Setting up for Corpus Lexicography

Adam Kilgarriff,[*] Jan Pomikalek,[*] Miloš Jakubíček[*‡], & Pete Whitelock[†]

[*]Lexical Computing Ltd; [‡]Masaryk University, Brno; [†]Oxford University Press

Keywords: *corpora, corpus lexicography, web crawling, dependency parsing.*

Abstract

There are many benefits to using corpora. In order to reap those rewards, how should someone who is setting up a dictionary project proceed? We describe a practical experience of such 'setting up' for a new Portuguese-English, English-Portuguese dictionary being written at Oxford University Press. We focus on the Portuguese side, as OUP did not have Portuguese resources prior to the project. We collected a very large (3.5 billion word) corpus from the web, including removing all unwanted material and duplicates. We then identified the best tools for Portuguese for lemmatizing and parsing, and undertook the very large task of parsing it. We then used the dependency parses, as output by the parser, to create word sketches (one page summaries of a word's grammatical and collocational behavior). We plan to customize an existing system for automatically identifying good candidate dictionary examples, to Portuguese, and add salient information about regional words to the word sketches. All of the data and associated support tools for lexicography are available to the lexicographer in the Sketch Engine corpus query system.

1. Introduction

There are a number of ways in which corpus technology can support lexicography, as described in Rundell & Kilgarriff (2011). It can make it more accurate, more consistent, and faster. But how might those potential benefits pan out in an actual project? If starting from a blank sheet of paper, how should one proceed?

In this paper we describe such an exercise. Oxford University Press is preparing a new Portuguese-English, English-Portuguese dictionary, to have around 40,000 headwords on each side. The work here concerns the new analysis of Portuguese for the Portuguese-source side.

The components of the process are:

- Collect the corpus (section 2)
- Process it with the best available tools for the language (section 3)
- From parser output to corpus system (section 4)
- Finding good examples (section 5)
- Regional variants (section 6)

2. Corpus Collection

Corpora for lexicography should be large and diverse. If they are, they will provide evidence about anything that should be in the dictionary. If they are not, they will miss things.

In the 1990s, the British National Corpus, comprising 100 million words of spoken and written British English, was a model lexicographic corpus. We might then ask: is 100m words sufficient?

A lemmatized frequency list from the BNC shows that the 40,000th most common lemma in it occurs 27 times. It is *prima facie* reasonable to use the top of

the frequency list from the corpus as the headword list for the dictionary. Then, we have 27 examples for "the least frequent headword in the dictionary" so the question becomes, is 27 examples sufficient?

The lemmas with frequency 27 in the BNC include *fieldnote, dyad, connectionist, wannabe, bantamweight, dogfish, unionisation, wallchart, cordite* and *kaftan.* Let us take *dogfish*. Fig 1. shows the 'word sketch' (Kilgarriff et al 2004) for *dogfish* in the BNC. It provides some evidence of the word's behaviour, but most words are there only on the basis of a single co-occurrence with *dogfish* in the corpus.

dogfish *(noun)* **British National Corpus freq = 27** (0.2 per million)

modifier	7	1.0
spotted	1	6.19
Odd	3	4.55
Spur	1	4.41
Lesser	1	4.09
breeding	1	3.75

modifies	2	0.3
Skin	1	1.38

and/or	18	5.5
gurnard	1	10.25
thornback	1	10.19
dogfish	2	10.14
pollock	1	10.09
Scuba	1	9.42
pout	3	9.33
codling	1	7.83
whiting	1	7.48

pp_obj_for-p	3	10.1
Mwnt	1	12.68
mark	1	1.72

pp_obj_with-p	2	9.2
daylight	1	4.73
poor	1	1.17

pp_to-p	1	2.9
lb	1	5.16

pp_obj_of-p	2	1.5
Catch	1	4.26
Nerve	1	3.05

Figure 1. Word Sketch for *dogfish* from the BNC.

