

3

Corpus tools for lexicographers

ADAM KILGARRIFF AND IZTOK KOSEM

3.1 Introduction

To analyse corpus data, lexicographers need software that allows them to search, manipulate and save data, a ‘corpus tool’. A good corpus tool is the key to a comprehensive lexicographic analysis—a corpus without a good tool to access it is of little use.

Both corpus compilation and corpus tools have been swept along by general technological advances over the last three decades. Compiling and storing corpora has become far faster and easier, so corpora tend to be much larger than previous ones. Most of the first COBUILD dictionary was produced from a corpus of eight million words. Several of the leading English dictionaries of the 1990s were produced using the British National Corpus (BNC), of 100 million words. Current lexicographic projects we are involved in use corpora of around a billion words—though this is still less than one hundredth of one percent of the English language text available on the Web (see Rundell, this volume).

The amount of data to analyse has thus increased significantly, and corpus tools have had to be improved to assist lexicographers in adapting to this change. Corpus tools have become faster, more multifunctional, and customizable. In the COBUILD project, getting concordance output took a long time and then the concordances were printed on paper and handed out to lexicographers (Clear 1987). Today, with Google as a point of comparison, concordancing needs to be instantaneous, with the analysis taking place on the computer screen. Moreover, larger corpora offer much higher numbers of concordance lines per word (especially for high-frequency words), and, considering the time constraints of the lexicographers (see Rundell, this volume), new features of data summarization are required to ease and speed the analysis.

In this chapter, we review the functionality of corpus tools used by lexicographers. In Section 3.2, we discuss the procedures in corpus preparation that are required for some of these features to work. In Section 3.3, we briefly describe some leading tools

and provide some elements of comparison. The Sketch Engine is introduced in Section 3.4. This innovative corpus tool was developed by the first author's company and used in the second author's projects; it has become a leading tool for lexicography and other corpus work since its launch in 2004. The Sketch Engine uses the formalisms and approach of the Stuttgart tools; it is available as a Web service, and corpora from forty different languages are already loaded within it. We focus first on basic features, which are used also by non-lexicographers, and then move on to the features that are targeted mainly at lexicographers. Section 3.5 is dedicated to the user-friendliness of corpus tools, a topic that, although rarely discussed in the literature, is becoming more relevant as corpus tools become more complex. Finally, we conclude by considering how corpus tools of the future might be designed to further assist lexicographers.

3.2 Preparing the corpus for automatic analysis

Many features of corpus tools work only if the corpus data is properly prepared. The preparation of a corpus has two parts: preparing the metadata, or 'headers', and preparing the text.

A corpus is a collection of documents, and instances of words come from a variety of documents representing different types of text. The lexicographer examining the instances of a word may want to know from what kind of text a particular instance has been extracted (i.e. the details and characteristics of the document, such as its date of publication, author, mode [spoken, written], and domain). For this to work, each document must come with metadata, usually located in a 'header', which lists the features of the document, in a way that the corpus tool can interpret. Using headers, corpus tools can not only provide information on the texts, but also use them to limit the searches to particular text types, build word lists and find keywords for a text type, etc.

Preparing the text starts with identifying and managing the character encoding and then typically involves marking up the text with (1) sections, paragraphs, and sentences, (2) tokens, (3) lemmas, (4) part-of-speech tags, and (5) grammatical structure. Each text comes with its character encoding. This is the way in which each particular character is encoded in a series of ones and zeros. Widely used character-encodings include ASCII, ISO 8859-1 (also called Latin-1), Big-5 (for Chinese), and UTF-8. There are many different character-encodings, most of which are language-specific or writing-system specific, and this can create a wide range of problems of misinterpretation where the system assumes that one encoding has been used, but in fact a different one was involved. In Latin-script languages, problems most often arise with accented and other non-standard characters since standard characters (a-z, A-Z, 0-9, etc.) are generally encoded in the same way. Over time, a growing proportion of documents are being encoded in UTF-8, which is

based on the Unicode standard; however, most documents do not yet use Unicode or UTF8, and the character encoding typically has to be guessed, with each text then converted to the same, standard, encoding.

Sentence, paragraph and section mark-up (using structural tags) supports functionality such as the display of sentences, or not seeking patterns spanning sentence ends. Tokenization is the process of identifying the tokens, usually the words, which the user typically searches for. For some languages, such as Chinese and Arabic, this is a major challenge, since in Chinese there is no white space between words, and in Arabic many grammatical words are written as clitics, without white space between them and the core word. This is not a great problem in English since, most of the time, white space reliably indicates a word break: there are just a few difficult cases, mostly relating to apostrophes (e.g. whether *don't* is counted as one token or two—*do* and *n't*) and hyphens (*co-operate*, *first-hand*). How a text has been tokenized has an effect on searching, filtering, sorting, and many other features.

Lemmatization (also known as morphological analysis) is (at its simplest) the process of identifying the base form of the word (or the dictionary headword), called a lemma. In a language such as English, many corpus words may be instances of more than one lemma. Thus *tricks* may be the plural of the noun, or the present tense, third person singular form of the verb. The process of identifying, by computer, which part of speech applies in a particular context is called part-of-speech (POS) tagging. Finally, parsing is used to annotate the syntactic structure of each sentence in the corpus.

Once all the words in a corpus have been lemmatized and part-of-speech tagged, and this information made available to the corpus tool, each word in the corpus can be thought of as a <word form, lemma, POS-tag> triple, and searches can be specified in terms of any of these three parts. In addition to simple searches for single words, lexicographers often want to search for a phrase or some other more complex structure. A good corpus tool will support complex searches (such as those by surrounding context), while keeping the interface simple and user-friendly for the simple searches that users most often want to do. Another form of search uses a corpus query language (CQL), such as that developed at the University of Stuttgart (Christ 1995). This allows sophisticated structured searches, matching all- or part-strings, to be built for as many fields of information as are provided (such as the word form, lemma, and POS-tag).

3.3 An overview of corpus tools

The number of corpus tools available has grown over the past thirty years, as not only lexicographers, but also researchers from other linguistics sub-disciplines have become aware of the potential of corpora. These researchers have been interested in many different aspects of language, and so corpus tools have become more diverse.

Some leading corpus tools have been designed around the needs of a particular institution, project, and/or corpus or corpora, and are tailored for working well in that environment.

