http://www.lexically.net/software/index.htm.

Sinclair, J. McH. 2004a. New evidence, new priorities, new attitudes. In J. McH. Sinclair (ed.) *How to use corpora in language teaching,* 271-299. Amsterdam: John Benjamins.

Sinclair, J. McH. 2004b. *Trust the text: Language, corpus and discourse*. London: Routledge.

Smith, P. H. 1966. *Computers and the Humanities*. Vol. 1, Issue 2, p. 39.

Tognini-Bonelli, E. 2001. *Corpus linguistics at work*. Amsterdam: John Benjamins.

Tribble, C. 2012. Teaching and language corpora: Quo Vadis? 10th *Teaching and Language Corpora Conference (TALC)*. Warsaw, 11th-14th July 2012.

Weisser, M. 2009. *Essential programming for linguistics*. Edinburgh: Edinburgh University Press.

Laurence Anthony
Faculty of Science and Engineering
Waseda University
3-4-1 Okubo, Shinjuku-ku,
Tokyo 169-8555, Japan
E-mail: anthony@waseda.jp

Received: 2013. 04. 30
Revised: 2013. 08. 23
Accepted: 2013. 08. 26