If we move to the 1.5 billion-word UKWaC (Baroni *et al.*, 2009) we have 462 occurrences and the word sketch is as shows in Fig. 2.

dogfish *(noun)* **ukWaC freq = 462** (0.3 per million)

object_of	77	1.3
white	2	8.59
dab	6	8.12
pout	3	7.63
catch	8	1.24

subject_of	96	2.8
swim	2	1.46
belong	3	1.32
rest	2	1.14

modifier	202	1.5
lesser-spotted	9	10.22
deep-sea	14	8.53
spiny	7	7.88
huss	2	7.81
wrasse	11	7.74
spotted	13	7.6
whiting	3	6.46
pollack	2	6.35
ling	3	6.34

modifies	88	0.6
coalfish	2	7.61
pollack	3	7.15
conger	3	6.92
wrasse	3	5.97
ling	2	5.9
hopper	5	5.66
eel	3	4.03
skate	2	3.79
cod	2	3.53

and/or	184	3.2
Huss	3	8.48
bullhuss	2	8.46
nursehound	2	8.42
wrasse	16	8.3
gulper	2	8.27
coalfish	4	8.25
smoothhounds	2	8.09
pollock	3	8.05
pollack	5	7.7

Figure 2: Word Sketch for *dogfish* from UKWaC.

This gives a full account of the word, including its varieties (e.g., modifiers) and other members of its semantic class (under *and/or*). We may conclude that, for a 40,000-headword dictionary, a corpus of 2 billion words is substantially better, missing much less, than a corpus of 100 million words.

Where might a corpus of that size, covering a very wide range of text types, be found? The answer is the web. There is now substantial evidence that web corpora, created through the same process of web crawling that the search engines use, offer diverse and very large corpora which compare well with designed collections (Baroni *et al* 2009, Sharoff 2006). Informal and speech-like genres tend to be better represented in web corpora than in many curated corpora, since they contain material from blogs and similar, while curated corpora in the order of a billion words are likely to include high proportions of journalism, the easiest text type to obtain in bulk. While there is no easy answer to the question "what text types, and in what proportions, do we get in a web corpus", Fig. 2 is evidence that they provide good lexicographic resources.

2.1 *Crawling*

The Portuguese corpus was gathered in two parts, the first for European (crawling only in the .pt domain), the second for Brazilian (.br domain). Following Baroni *et al*., we used the Heritrix crawler [http://crawler.archive.org/] and set it up to only download documents of mime type *text/html* and between 5 and 200KB in size. The rationale of mime type restriction is to avoid technical difficulties with converting non-HTML documents to plain text. The size limit weeds out too small documents which typically contain almost no text and very large documents which are very likely to be lists of various sorts. Table 1 summarizes the sizes of the downloaded data as well as the time required for crawling.

	European Portuguese	Brazil Portuguese
HTML data downloaded	1.10 TB	1.37 TB
Unique URLs	31.5 million	39.1 million
Crawling time	8 days (1-8 Mar 2011)	10 days (1-10 Jun 2011)

Table 1: Web crawling stats

2.2 *Junk*

We do not want our Portuguese corpus to contain material that is not Portuguese text. We do not want it to contain navigation bars, banner advertisements, menus, formatting declarations, javascript, html, or material in languages other than Portuguese. It is also important that we represent all texts in a single character encoding (preferably UTF-8) in order prevent incorrect character display.

Detecting original character encoding of each document is our first step, for which we use the chared tool.[1] Once we know what the original encoding is, converting it to UTF-8 is straightforward.

Next, we remove junk (navigation links, advertisements, etc) with jusText.[2] We run it with the inbuilt Portuguese model and with the default settings.

[1] http://code.google.com/p/chared

In order to preserve only texts in Portuguese, we apply the Trigram Python class for language detection using character trigrams.[3] We train a Portuguese language model from a 150,000 word text sample taken from Wikipedia and discard all documents for which the similarity score with the language model is below 0.4. This threshold is based on the results of our previous experiments.