Corpus tools can be categorized using the following typology:

- a) *Computer-based (stand-alone) tools vs. online tools*: Some tools work as stand-alone software that requires that the tool and the corpus are stored on the user's computer. Leading players here are WordSmith Tools and MonoConc Pro, both of which have been widely and successfully used in teaching. WordSmith and MonoConc Pro are both commercial projects: a free alternative that works in a similar way is Antconc. On the other hand, online corpus tools allow the users to access the corpus, or corpora, from any computer. Examples of online tools include the Sketch Engine (Kilgarriff *et al.* 2004), KorpusDK (developed by the Department for Digital Dictionaries and Text Corpora at the Society for Danish Language and Literature), and Mark Davies's tools at <http://corpus.byu.edu>.
- b) *Corpus-related tools vs. corpus-independent tools*: Some corpus tools can be used only with a particular corpus, most often because they were designed as a part of a specific corpus project or for a specific institution. Examples include SARA¹ (and its newer XML version, XAIRA²) and BNCWeb, two high-specification interfaces designed to access the British National Corpus (BNC), a tool offered by Real Academia Española to access their Spanish reference corpus, Corpus de Referencia del Español Actual (CREA),³ and special groups of corpus-related tools that use the same interface to access several different preloaded corpora (e.g. the tool KorpusDK that is used to access several Danish corpora). A set of corpus tools and software developed by Mark Davies, at Brigham Young University, are used to access leading corpora for Spanish, Portuguese, and American English. His websites are among the most widely-used corpus resources, particularly his Corpus of Contemporary American (COCA) (Davies 2009). Other tools are corpus-independent, which means that they can be used to upload and analyse any corpus. Examples include the Sketch Engine, Corpus WorkBench, WordSmith Tools, MonoConc Pro, and AntConc.
- c) *Prepared corpus vs. Web as corpus*: The majority of corpus tools are used to access a corpus that has been compiled with linguistic research in mind. But the web can be viewed as a vast corpus, with very large quantities of texts for many languages, and lexicographers frequently use it in this way (Kilgarriff and Grefenstette 2003). Google and other Web search engines can be viewed as

¹ <http://www.natcorp.ox.ac.uk/tools/chapter4.xml>.

² <http://xaira.sourceforge.net/>.

³ An online version of the tool is freely accessible, with limitations on searches (e.g. the maximum number of hits displayed is 1,000).

corpus tools: in response to a query, they find and show a number of instances of the query term in use. They are not designed specifically for linguists' purposes, but are often very useful, having access, as they do, to such an enormous source of language data. Some tools have been developed which sit between the search engine and the user, reformatting search results as a concordance and offering options likely to be useful to the linguist. They have been called Web concordancers. One leading system is Webcorp (Kehoe and Renouf 2002).

- d) *Simple tools vs. advanced tools*: Due to the increasing size of corpora, and the increasing number of (different) users, corpus tools have become more and more multifunctional, i.e. they have started offering many different features to assist their users with analysis. The features of corpus tools range from basic features such as concordance, collocation, and keywords, to advanced features, such as word sketches and CQL searches. Most of these features are discussed in more detail in Section 3.4; for more on keywords, see Scott (1997) and Scott and Tribble (2006). Examples of simple corpus tools are AntConc and Mono-Conc Easy. Advanced corpus tools are designed for users who need access to more advanced functionality, e.g. lexicographers. Examples of advanced corpus tools are the Sketch Engine, XAIRA, and KorpusDK.
- e) *Typical users*: The three main types of users of corpus tools are lexicographers, linguistics researchers and students, and language teachers and learners. Different tools have been designed with different target users in mind.

There are numerous corpus tools, but few with the full range of functionality that a lexicographer wants. Of these, most have been in-house developments for particular dictionary or corpus projects. The tools developed within the COBUILD project were used for lexicography at Collins and Oxford University Press throughout the 1980s and 1990s as well as with the 'Bank of English' corpus and WordBanks Online web service (Clear 1987). They set a high standard, and have only recently been decommissioned despite using a 1980s pre-Windows, pre-mouse interface.

The University of Stuttgart's Corpus WorkBench, sometimes also called 'the Stuttgart tools', was another influential early player, establishing in the early 1990s a very fast tool suitable for the largest corpora then available, which could work with sophisticated linguistic mark-up and queries. It is available free for academic use. Both the format it used for preparing a corpus, and the query language it used for querying corpora, have become de facto standards in the field. The group that prepared the corpus worked closely with several German dictionary publishers, so the tools were tested and used in commercial lexicographic settings.

As corpora have grown and Web speeds and connectivity have become more dependable, computer-based corpus tools have become less desirable for large lexicography projects since the corpus and software maintenance must be managed for each user's computer, rather than just once, centrally. Consequently, most lexicographic projects nowadays use online corpus tools that use http protocols (so

users do not have to install any software on their computer) and work with corpora of billions of words.⁴

3.4 Moving on from concordances: the Sketch Engine

The number of features offered by corpus tools is continually increasing, and the development of a new feature often results from an attempt to meet a certain user's need. Recently, many new features have been introduced in the Sketch Engine, a tool aimed particularly at lexicography, and which is available for use with corpora of all languages, types, and sizes. Since its inception the Sketch Engine has had a steady programme of adding functionality according to lexicographers' and corpus linguists' needs.

This section focuses on various features of the Sketch Engine, with particular attention being paid to the features used extensively by lexicographers. Many features, especially those presented in Section 3.4.1, are found in most corpus tools and should not be considered Sketch Engine specific. It should also be pointed out that while in general each new feature is at first used predominantly by lexicographers, at a later stage they are frequently widely adopted by linguists, educators, and other researchers. The features presented here should therefore not be regarded as simply lexicographic, although some of the most innovative features described in Section 3.4.2, such as sketch differences and the Good Dictionary Examples (GDEX) option, have (so far) mainly been used in dictionary-making.

3.4.1 *Analysing concordance lines*

The concordance, "a collection of the occurrences of a word-form, each in its textual environment" (Sinclair 1991b: 32), is the basic feature of corpus use, and is at the heart of lexicographic analysis. Concordance lines can be shown in the sentence format or in the KWIC (Key Word in Context) format. The KWIC format, preferred in lexicography, shows a line of context for each occurrence of the word, with the word centred (see Figure 3.1). Using the concordance feature, lexicographers can scan the data and quickly get an idea of the patterns of usage of the word, spotting meanings, compounds, etc.

The problem with reading raw concordance data is that it can be very time-consuming for lexicographers to gather all the required information on the item

⁴ Recently, lexicographers have become interested in the potential of the World Wide Web for their data analysis, and consequently also in web concordancers. However, web concordancers rely heavily on search engines. This is problematic in various ways, for example there is a limit (for Google, 1,000) on the number of hits the user can access for any search, the corpus lines are sorted according to the search engine's ranking criteria, etc. There are also those who question the lexicographic potential of the web due to its constantly changing size and content. The debate is still continuing, but considering that the web makes so many documents so easily available, it would be a shame not to utilize such a resource.

