GDEX: Automatically Finding Good Dictionary Examples in a Corpus

Adam Kilgarriff
Lexical Computing Ltd., Lexicography MasterClass Ltd.

Miloš Husák
Lexical Computing Ltd., Masaryk University

Katy McAdam
A&C Black Publishers Ltd.

Michael Rundell
Lexicography Masterclass Ltd

Pavel Rychlý
Masaryk University, Lexical Computing Ltd.,

Users appreciate examples. If a dictionary entry includes contextualized examples of the different senses a word may have, then the user generally gets what they want in a quick and straightforward way. Thus, there are grounds for including lots of examples and contexts. Producing good examples, however, can be labour-intensive, thus, expensive. We automatically found good candidate sentences in a corpus, with which lexicographers could work. The technology used to add examples to an online version of a leading dictionary: we describe and evaluate the project. We consider a range of other ways in which the finding of good examples can bridge the gap between corpuses, dictionaries, and language learning.

1. Introduction

Users appreciate examples. If a dictionary entry includes an example which is a good match for the context in which the user has encountered a word, or for the context in which they want to use it, then the user generally gets what they want quickly and straightforwardly. Thus there is a case for including lots of examples, for lots of different contexts. In paper dictionaries, the opportunities are limited since examples take up space, and space is limited. But in electronic dictionaries, space is not limited, so the question returns: why not provide lots of examples?

The next constraint is cost: if we are going to add tens, or hundreds, of thousands of examples to a dictionary, the cost of each becomes salient. The work reported in this paper addresses this issue. We automate, or semi-automate, the finding of good examples. Thus the cost of providing very large numbers is reduced. We hope, thereby, to support more dictionaries in providing more examples.

The work was initially for a project for adding examples for a large set of collocations to an English learner's dictionary, and this work is described. We then characterise good dictionary examples and describe how we score sentences. We then review how successful the exercise was, discuss how the methods might be improved, and consider some other ways in which methods for finding good examples could be applied in dictionaries and language teaching.

2. A learner's dictionary project

The first use of GDEX was in the preparation of an electronic version of a leading learners' dictionary, the Macmillan English Dictionary (Macmillan 2002, 2007). The second edition had 500 collocation boxes, for headwords selected using a corpus measure of "collocationality" (Kilgarriff 2006). 500 further headwords were selected in the same way. Each box contained an average of eight collocations. So the task of exemplifying every collocation entailed supplying

8,000 new example sentences, one per collocation. The lexicography for the whole dictionary project was corpus-based, and it was a given that the examples should be corpus-based too.

The default way for a lexicographer to find a good corpus example is, firstly, to get a concordance for the term to be exemplified, and then, read through it until they find an example which is suitable, either as it is or (usually) with some editing. GDEX would analyse all the corpus data for a particular node+collocate pair (such as *commit+crime*), and score the sentences according to their suitability. Instead of scanning sometimes hundreds of concordance lines, human editors would be presented with a shortlist of twenty candidates, from which they would select one for the dictionary. We hoped that the automated pre-selection of a shortlist would significantly reduce the editorial labour involved.

The mechanics of the method were this: for each of the 8,000 collocations, GDEX put the twenty highest-scoring examples into a spreadsheet. The lexicographer then worked on the spreadsheet, putting a tick beside the one they considered best, editing it if necessary. The ticked items were later automatically copied across to the appropriate spot in the lexical database.

3. What is a good example?

Examples have been integral to the monolingual learner's dictionary (MLD) since the early efforts of Michael West and A. S. Hornby in the 1930s and '40s. The arrival of corpora in the early 1980s sparked a lively debate on the relative merits of "made up" versus "authentic" examples, and antagonists delighted in quoting the worst instances of each type. Despite this polarization, MLDs have long contained plenty of examples which did an excellent job regardless of their provenance (Laufer 2007). Now that all ELT lexicography uses corpora, a methodology has developed whereby lexicographers study corpus data to identify recurrent patterns of usage, then select sentences reflecting these norms as the starting point for a dictionary example. Even in the very large corpora available today, it is hard to find whole sentences which perfectly meet all the relevant criteria, and some degree of editorial intervention is usually needed, perhaps to delete an irrelevant and distracting clause, or to change a complex name for a simple one.