The first manual examination of the corpus data revealed a substantial amount of English text despite the applied language filtering. It turned out that there are numerous documents in the corpus which contain half-Portuguese, half-English paragraphs and score slightly above the language filtering threshold. To fix this problem, we applied further anti-English filtering. We compiled a list of the 500 most frequent words of English and removed from the corpus all paragraphs longer than 50 words where the frequent English words accounted for over 10% of the words.

2.3 *Duplicates*

Duplicates (and, worse still, many-times-replicated material) are bad both because the lexicographer wastes time passing over concordance lines they have already seen, and because they distort and invalidate statistics.

A central question regarding duplication is "at what level"? Do we want to remove all duplicate sentences, or all duplicate documents?

For lexicographic work and other research at the level of lexis and syntax, the sentence is too small a unit, because if we remove all but one copy of a short sentence such as "Yes it is" or "Who's there?" the remaining text will lose coherence and be hard to interpret. The whole document is too large a unit because we do not want to include long sections of text twice over where one appeared in document X, and the other in document Y, and the other parts of document X did not duplicate the other parts of document Y.

The appropriate unit is the paragraph. We identify paragraphs, and then take additional steps to handle short paragraphs (including dialogue turns like "Yes it is"), only removing them if their context is also duplicate material.

A naïve approach to de-duplication results in a process that gets slower per million words, the larger the corpus (since there are more already-seen paragraphs to compare a new paragraph with). Our approach increases linearly with the size of the corpus. We de-duplicate after cleaning, since this reduces the bulk of material to de-duplicate. The de-duplication process was applied separately for the European and Brazilian parts. It took 4 hours and 5 hours respectively on a single Intel Xeon 2.13GHz CPU and removed 75% and 68% of the cleaned material that we had gathered, leaving 804 million tokens of European Portuguese and 3.19 billion of Brazilian.

3. Language technology tools for processing Portuguese

The prospects for getting the computer to help the lexicographer are improved if the text is lemmatized, part-of-speech-tagged and parsed. Then the lexicographer can ask queries about lemmas, word classes, and grammatical relations ("what nouns often occur as objects of this verb?") as well as of word forms and positions ("what words

---

[2] http://code.google.com/p/justext

[3] http://code.activestate.com/recipes/326576 language detection using character trigrams

often come between two and five words after this word?"). We shall be able to provide better reports to the lexicographer.

We investigated past research on the computational processing of Portuguese (e.g., Santos *et al.*, 2008) and established that the leading system was Palavras (Bick 2000). Further investigation revealed that Palavras development has been ongoing for over ten years, and did not reveal any newcomers that looked better. We concluded that it was probably, in 2011, the most accurate software for processing Portuguese. We contacted the author and negotiated a licence.

Parsing tends to be a slow process. One concern of ours was that parsing a 2 billion word corpus would take months or even years.

We parallellised the processing by splitting the corpus into 12 parts and parsing all of them at the same time on a double 12-core AMD Opteron 800 MHz server. We experienced technical problems with the parser and had to re-start several times with software bug fixes and updates obtained from the developers upon our error reports. Despite good technical support, we were unable to parse the whole data set in a single run without the process dying. At last, we split the data into many files of around 10 MB and ran a fresh instance of Palavras for each file. In the final run, with 12 concurrently running instances of the parser, the processing of the whole data set took 15 days.

The parser crashed on most of the input files. Nevertheless, in most cases it managed to process a significant part of the input first. A substantial part of the corpus data was lost during parsing. The final size of the corpus is 773 million tokens for the European part and 1.2 billion tokens for the Brazilian.

4. The Corpus Query System

Preparing the data is just one part of the task: the other is the tool that it is available in.

The publisher was already using the Sketch Engine (Kilgarriff *et al.*, 2004) on other projects, and it is a leading system offering a wide range of benefits, so this was the choice made.