Corpus: British National Corpus		
Hits: 20		
J3H	rather than be dumped in landfill sites, argues	a report by Britain's Royal Commission
J15	inelastic. </p><p> At the same time monetarists argue	that physical goods are a relatively close
HLD	group of younger" doves". This group also argued	for measures of economic liberalization
HL2	occasion, however, Kaifu had resisted the move, arguing	that Cabinet stability and continuity were
HHW	consultation period on local government funding, we argued	our case on local income tax and pursued
HXT	to misplaced attempts at therapy. Jennett argues	: </p><p> Much of the debate about therapeutic
HNM	addition, it is worth noting that Markowitz has argued	that the inconclusive nature of the empirical
AD9	them were hovering in the reception area, arguing	about what to do. Leila's first responsibility
ADX	in space research and travel. The critics argued	that the space probes were a useless waste
ANF	German army on Paris. They quarrelled and argued	about everything: spiritualism, art, philosophy
A65	society as a whole, and, Marx and Engels argue	, it therefore has to hide the exploitation
BML	qualities that some myopic adults would argue	were the unique contribution of 'Literature
G20	discussed more fully in Chapter 7 where it is argued	, with no claim to originality, that community
EB7	have felt obliged to provide defences: so I argued	. In later seasons, it became evident that
KGR	money. I never moan about hardship. I was arguing	about the principle, that er, I mean, the
KRT	would do, that's what we will do. And third argued	Mr Kinnock, they should negotiate entry
CS1	in the early 1970s became - and we would argue	, has continued to be deliberately titillating
C53	collective action (discussed on pp. 159-63). They argue	that workers are individuals with irreducibly
CTK	to the job too. The problem then, as now, argues	Mace, is that 'Unix systems are developed
CB1	doing Y. Every choice of means, however well argued	, proves groundless with the discrediting

FIGURE 3.1 Concordance lines: the KWIC format.

being analysed. Lexicographers may also want to focus on a particular pattern found in the concordance, group similar concordances together, etc. It is therefore useful for lexicographers to have available additional features (such as 'sorting', 'sampling', and 'filtering') that help manipulate the concordance output and give some statistical information on it.

Sorting the concordance lines will often bring a number of instances of the same pattern together, making it easier for the lexicographer to spot the patterns. The most typical sorts are by the first word to the left, first word to the right, and by the node word. Sorting by the node word can be useful for lexicographers working with highly inflected languages where lemmas often have many different word forms. The type of sorting that yields the most useful results depends on the grammatical characteristics of the word. For English nouns, for example, sorting by the first word on the left will normally highlight the relevant patterns involving adjective modifiers and verbs of which the noun is object, whereas sorting on the right will show verbs of which the noun is subject. In Figure 3.2, where the concordance lines are sorted by the first word to the right, it is much easier to spot recurring patterns such as *argue about*, *argue for*, and *argue that*, compared to the sort in Figure 3.1. Other types of sorting include sorting according to the second, third, etc. word to the right or to the left of the node word, and more complex options such as sorting according to word endings.

There are two more types of sorting that differ from those mentioned so far, namely sorting according to the meaning of the node word, and sorting according to

Corpus: British National Corpus	
Hits: 20	
CS1	in the early 1970s became - and we would argue , has continued to be deliberately titillating
A65	society as a whole, and, Marx and Engels argue , it therefore has to hide the exploitation
CB1	doing Y. Every choice of means, however well argued , proves groundless with the discrediting
G20	discussed more fully in Chapter 7 where it is argued , with no claim to originality, that community
EB7	have felt obliged to provide defences: so I argued . In later seasons, it became evident that
HXT	to misplaced attempts at therapy. Jennett argues : </p><p> Much of the debate about therapeutic
J3H	rather than be dumped in landfill sites, argues a report by Britain's Royal Commission
ANF	German army on Paris. They quarrelled and argued about everything: spiritualism, art, philosophy
KGR	money. I never moan about hardship. I was arguing about the principle, that er, I mean, the
AD9	them were hovering in the reception area, arguing about what to do. Leila's first responsibility
HLD	group of younger" doves". This group also argued for measures of economic liberalization
CTK	to the job too. The problem then, as now, argues Mace, is that "Unix systems are developed
KRT	would do, that's what we will do. And third argued Mr Kinnock, they should negotiate entry
HHW	consultation period on local government funding, we argued our case on local income tax and pursued
HL2	occasion, however, Kaifu had resisted the move, arguing that Cabinet stability and continuity were
J15	inelastic. </p><p> At the same time monetarists argue that physical goods are a relatively close
HNM	addition, it is worth noting that Markowitz has argued that the inconclusive nature of the empirical
ADX	in space research and travel. The critics argued that the space probes were a useless waste
CS3	collective action (discussed on pp. 159-63). They argue that workers are individuals with irreducibly
BML	qualities that some myopic adults would argue were the unique contribution of 'Literature

FIGURE 3.2 Sorting concordance lines.

how good a candidate for a dictionary example the concordance line is. Both types of sort require an additional preliminary stage: the former requires manual annotation of the concordance lines of the word (see Section 3.5.4), whereas the latter requires the computation of the good example score (see Section 3.4.3).

Sampling is useful as there will frequently be too many instances for the lexicographer to inspect them all. It is misleading just to look at the first instances as they will all come from the first part of the corpus: if the lexicographer is working on the entry for *language*, and there are a few texts about *language development* near the beginning of the corpus, then it is all too likely that the lexicographer who just works straight through the corpus will get an inflated view of the importance of the term, while missing others. The sampling feature in the corpus tool allows the lexicographer to take a manageable-sized sample of randomly selected concordance lines from the whole corpus.

Filtering allows the lexicographer to focus on a particular pattern of use (a positive filter), or to set aside the patterns that have been accounted for in order to focus on the residue (a negative filter). For example, if the lexicographer spots *local authority* as a recurrent pattern of the word *authority*, he or she can first focus on that pattern by using either the positive filter (searching for all the concordances where *local* occurs one word to the left of *authority*), or performing a search for the phrase *local authority*, and then continue the analysis by excluding the pattern *local authority* from the concordance output with the negative filter.

Search by sub-corpora can be considered as a type of filtering as it can be used to limit the analysis of the pattern to part of the corpus. Many words have different

doc.year	Freq	Rel [%]
p/n 2000	111	15.8
p/n 2001	296	32.0
p/n 2002	899	69.0
p/n 2003	1580	102.5
p/n 2004	2557	129.8
p/n 2005	2811	194.2
p/n 2009	18	4.7

FIGURE 3.3 Frequency distribution of the lemma *random* in the OEC blog sub-corpus

meanings and patterns of use in different varieties of language, and the lexicographer needs to be able to explore this kind of variation. A vivid example is the English noun *bond*: in finance texts it means a kind of finance, as in *treasury bonds*, *Government bonds*, *junk bonds*; in chemistry, a connection between atoms and molecules as in *hydrogen bonds*, *chemical bonds*, *peptide bonds*; and in psychology, a link between people, as in *strengthening*, *developing*, *forging bonds*.

Frequency analyses are often useful to lexicographers. For example, analysing the word *random* shows the importance of combining analysis by text type and change over time using the Sketch Engine's frequency feature. The goal here was to explore the hypothesis that *random* has recently added an informal use to its traditional, formal, and scientific one, as in:

- (1) Last was our drama but unfortunately our original drama went down the drain way down so Iffy came up with one very **random** drama involving me doing nothing but just sit down and say my one and only line "Wha?" and she just yell at me coz she was pissed off of something.

The Oxford English Corpus (OEC), containing over two billion words, includes a large component of blog material, so the blog sub-corpus could be used to explore the new pattern of use. Also, each text has the year in which it was written or spoken in its metadata. Figure 3.3 shows the frequency distribution of the word *random* in blogs over the period 2001–2005.