A good example must be:

- typical, exhibiting frequent and well-dispersed patterns of usage
- informative, helping to elucidate the definition
- intelligible to learners, avoiding gratuitously difficult lexis and structures, puzzling or distracting names, anaphoric references or other deictics which cannot be understood without access to the wider context. We call this its "readability".

For a fuller discussion of what makes a good dictionary example, see Atkins and Rundell (2008).

Our next task was to translate these requirements into practical and measurable features. Sentence length, for example, has a bearing on informativeness and readability: if the sentence is very short, it may be too context-dependent (or just have too little content) to be informative; but if it is very long, there is more work for the reader to do to read and understand it, and it is more likely to be structurally complex. In the first version of the program we used the following features:

- Sentence length: a sentence between 10 and 25 words long was preferred, with longer and shorter ones penalized.
- Word frequencies: a sentence was penalized for each word that was not amongst the commonest 17,000 words in the language, with a further penalty applied for rare words.
- Sentences containing pronouns and anaphors like *this that it* or *one* often fail to present a self-contained piece of language which makes sense without further context, so sentences containing these words were penalized.

- Sentences where the target collocation is in the main clause were preferred (using heuristics to guess where the main clause begins and ends, as we do not yet use a parser).
- Whole sentence—identified as beginning with a capital letter and ending with a full step, exclamation mark, or question mark, were preferred.
- Sentences with "third collocates", that is, words that occurred with high salience in sentences containing the node and primary collocate, were preferred.
- We note that good examples often first introduce a context, and then contain the collocation which, to speak figuratively, fits into the space that the context has created for it: this is helpful as a user who is unsure of the meaning of the collocation will be able to make inferences about what it must be from the context in which it appears. In sentences having this structure, the collocation is likely to be towards the end of the sentence. Sentences with the target collocation towards the end were given credit.

Once the features have been identified, the question arises: how should they be weighted? Which features are most important, and by how much? With this in mind, we asked two students to select good examples for 1,000 collocations. We then used those "known good" examples to set the weights, by automatically finding the combination of weights that would give the "known good" examples the highest average rank. The first two features, sentence length and word frequencies, were given greatest weight.

4. Was it successful?

The goal of the exercise was to add large numbers of high-quality examples efficiently. The exercise has now been completed, and the Project Manager is in no doubt that it was more efficient with GDEX than it would have been without. Naturally, in an innovative exercise such as this, there were unanticipated difficulties. These relate to the underlying corpus data rather than the algorithm.

The work was started using the British National Corpus (BNC, *http://natcorp.ox.ac.uk*) as the corpus. This was promptly rejected by the lexicographers, simply because it was too old. The BNC, with most texts from the 1980s, discusses Thatcher's Britain. This is not what was wanted for a 21st century dictionary.

A switch was made to UKWaC, a 2 billion word corpus of material gathered from the web (in 2006—see Ferraresi et al. 2008). This was better, and recency was no longer an issue, but "web noise" was. Despite the preference of GDEX for simple, short, grammatically straightforward sentences, the twenty examples for a collocation were still occasionally dominated by lists and other web junk. Occasionally lexicographers did not find any suitable examples in the shortlist and had to go to the full concordance to find one, or make one up.

In sum: yes it worked, but we have an agenda for making it work better.

5. Related work

Sinclair et al. (1998) presents a program, called TYPICAL, which aims to sort concordances according to how typical each line is for the word under scrutiny or "headword". It works by looking at each word in a span to right and left of the headword, determining its relative frequency in the vicinity of the headword as against its frequency in the corpus at large, and summing the scores for each word to give a score for the concordance. The program tends to find, first, a batch of concordances for one collocate, then, a batch for another, and so on. The task of automating the procedure of distinguishing the batches is left for further work. This research is in a similar spirit to ours, but we think it is more constructive to first, find collocations, and then, find good examples for each collocation, as each task can then be separately fine-tuned, and the user can first see the collocations and then select examples where they want them. Also, the work looks at a span, whereas we think it more useful to look at sentences; our work benefits from shallow parsing, which theirs does not; and ours considers readability as well as typicality.