A key report that the Sketch Engine makes available is a word sketch, a one-page summary of a word's grammatical and collocational behavior, as already illustrated in Figures 1 and 2. The raw data for generating word sketches is the set of triples, <grammatical-relation, lemma1, lemma2> in the corpus. It then counts to find the number of examples of each triple, and uses collocation statistics to find the most salient ones.

Generating triples for word sketches has typically been carried out by tagging text for part-of-speech then applying regular expressions over tag sequences. For the current project, we decided to adopt an alternative strategy based on the use of a full parser, which in theory should improve the accuracy with which grammatical relations are detected.

Palavras is a dependency parser. In dependency grammar, the structure of a sentence is identified via a set of labeled dependency links, for each word to its governor. For each word in a sentence, Palavras output provides the lemma, the part-

of-speech tag, the name of the grammatical relation it stands in to its governor, and a pointer to its governor.

Although the dependency relations computed by Palavras are eminently suitable for the generation of word sketches, there are many minor ways in which Palavras output is incompatible with or insufficient for the demands of a practical lexicographic tool. Thus an extensive post-processing phase takes place to adjust Palavras output and enrich it in a variety of ways.

In order to explicitly represent a variety of dependencies, Palavras deconstructs items such as preposition-article contractions and verbs with infixed pronominal objects. For instance, the contraction *dos* ("of the") becomes two separate words (*de os*) with distinct dependencies, while the verb form *levá-lo-á* ("will lead you") becomes two separate words (*levará, o*). It was necessary to reconstruct the surface forms lost by Palavras in order that the lexicographer can extract illustrative examples from the corpus with minimal difficulty.

Palavras also treats a wide variety of multi-word units (eg compound nouns such as *direitos humanos*, as well as many others) as single items in the dependency structure. Untreated, this would have the unfortunate effect of omitting the component words from each other's word sketches. A simple parser was developed to establish the internal dependency structure and headedness of such units, and the result was plugged back into the larger structure with the correct dependencies.

In providing each word with a single governor, Palavras does not explcitly capture relations of importance for complete word sketches. For instance, in the phrase *é viável sua aplicação* ("its application is viable"), a subject relation is established between *aplicação* and *ser*. Post-processing adds in the controlled subject relation between *aplicação* and *viável*, information which may be important in the sketch for these two lemmas. In general, a noun phrase subject will get a subject relation to each verb or adjective in an auxiliary sequence (or an object relation if the verb is passive).

Another type of relation that is added is the trinary relation corresponding to a prepositional phrase and its attachment site. Palavras generates binary relations between the preposition and its governor, and between the preposition and its object. Post-processing adds in the composition of these two, so that each full lexical item will appear on the sketch for the other, in a table headed by the preposition.

A similar treatment is followed for coordination, with post-processing establishing an *e|ou* relation between the heads of the two conjuncts, so that once again they appear on each other's sketches.

As well as augmenting the relations correctly computed by Palavras with various others, it is desirable to correct some of the decisions made by the parser. Betraying its lack of statistical processing, Palavras often attaches constituents to remote heads in ways that may be linguistically possible but are much less likely than the more proximate attachments. For instance, in the phrase *dedicam-se aos temas contemporâneos* ("is dedicated to contemporary themes"), Palavras's choice of *dedicar* as the governor of *contemporâneo* is jettisoned in favour of the much more plausible *tema*.

Finally, for the purpose of collecting as much data as possible within sketches, spelling variations are neutralized in the lemma chosen for each word, with modern Brazilian spelling being used as the standard. Masculine and feminine forms of nouns

are also mapped to the same lemma, in accordance with their usual treatment within the same dictionary entry.

5. Finding good examples

A good dictionary provides lots of examples. Since the early days of corpus lexicography, the practice of inventing examples has been discredited (Sinclair 1987; for a recent and detailed case study see Hanks 2009). Examples should be found in a corpus (though they then may need editing, for example to remove irrelevant material or to change unusual vocabulary items for common ones.)