Sometimes the lexicographer cannot decipher the meaning of the word being analysed because the concordance line does not provide enough information. For example, for the concordance line for *random* offered in example (1), the default Sketch Engine context size of forty characters (excluding spaces) to the left and to the right of the searched word⁵ does not provide enough information to get an idea of the meaning of *random*, as shown in example (2):

- (2) drain way down so Iffy came up with one very random drama involving me doing nothing but just sit

⁵ Only whole words within the context size are shown in the concordance.

It is thus useful to have quick access to more context, which in most corpus tools can be accessed by clicking on a concordance line.

3.4.2 From collocation to word sketches

Since COBUILD, lexicographers have been using KWIC concordances as their primary tool for finding out how words behave. But corpora continue to grow. This is good because the more data we have, the better placed we are to present a complete and accurate account of a word's behaviour. It does, however, present challenges. Given fifty corpus occurrences of a word, the lexicographer can simply read them. If there are five hundred, reading them all is still a possibility but might take longer than an editorial schedule permits. Where there are five thousand, it is no longer viable. Having more data is good—but the data then needs summarizing.

One way of summarizing the data is to list the words that are found in close proximity of the word that is the subject of analysis with a frequency far greater than chance, i.e. its collocations (Atkins and Rundell 2008). The sub-field of collocation statistics began with a paper by Church and Hanks (1989) who proposed a measure called 'mutual information' (MI), from information theory, as an automatic way of finding a word's collocations: their thesis was that pairs of words with high mutual information for each other would usually be collocations. The approach generated a good deal of interest among lexicographers, and many corpus tools now provide functionality for identifying salient collocates along these lines.⁶

One flaw in the original work is that MI emphasizes rare words (and an ad hoc frequency threshold has to be imposed or the list would be dominated by very rare items). This problem can be solved by changing the statistic, and a number of proposals have been made. Evert and Krenn (2001) evaluated a range of proposals (from a linguist's rather than a lexicographer's perspective). Statistics for measuring collocation, in addition to MI, include MI₃, the log-likelihood ratio, and the Dice coefficient (for a full account see Manning and Schütze, 1999, Chapter 5). Another, more recently proposed collocation statistic is logDice (Rychly 2008).

Tables 3.1 to 3.4 (each containing the top fifteen collocate candidates of the verb *save* in the OEC corpus, in the window of five tokens to the left and five tokens to the right, ordered according to MI, MI₃, log-likelihood, and logDice scores respectively), offer a good demonstration of the differences between the various statistics. Collocate candidates offered by MI are very rare, and not at all useful to lexicographers. Better collocate candidates, many of them the same, are offered by MI₃ and log-likelihood; however in this case very frequent functional words dominate the list. Even more useful candidate collocates are provided by logDice, from which the lexicographer can already get an idea of a few meanings of the verb *save*, for example 'use less of or

⁶ In our terminology, a *collocation* comprises *node word* + *collocate(s)*, in particular grammatical relations.

invest' (*money, million*), 'prevent from harm' (*life*), and 'store' (*file*). Collocation can thus be used not only to describe word meanings (Sinclair 2004), but also to distinguish between them (see also Hoey 2005). A list of collocates, representing an automatic summary of the corpus data, is therefore very useful for the lexicographer.

As shown in Tables 3.1 to 3.4, collocates are normally provided in the form of a list. Another way of displaying collocates, available in the COBUILD tools (called 'picture') and WordSmith Tools (the Patterns view), is to list collocates by frequency or by score of whichever statistical measure is used,⁷ in each position between the selected span (see Figure 3.4). The information in this display needs to be read vertically and not horizontally. The drawbacks of this display are that it gives the user a lot of information to wade through, and fails to merge information about the same word occurring in different positions.

Word Sketches Collocation finding as described above is grammatically blind. It considers only proximity. However, lexicographically interesting collocates are, in most cases, words occurring in a particular grammatical relation to the node word. For example, an examination of the concordance of the top collocates in Table 3.4 shows that a number of them occur as the object of the verb (e.g. *life, money, energy,*

TABLE 3.1. Top fifteen collocates of the verb *save* (ordered by MI score)

Lemma	freq	MI
BuyerZone.com	7	13.19
ac);	5	13.19
count-prescription	5	13.19
Christ-A-Thon	7	13.19
Teldar	6	12.61
Re:What	26	12.55
Redjeson	5	12.51
INFOPACKETS30	3	12.46
other-I	4	12.39
SetInfo	4	12.39
Ctrl-W	9	12.36
God	18	12.23
Walnuttree	3	12.19
Hausteen	5	12.19
MWhs	3	12.19

⁷ WordSmith Tools lists collocates in the 'Patterns' view by frequency only.

TABLE 3.2. Top fifteen collocates of the verb *save* (ordered by MI₃ score)

lemma	freq	MI ₃
to	99846	37.29
life	27606	36.98
.	102829	36.65
money	19901	36.51
the	106241	36.39
,	86327	35.69
be	70859	35.25
and	62030	35.22
from	28399	34.44
a	47129	34.14
of	41271	33.38
have	29869	33.21
you	20610	33.02
that	29260	33.01
for	25291	32.90

TABLE 3.3. Top fifteen collocates of the verb *save* (ordered by log likelihood score)

lemma	freq	log likelihood
to	99846	417952.13
.	102829	333836.91
the	106241	297431.94
life	27606	234592.45
,	86327	222779.39
and	62030	192235.46
be	70859	190628.16
money	19901	181861.45
from	28399	139301.25
a	47129	126211.75
have	29869	92837.93
of	41271	90602.61
you	20610	86952.62
that	29260	85631.78
for	25291	83634.16

TABLE 3.4. Top fifteen collocates of the verb *save* (ordered by logDice score)

lemma	freq	logDice
money	19901	9.34
Life	27606	9.05
save	2976	7.52
energy	2648	7.37
million	4742	7.17
dollar	1847	7.16
File	2147	7.14
Try	6380	7.11
\$	6193	7.07
could	11904	7.05
effort	2844	7.05
◆	2583	7.01
retirement	1181	6.94
planet	1194	6.91
thousand	1894	6.89