There is a large body of work on readability (DuBay 2004). Historically, the goals have been to identify texts suitable for native speakers at different educational levels, for purposes including teaching people how to read, assessing levels of literacy, and presenting information (for example, in medical forms, or in instructions for how to use machinery). The Flesch Reading Ease measure has been in use for sixty years (Flesch 1948): it computes a readability score from a combination of the average sentence length of a text, and the average number of syllables per word. A range of other measures use similar, easily-computed statistics, sometimes alongside wordlists of easy words. The work has largely proceeded in the USA, with outputs of the tests being mapped to US school grade reading levels.

Recently, language technologists have developed more sophisticated measures which take a corpus of texts where the reading level is known as an input, and use language modelling techniques to develop classifiers which assign grade levels to new texts. Collins-Thompson and Callan (2004) use a corpus of 550 documents found on the web where the author had assigned a grade level. They develop a unigram model for classifying new documents. Their goal is to help web users by filtering web pages according to readability. They claim that traditional measures are ill-suited to the task because they make the assumption that texts comprise well-defined sentences, and also because web texts are often too short, at less than 100 words, for the traditional measures to work well.

Schwarm and Ostendorf (2005) have the goal of finding suitable English reading materials for "Limited English Proficient" students in the American public school system. For their training corpus they use 2,400 articles from *The Weekly Reader*, an educational newspaper with versions targeted at different grade levels, and additional material in both "full" and "abridged-for-children" versions from CNN and Encyclopaedia Britannica. They parse the training corpora to give features including average number of noun phrases and verb phrases per sentence. They also build n-gram language models. They use a range of other features including average sentence length and Flesch score, and combine all features using a Support Vector Machine. Their results show their SVM performing substantially better than traditional measures.

These pieces of work both suggest ways for our research to proceed, although the methods are designed to work on texts, not sentences, and in both cases the training corpus is gathered opportunbisitcally, leaving it unclear hwo well the results will generalise—the authors discuss this issue and conduct experiments to address it, but the question is not fully resolved. The use of a parser by Schwarm and Ostendorf is intriguing, but they do not provide any analysis of whether the statistics from the parse correlated usefully with the grade level.

Kotani et al (2008) build on work that establishes the correlation between reading time and difficulty, and take the task to be one of predicting the reading time that a learner of English will take to read a sentence, given both the sentence and the learner's level of English. The textual features that they use in their model are lexical, syntactic, and discourse. Lexical features include average word length and word difficulty scores based on the "Word level checker" (Someya 2000), which assigns difficulty scores based largely on frequency. Syntactic features involve first parsing the sentence and then computing the width of the parse tree, its height, and the number of branching nodes (the team's earlier research had found a strong correlation between number of branching nodes and reading times). Sentence length was also used. The discourse feature was the number of pronouns, as resolving anaphor is taken to be a challenging task for learners. They used texts from a TOEIC preparation textbook, and obtained reading times by getting students, of a range of levels, to read them. Various language models, using different subsets of features, were learned using multiple regression analysis. The resulting model was able to predict reading times for a particular student and a particular sentence with some accuracy, and would in principle support teachers in identifying particular problems that students had, where there was a large discrepancy between predicted time and actual time.

Both lexical and syntactic factors made a substantial contribution to the accuracy of the model: amongst the syntactic factors, it is not clear whether the parse-tree features improve accuracy beyond the improvement provided for by sentence length.