If the lexicographer needs to read through, on average, twenty of thirty corpus lines before they find a sentence that is suitable to use as a dictionary example, and there are many thousands of examples in the dictionary, then the example-finding starts to use up a large share of the lexicographer-hours in the budget. To address the issue, the Sketch Engine provides the GDEX (Good Dictionary Example finder) program, which sorts concordance lines according to the system's idea of what is likely to make a good example (Kilgarriff *et al.*, 2008). It reduces the average number of corpus lines needing checking to around five.

The original GDEX program was prepared for English. Since then, a version has been prepared for Slovene (Kosem, Husak, and McCarthy 2011), and we are planning one for Portuguese.

6. Regional variants

There are two main regional variants of Portuguese: Brazilian and European. We had corresponding subcorpora within the corpus as a whole, and the Sketch Engine provides a 'keywords' function which can list, in order, all words according to how distinctively Brazilian or European they were.

While this provided the data that we wanted for objective and systematic regional labeling in the dictionary, it did not yet provide it at the right point in the lexicographic process. The point at which the lexicographer should decide whether to include a label for a word is when they are preparing the entry, and they do not want to have to consult a list each time. The most useful place to provide the information is in the word sketch, as the lexicographer will be looking at that when preparing an entry. The Sketch Engine offers a mechanism for adding this kind of information into the word sketch by stating 'hypotheses'. The hypotheses in this case will be:

Is the word one of the top *x* % most-Brazilian words?

Is the word one of the top *x* % most-European words?

(*x* will probably be 0.5.) If the answer to either of these questions is 'yes', a flag will be added to the word sketch saying "highly Brazilian" or "highly European".

7. Summary

We have presented our experience in 'setting up for corpus lexicography' for Portuguese, including building a corpus from the web, cleaning it, deduplicating it,

parsing it, loading it into a corpus tool, and preparing word sketches, good examples and regional labels from it.

References

Baroni, M., S. Bernardini, A. Ferraresi and E. Zanchetta. 2009. The WaCky Wide Web: A collection of very large linguistically processed Web-crawled corpora. Journal of Language Resources and Evaluation 43 (3): 209-226.

Bick, E. 2000. The Parsing System Palavras - Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Famework, PhD thesis. Aarhus: Aarhus University Press.

Hanks, P. 2009. Review of Perrault, S.. editor, *Merriam-Webster's Advanced Learner's English Dictionary.* Int Jnl Lexicography 22 (3), 301-314.

Kosem, I., M. Husak and D. McCarthy 2011. GDEX for Slovene. Proc. Elex 2011 Conference, Bled, Slovenia.

Kilgarriff, A., M. Husák, K. McAdam, M. Rundell and P. Rychlý 2008. GDEX: Automatically finding good dictionary examples in a corpus

Kilgarriff, A., P. Rychlý, P. Smrz and D. Tugwell. 2004. The Sketch Engine. EURALEX Proceedings, Lorient, France.

Rundell, M. and A. Kilgarriff 2011. Automating the creation of dictionaries: where will it all end? In '*A Taste for Corpora: In honour of Sylviane Granger'*F. Meunier, S. De Cock, G. Gilquin, M. Paquot (eds). John Benjamins, pp 257-281.

Santos, F., C. Freitas, H. Oliveira, P. Carvalho. 2008. Second HAREM: new challenges and old wisdom. In *Proc. Computational Processing of the Portuguese Language,* (PROPOR 2008), pp 212–215. Springer Verlag.

Sinclair, J., ed. 1987, Looking Up. COBUILD, Collins.

Sharoff, S. 2006. Creating general-purpose corpora using automated search engine queries. In *Wacky! Working Papers on Web as Corpus.* M. Baroni and S. Bernardini, (eds.) 63-98. Bologna: Gedit.