N	L5	L4	L3	L2	L1	Centre	R1	R2	R3	R4	R5
1	THE	THE	THE	OF	THE	LOCAL	AND	AND	THE	THE	THE
2	OF	OF	OF	TO	OF		GOVERNMENT	OF	AND	AND	OF
3	AND	AND	TO	IN	A		AUTHORITIES	IN	IN	OF	AND
4	TO	TO	AND	AND	AND		LEVEL	THE	A	TO	TO
5	IN	IN	IN	THE	TO		AUTHORITY	TO	TO	IN	IN
6	A	A	A	AT	IN		COMMUNITY	IS	OF	A	A
7	THAT	IS	THAT	BY	FOR		GOVERNMENTS	FOR	IS	IS	IS
8	IS	THAT	IS	WITH	BY		COMMUNITIES	AS	AS	AS	THAT
9	FOR	AS	BE	ON	WITH		PEOPLE	ARE	ARE	THAT	AS
10	AS	FOR	BY	FOR	ON		MARKET	THAT	THAT	FOR	FOR
11	BY	BE	AS	THAT	THAT		OR	OR	FOR	WITH	WITH
12	WITH	WITH	FOR	FROM	FROM		CONDITIONS	WITH	THIS	BY	BY
13	ARE	BY	WITH	A	THEIR		RESIDENTS	WERE	WITH	ARE	BE
14	ON	ARE	ARE	IS	AS		CONTEXT	AT	BE	BE	THIS
15	BE	WAS	ON	AS	OR		POLITICAL	WAS	WHICH	ON	ARE
16	LOCAL	WERE	WERE	STATE	BETWEEN		POPULATION	REGIONAL	BY	THIS	LOCAL
17	WAS	ON	NOT	GLOBAL	ITS		KNOWLEDGE	BUT	NOT	LOCAL	ON
18	THIS	LOCAL	LOCAL	NATIONAL	OTHER		POWER	THIS	AT	HAVE	WAS
19	FROM	AN	GLOBAL	OR	AT		OFFICIALS	HAVE	ON	IT	FROM
20	IT	FROM	THIS	REGIONAL	THESE		AREA	ON	WERE	NOT	IT

FIGURE 3.4 WordSmith Tools' Picture view for *local* in the Corpus of Academic Journal Articles (Kosem, 2010).

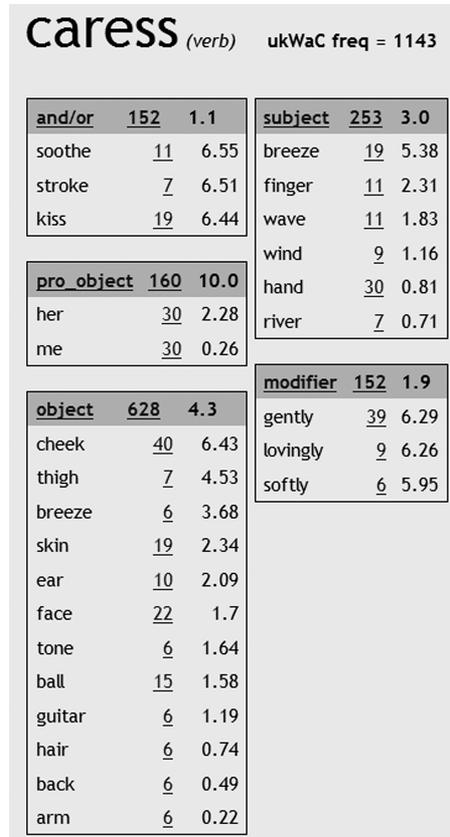


FIGURE 3.5 Word sketch of the verb *caress* in the ukWaC corpus.

file, planet). In order to identify grammatical relations between words, the corpus has to be parsed.

Corpus features combining collocation and grammar are Sketch Engine’s ‘word sketches’.⁸ Word sketches are defined as “one-page automatic, corpus-based summaries of a word’s grammatical and collocational behaviour” (Kilgarriff *et al.* 2004: 105). Figure 3.5 shows the word sketch for the verb *caress* in the ukWaC corpus (Ferraresi *et al.* 2008), which offers the lexicographer the most salient collocates that occur as the object, subject, modifier, or in the ‘and/or’ relation to *caress*.

Word sketches were first used for the Macmillan English Dictionary (Kilgarriff and Rundell 2002). Atkins and Rundell (2008: 107–11) saw word sketches as a type of lexical profiling, and they have become the preferred starting point for lexicographers analysing complex headwords.

⁸ A similar feature is also provided by the DeepDict Lexifier tool (Bick 2009).

For word sketches to be built, the system must be told what the grammatical relations are for the language, and where in the corpus they are instantiated. There are two ways to do this. The input corpus may already be parsed, with grammatical relations given in the input corpus. However, such a corpus is only occasionally available. The other way is to define the grammatical relations, and parse the corpus, within the tool. To do this, the input corpus must be POS-tagged. Then each grammatical relation is defined as a regular expression over POS-tags, using corpus query language. The CQL expressions are used to parse the corpus, giving a database of *-tuples* such as *<subject, caress, breeze, 14566778>* where *subject* is a grammatical relation holding between the verb *caress* and the noun *breeze* at corpus reference point (for *caress*) 14566778. Word sketches are generated at run-time from the *-tuples* database. Parsing is done at compile time, and the results are stored, so users do not have to wait. The accuracy of the process is discussed and evaluated in Kilgarriff *et al.* (2010a).

A list of collocates is sometimes directly transferred, by the lexicographer, from the corpus tool to the dictionary entry, as shown in Figure 3.6. In the Macmillan English Dictionary Online, the box ‘Collocations: result’, for example, lists verbs that take *result*, in dictionary sense 3, as an object, as identified within the Sketch Engine.

Thesaurus The thesaurus feature provides a list of “nearest neighbours” (Kilgarriff *et al.* 2004: 113) for the word. Nearest neighbours are those that ‘share most collocates’ with their node word: if we have encountered *<subject, caress, breeze>* and *<subject, caress, wind>* then *breeze* and *wind* share a collocate: the process of generating the thesaurus is one of finding, for each word, which other words it shares collocates with (and weighting the shared items; see Rychly and Kilgarriff 2007). The thesaurus provides a lexicographer with a list of potential (near)synonyms (and, in some cases, antonyms). For example, the thesaurus output of the ten nearest neighbours of the adjective *handsome* (1,578 occurrences in the BNC), as shown in Table 3.5, contains

3 [COUNTABLE] [OFTEN PLURAL] a piece of information that is obtained by examining, studying, or calculating something

Our results show that an effective vaccine is feasible.

result of: *The results of the survey will be published shortly.*

T Thesaurus entry for this meaning of result

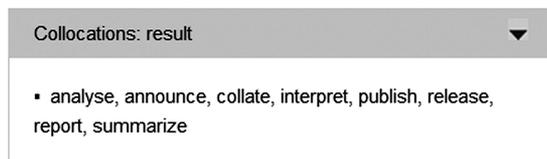


FIGURE 3.6 Macmillan English Dictionary’s online entry for the noun *result*, collocation box under sense 3.

TABLE 3.5. Ten nearest neighbours for *handsome* offered by Thesaurus

lemma	similarity score
good-looking	0.27
elegant	0.24
charming	0.24
beautiful	0.23
pretty	0.23
tall	0.22
lovely	0.20
attractive	0.20
clever	0.19
slim	0.19

several synonym candidates, such as *good-looking*, *beautiful*, *pretty*, *lovely*, and *attractive*.

Sketchdiffs Sketch differences or ‘sketchdiffs’ compare word sketches for the two words, showing the collocations that they have in common and those they do not. Figure 3.7 shows the sketch difference for the adjectives *handsome* and *attractive* in ukWaC. The collocates *particularly*, *quite*, *extremely*, *so*, *very*, *really*, and *as* (highlighted in shades of red in the Sketch Engine) are more typical modifiers of *attractive*; *strikingly* and *devastatingly* (highlighted in green in the Sketch Engine), are more typical of *handsome*, while the remaining collocates in this relation show similar salience with both *handsome* and *attractive*.