6. Future work: additional GDEX features

There are several additional features which we are currently implementing:

1. As noted, some lists and "junk" sentences are being selected. This needs exploring and addressing. Since the original work, heuristics which penalise sentences with more than two or three capital letters, and more than two or three punctuation marks and other non-alphanumeric characters, have been implemented.
2. Parsing: it seems a reasonable hypothesis that sentences that are grammatically difficult for learners will also be difficult for automatic parsers. If this is true, then we can apply an automatic parser, and use the automatic parser's performance on a sentence as a feature. At the simplest, we may say that a sentence that the parser cannot parse is a bad example. Amongst sentences that the parser does parse, we may use more sophisticated measures. For example, if the collocation is in a highly embedded clause, the sentence can be penalised. While two of the three papers described in the previous section have used parsers, it is not clear from the reports whether the parser-derived features helped with the task, so the value of parsing for assessing readability remains an interesting and open question.
3. "Language models" are widely used in computational linguistics for assigning probabilities to sequences of words, or sentences, to judge, for example, what the words were by a speech-recognition system, or what is likely to be a fluent translation by a machine translation system. We, like Collins-Thompson and Callan and Schwarm and Ostendorf, can use a language model to assess which of the candidate sentences have a high probability of occurring in English. Provided the language model is chosen and built with care, these will be typical uses of the collocation. An approach to language modelling based on triples of content words, in specified grammatical relations, has been implemented, and other methods will also be explored shortly.

We are in addition preparing a "GDEX customisation interface" which will let dictionary publishers make versions of the system that suit their dictionaries, for example to constrain sentences to contain 75% words drawn from the defining vocabulary of the dictionary in question.

7. GDEX as a bridge between dictionary, learner, and corpus

For over twenty years now, there has been a community of researchers aiming to bring corpora into English Language Teaching: pioneers included Tim Johns and Chris Tribble, and since 1994 there has been the TALC (Teaching and Language Corpora) conference series. Corpora have had a substantial indirect impact on teaching, via dictionaries and teaching materials, but the goal of bringing "corpora into the classroom", so students interact with corpora and find out about the language themselves, has not taken off in the way that advocates predicted. The bald fact is that reading concordances is too tough for most learners. Reading concordances is an advanced linguistic skill. It is hard for a number of reasons:

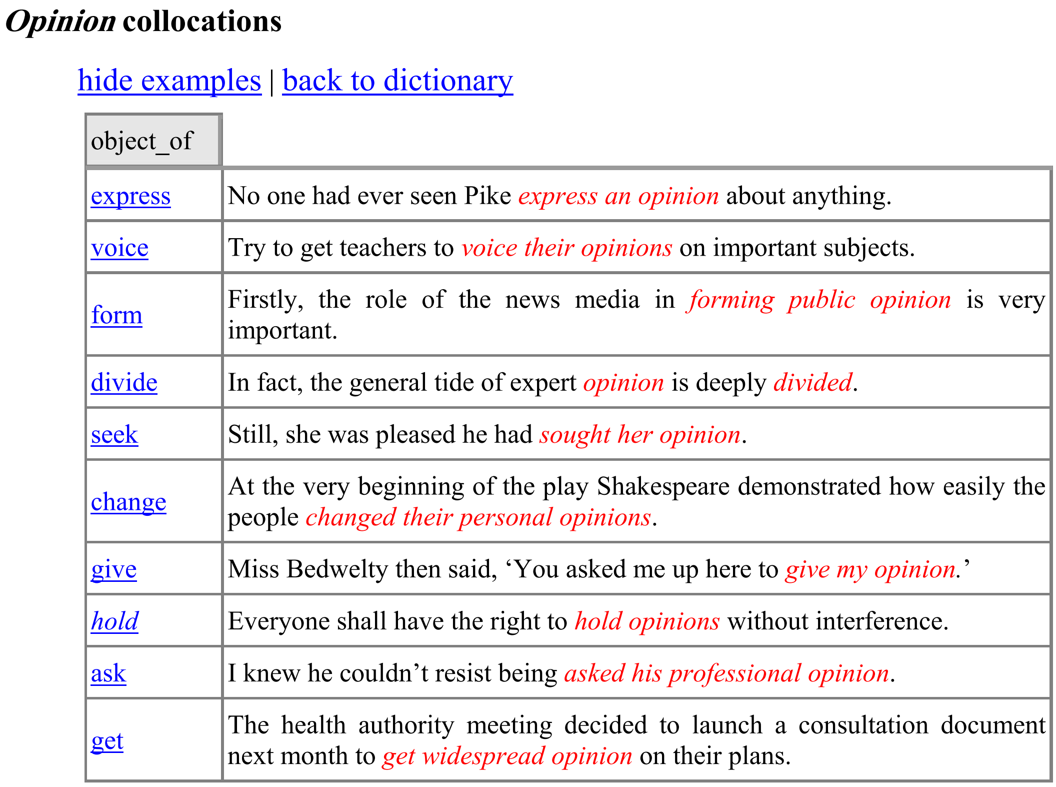
- There is no context available to support interpretation, and no continuity between one line and the next.
- Fragments may not be grammatical and may have unknown vocabulary in them.
- Some fragments will simply be junk and should be discarded, while others are non-standard uses of the keyword which should not be used as models.
- The point of reading concordances—to pick up the common patterns that a word occurs in—is itself an abstract and high-level task.