The thesaurus and sketchdiff are linked. Clicking on a lemma in a thesaurus entry automatically opens the sketch difference comparing the original lemma with the one found in the thesaurus entry. Thesaurus and sketchdiffs were used extensively in compiling the *Oxford Learner’s Thesaurus—a dictionary of synonyms* (Lea 2008).

3.4.3 Good Dictionary EXamples (GDEX)

Good dictionary examples are hard to find; lexicographers have often invented, rather than found, them but that runs the risk of accidentally failing to provide a natural context for the expression being illustrated (see Hanks, this volume). Sketch Engine’s GDEX attempts to automatically sort the sentences in a concordance according to how likely they are to be good dictionary examples (Kilgarriff *et al.* 2008). GDEX operates as an option for sorting a concordance: when it is on, the ‘best’ examples will be the ones that the user sees first, at the top of the concordance. GDEX scores sentences using heuristics for readability and informativeness. Readability

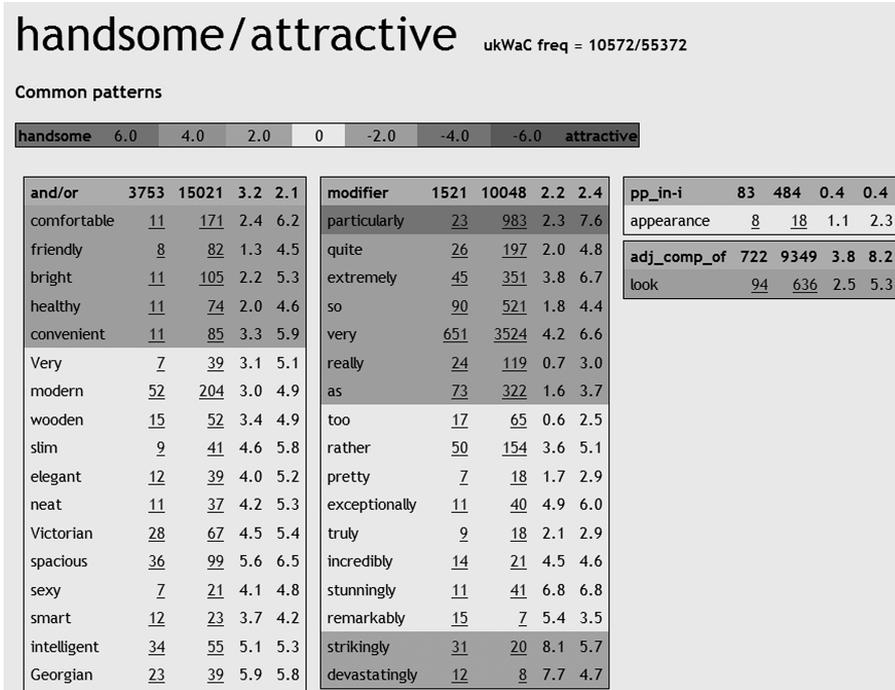


FIGURE 3.7 Sketch difference for the adjectives *handsome* and *attractive*.

heuristics include sentence length and average word length, and penalize sentences with infrequent words, more than one or two non-a-z characters, or anaphora. Informativeness heuristics include favouring sentences containing words that are frequently found in the vicinity of the expression: it is likely that they are typical collocates of the expression. GDEX was first used in the preparation of an electronic version of the *Macmillan English Dictionary* (2nd edition, 2007).

GDEX was designed for English, so several heuristics are specific to the English language (e.g. a classifier that penalizes multi-sense words, based on the number of Wordnet synsets the word belongs to; see also Kosem *et al.* 2011) or were included with the needs of a specific group of dictionary users in mind (e.g. advanced learners of English). The usefulness of GDEX for other languages is more limited. This has been confirmed by experience when using it to produce a new lexical database of Slovene in the ‘Communication in Slovene’ project,⁹ where the examples offered first by GDEX were rarely found to be useful to lexicographers. Infrastructure for customizing GDEX has recently been completed, and Slovene and other GDEXes are currently under development.

⁹ www.slovenscina.eu.

3.4.4 *Why we still need lexicographers*

No matter how many features are used to summarize the data, the lexicographer still needs to critically review the summary to determine the meaning of each word. Concordances should always be available to check the validity of results: there are many stages in the process where anomalies and errors might arise, from the source data, or in its preparation, lemmatization, or parsing. It ought to be easy for the lexicographer to check the data underlying an analysis, to check for instances where the analysis does not immediately tally with their intuition.

One recurring area of difficulty, in all the languages for which we have been involved in lexicography—two recent examples being Polish and Estonian—is participles/gerunds. In English, most *-ed* forms can be verb past tenses or past participles, or adjectival, and *-ing* forms can be verbal, adjective, or gerunds; comparable distinctions apply to most European languages. In theory, it may be possible to distinguish the form (verbal participle) from the function (verbal, adjectival, or nominal) but the theory still leaves the lexicographer with a judgement to make: should the *-ing* form get a noun entry, or should the *-ed* form get an adjective entry? POS-taggers are stuck with the same quandary: Where they encounter an *-ing* form, should they treat it as part of the verb lemma, as an adjective, or as a noun?

The problem has two parts: some syntactic contexts unambiguously reveal the function (*the painting is beautiful; he was painting the wall*) but many do not (*I like painting; the painting school*). But this is only the first problem. The second problem is that some gerunds and participial adjectives are lexicalized, deserving their own entry in the dictionary, and others are not: thus we can have *the manoeuvring is beautiful* and there is no question that *manoeuvring* is functioning as a noun, but there is also no question that it is not lexicalized and does not need its own dictionary entry. The upshot is that many word sketches contain verb lemmas which would ideally not be there, because they are the result of lemmatization of adjectival participles and gerunds, which should have been treated as adjective and noun lemmas in their own right.

3.5 Developing corpus tools to meet lexicographers' needs

Lexicographers are demanding corpus users; they soon come to understand the potential of corpora, and expect a wide range of features. Initially, not a great deal of thought was given to the actual look and user-friendliness of the interface—functionality and speed were considered more important. But with the regular use of corpus tools, more time has to be spent on devising interfaces that are friendly to the lexicographers who use them on a daily basis. Training lexicographers in how to analyse data is already time-consuming, and a user-friendly interface helps them focus on the analysis.

3.5.1 *User-friendliness*

A comparison of older tools with modern ones testifies to progress in user-friendliness. Conducting searches no longer requires typing in complex commands. Corpus tools have become more Google-like, where the users write the search term in the box, specify the search (often using a drop-down menu) if they want to, and promptly get what they want (see Verlinde and Peeters, this volume).

Another difference is in the use of colour. Black and white are no longer the only options, and modern tools use colour highlighting to aid navigation in the output and/or to separate different types of information. For example, sketchdiff uses green for collocates more strongly associated with the first lemma, and red for those more strongly associated with the second, with the strength of colour indicating the strength of the tendency.

Some corpus tools also offer graphical representations of numerical data. Graphical representation often helps lexicographers quickly identify usage-related information, for example an increase or decrease in the use of a word or phrase over a period of time (see Figure 3.3), predominant use of the word in a certain domain, register, etc., typical use of the word in a specific form (e.g. when a noun occurs mainly in the plural form), and so forth.

Lexicographers have different preferences and use different equipment, such as computer screens of different sizes, so customizability is part of user-friendliness. An example of a basic customizable feature is adjustable font size. With online corpus tools, font size can also be changed in the settings of the Internet browser.

Many corpus tools also offer the option to change the concordance output, in terms of how much data is displayed (e.g. the number of concordance lines per page, the amount of context shown), and which type of data is displayed, e.g. attributes of the searched item (word form, lemma, POS-tag, etc.) and structure tags (document, paragraph, and sentence markers). A form of customization requiring deeper understanding is control of the word sketches by changing parameters such as the minimum frequency of the collocate in the corpus, or the maximum number of displayed items. The Sketch Engine also provides 'more data' and 'less data' buttons to make the word sketches bigger or smaller.

Recent developments relating to character sets have been a great boon for corpus developers and lexicographers. Not so long ago, the rendition of the character set for each new language, particularly non-Latin ones, would have been a large and time-consuming task. Now, with the Unicode standards and associated developments in character encoding methods, operating systems, and browsers, these problems have largely been solved, and well-engineered modern corpus tools can work with any of the world's writing systems with very little extra effort. The Sketch Engine, for example, correctly displays corpora for Arabic, Chinese, Greek, Hindi, Japanese, Korean, Russian, Thai, and Vietnamese, amongst others.

A related issue is the interface language. Chinese lexicographers working on Chinese, or Danes working on Danish, do not want an English-language interface. This has doubtless contributed to various institutions developing their own tools. The Sketch Engine is localizable, and the interface is currently available in Chinese, Czech, English, French, and Irish.

3.5.2 *Integration of features*

Because the number of features offered by corpus tools is increasing, it is useful and time-saving if the features are integrated. The lexicographer looking at a list of collocates is likely to want to check their concordance lines. If the collocation and the concordance features are integrated, the user can move between the two with a mouse-click.

Another time-saving technique that may help lexicographers in the future would be to combine two features into one. An example of this can be found in the online tool for Gigafida, a 1.15-billion-word corpus of Slovene (which targets lay users and not lexicographers), where the filters are offered in the menu to the left of the concordance output (see Figure 3.8) and enable the user to filter concordance lines by basic forms, text type, source, and other categories. They also provide frequency information for each available category in the filter (filter categories with zero concordance lines are not shown), ordering categories by frequency.

3.5.3 *Integration of tools*

A corpus tool is not the only piece of software a lexicographer needs to master. There is always at least one other tool, the dictionary writing system (see Abel, this volume). Lexicographic work often involves transferring corpus data to the dictionary database, and time and effort can be saved if the transfer is efficient. Copy-and-paste is possible in some cases, but the information often needs to be in a specific format (normally XML) for the dictionary writing system to read it. This issue is addressed by the Sketch Engine's 'TickBox Lexicography'.

TickBox Lexicography (TBL) allows lexicographers to select collocates from the word sketch, select examples of collocates from a list of (good) candidates, and export the selected examples into the dictionary writing system (see Figures 3.9 and 3.10). An XML template, customized to the requirements of the dictionary being prepared, is needed for the data to be exported in the format compatible with the dictionary writing system. Lexicographers do not need to think about XML: from their perspective, it is a simple matter of copy-and-paste.

Another option is to combine a corpus tool and a dictionary writing system in a single program, so that lexicographers use the same interface to search the corpus and write dictionary entries. Such software is already available, namely the TLex

Gigafida | [Concordance](#) | [Collocations](#)

You are using [simple search](#) [Advanced search](#)

[Search history](#) ▼ 1 2 3 4 5 6 7 8 9 11

Showing 1-20 of 103,552 concordances

Basic forms

- ▶ [ujeti \(82,443\)](#)
- ▶ [ujet \(21,109\)](#)

Text type

- ▶ [Newspapers \(51,300\)](#)
- ▶ [Magazines \(30,766\)](#)
- ▶ [Internet \(12,327\)](#)
- ▶ [Fiction \(4,677\)](#)
- ▶ [Non-fiction \(3,942\)](#)
- ▶ [More](#)

Source

- ▶ [Other \(25,134\)](#)
- ▶ [Dnevnik \(18,007\)](#)
- ▶ [Delo \(14,804\)](#)
- ▶ [Adria Media \(6,250\)](#)
- ▶ [Ekipa \(5,829\)](#)
- ▶ [More](#)

▼

nelagodno preusmeri drugam. Očiten simptom nesvobodnega stanja, znamenje ujetega
izšla knjiga Stoletje pozdravov, ki jo sestavljajo utrinki, ujeti na i
povečini za profesionalce, ki se ne dajo z lahkoto ujeti tudi
več idej, da bi njihove državljane, ki so ujeti v Az
Ko je uganil pravega, je prevzel mesto delilca, ujeti igra
bosta sonce in zemlja spet znašla v vedrem klepetu. Ujeti sm
smo pač v ta začarani krog življenja in odhajanja. Ujeti v n:
svojega domačega kraja in prežvela. Dva kita sta ostala ujeta v le
Zrelost raste iz dostojanstva. Ne pusti se vkleniti in ujeti .
vse mogoče načine škodili. Knoblehar in njegovi kolegi so ujete su
Živilska industrija ujeta v p
za kulturno obnašanje ali pa za upoštevanje čustev drugih. Ujeti ste

FIGURE 3.8 Available filters to query the Gigafida corpus.

Dictionary Production System (Joffe and de Schryver 2004i), as reviewed by Abel (this volume).

3.5.4 Project customization

A certain feature often needs to be customized to the requirements of a particular dictionary project. A critical concern at the Institute for Dutch Lexicology (INL) was bibliographical references: in the ANW (a Dictionary of Contemporary Dutch, in preparation), each example sentence was accompanied by its bibliographical details. These were available to the corpus system. However, the time it took to type, or copy-and-paste, all those details into the appropriate fields in the dictionary writing system was severely limiting the numbers of examples the lexicographers were using, and putting the whole schedule of the project at risk. The Sketch Engine team was able to customize the TBL machinery to provide a ‘special copy-and-paste’ option which automatically gathered together the bibliographic data for a sentence that the

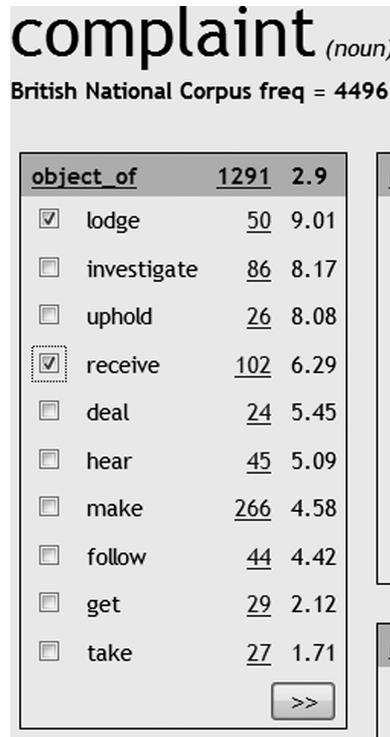


FIGURE 3.9 TickBox Lexicography: selecting collocates.

lexicographer had selected, and, on pasting, inserted the ‘example’, ‘author’, ‘title’, ‘year’, and ‘publisher’ into the appropriate fields of the dictionary writing system.

Implementing a customized version of TBL does not require any changes to the corpus interface, but adding a new feature does. This is evident in the *Pattern Dictionary of English Verbs* (Hanks and Pustejovsky 2005, Hanks 2008, this volume) where lexicographers are using an enhanced version of the Sketch Engine, designed specifically for the project to annotate concordance lines of the verb with the number of the associated pattern in the database entry (Figure 3.11). In addition, the dictionary database is linked to the Sketch Engine, so that the users can view all the concordance lines associated with a pattern with a single click.

The relationship between the lexicographers working on a dictionary project, and the developers of the corpus tool used in the project is cyclical. Lexicographers benefit from the functionality of the corpus tools, and, since they are regular users of the tool and most of its features, provide feedback for the developers. This often results in further improvements to the tool, which again benefits lexicographers (as well as other users of the tool).

Tickbox Lexicography - Select Examples

Lemma: **complaint**
 Gramrel: **object_of**
 Template: **vanilla**

lodge

She agrees to go the city council and lodge a complaint .
 And they upheld a complaint lodged by a viewer.
 The couple have lodged an official complaint against Gloucestershire police.

receive

Abbey says it is still receiving complaints .
 So far, it has received more than 2,000 complaints .
 It received more than 100 complaints this year.

FIGURE 3.10 TickBox Lexicography: selecting examples for export.

Annotating: **abate-v** Info Sort Finish New pattern: Add Number globally:

Page 1 of 2 Next | Last

A2X	France for a month showed some signs of abating	<input checked="" type="checkbox"/>	yesterday as prison officers agreed to
A3S	promised a 'soft landing' in which inflation abates	<input checked="" type="checkbox"/>	but growth continues moderately. </p><p>
A7H	obsession with her was showing no signs of abating	<input checked="" type="checkbox"/>	. The media simply could not get enough.
A7Y	lay an information alleging the failure to abate	<input checked="" type="checkbox"/>	a statutory nuisance without first giving
A8K	according to the unions, shows no sign of abating	<input checked="" type="checkbox"/>	. With no overtime being worked, even ambulance
A8X	<p> The 12-week dispute showed no signs of abating	<input checked="" type="checkbox"/>	yesterday. Crews in Greater Manchester
A9J	years on, the Intifada shows little sign of abating	<input checked="" type="checkbox"/>	. It is a cliché to say that it has become
A9W	Britain and the epidemic showed no sign of abating	<input checked="" type="checkbox"/>	. </p><p> The Department of Health said tests
AAA	wage settlements -- has shown no signs of abating	<input checked="" type="checkbox"/>	in recent months, according to the Confederation
AB6	Energy efficiency may be the quickest way to abate	<input checked="" type="checkbox"/>	emissions of carbon dioxide but it is hard
ABE	activists had been arrested and street violence abated	<input checked="" type="checkbox"/>	, the ruling party stopped besieging itself
ABJ	upper house of parliament -- has at last abated	<input checked="" type="checkbox"/>	. If so, this is a good time to tackle tricky
ACA	to secure a safe supply. The scourge had abated	<input checked="" type="checkbox"/>	, but psychological damage had been done
AHJ	. Inflation, such as it is, continues to abate	<input checked="" type="checkbox"/>	. The government's core rate of inflation

FIGURE 3.11 Corpus Pattern Analysis' version of the Sketch Engine: annotated concordance lines for the verb *abate*.

3.6 Conclusion

People writing dictionaries have a greater and more pressing need for a corpus than most other linguists, and have long been in the forefront of corpus development. From the Bank of English corpus (used in the COBUILD project), to the BNC, the largest corpora were built and used for lexicographic (as well as for natural language processing) purposes. Building large corpora is no longer problematic as many texts are readily available in electronic form on the Internet. But precisely because corpora have got larger and larger, it has become more important than ever that lexicographers have corpus tools with summarization features at their disposal.

This chapter has shown that the functionality and user-friendliness of corpus tools have improved considerably since they were first used in dictionary projects. Today's corpus tools are faster and more diverse on the one hand, but easier to use on the other. The needs of lexicographers have also prompted the creation of features such as TickBox Lexicography, which ease the exporting of corpus information into the dictionary writing system. Lexicographically-oriented features are also being used by linguists, teachers, and others, which indicates that the distinction between lexicographic corpus tools and linguistic corpus tools is blurred.

There is, however, still more work to be done in terms of making corpus tools as useful to lexicographers as possible. This includes coming up with more features that bridge the gap between raw corpus data and the dictionary. One strategy is to establish closer links between a corpus tool and a dictionary writing system, incorporating more features like TickBox Lexicography which support seamless data transfer. Currently, most of the focus is on examples; definitions are written in the dictionary writing system, which means that the lexicographer may need to switch between corpus tool and dictionary writing system quite often. Corpus tools of the future should perhaps offer a more complete solution, e.g. allowing the lexicographer to mark examples, devise a draft definition (in a pop-up window) and any other part of the meaning in the corpus tool, and only then export it into the dictionary entry.

Corpora and associated software do more and more by way of summarizing the information to be found about a word or phrase. A question worth asking then is: Will corpus tools reach a point where they act as dictionaries? The idea does not seem too far-fetched. There is already research showing that definitions of words can be extracted directly from corpora (Pearson 1996, 1998). In addition, GDEX incorporates a feature that helps identify good dictionary examples. Nonetheless, as Rundell and Kilgarriff (2011) point out, providing the users with automatically extracted corpus data, rather than data in a traditional dictionary format, may pose problems for some types of users, for example language learners. The position we take is this: lexicographers are better at preparing brief, user-friendly accounts of a word's meaning and behaviour than automatic tools—but they do not cover everything

(just as no dictionary covers all new and obscure words, specialized uses, contextually appropriate collocations, etc.). Where a user wants to find something out, it is most convenient if the information can be found in a dictionary; but if the dictionary does not meet their needs, then, of course, yes, they should turn to the corpus.

Dictionaries

Rundell, Michael (ed.) (2002). *Macmillan English Dictionary for Advanced Learners*, First Edition. London: Macmillan.

Rundell, Michael (ed.) (2007). *Macmillan English Dictionary for Advanced Learners*, Second Edition. London: Macmillan.