Advanced and highly motivated learners may be able to benefit from direct interaction with concordances, but, for most learners, concordances are just too tough.

Above we have described how GDEX was used to select candidate sentences, for further selection and editing by a lexicographer, for adding to an electronic version of a learner's dictionary. This can be seen as a halfway house between confronting learners directly with concordances, and using them indirectly for dictionary-making.

There are other models for how GDEX might provide a stepping stone between direct and indirect use of corpora in language teaching. Five approaches currently being explored are:

1. Direct use in the dictionary: GDEX allows us to prepare a file as illustrated in Fig. 1 fully automatically. The collocations are found by the Sketch Engine (Kilgarriff et al. 2004) and each collocation is illustrated by the best example according to GDEX. Electronic dictionary users who have studied the standard dictionary entry, but then want more collocations and examples than it offers, can click on a link in the dictionary entry to see this extended, automatically-generated entry.



***Opinion* collocations**

hide examples | back to dictionary

object_of	
express	No one had ever seen Pike *express an opinion* about anything.
voice	Try to get teachers to *voice their opinions* on important subjects.
form	Firstly, the role of the news media in *forming public opinion* is very important.
divide	In fact, the general tide of expert *opinion* is deeply *divided*.
seek	Still, she was pleased he had *sought her opinion*.
change	At the very beginning of the play Shakespeare demonstrated how easily the people *changed their personal opinions*.
give	Miss Bedwelty then said, 'You asked me up here to *give my opinion*.'
hold	Everyone shall have the right to *hold opinions* without interference.
ask	I knew he couldn't resist being *asked his professional opinion*.
get	The health authority meeting decided to launch a consultation document next month to *get widespread opinion* on their plans.

Fig 1. Corpus derived "more info" for English *opinion* (truncated)

2. Sorted concordances: For the user who does wish to study the concordance, we can help by sorting the concordances "best first". The lines they see first are the ones with highest GDEX scores. This has been implemented. It has the additional advantage, for any corpus user, that non-grammatical and "junk" corpus examples are tucked away towards the end of the concordances so are not shown to the lexicographer unless they scroll through hundreds of examples.

3. Corpus development: in any corpus use, a key question is always "which corpus". Increasingly, the obvious place to go for a corpus is the web (see e.g. Baroni and Bernardini 2004, Kilgarriff and Grefenstette 2003). This raises a host of questions about which web pages should be included in the corpus. One possibility is this: apply a readability measure such as the one embedded in GDEX to all sentences in all candidate web pages, and then, only include a page in the corpus, if most of the sentences have a high-enough GDEX score. In this way a corpus could be generated which was, by design, a corpus that was good for language learning.

4. Corpus annotation: A variant of the previous point is to use a readability score to annotate each document in a corpus with its readability. Then, corpus searches could be

constrained to a subcorpus of a specific readability level. While the US-dominated readability literature usually makes reference to the US school grade levels, in a European and international context the relevant levels are those defined in the Common European Framework (CEF 2001).

5. Automatic collocations dictionary: a fully automatic dictionary, with entries such as the one for *opinion* above, has now been created and can be presented to language learners as a free-standing resource. Although it lacks any analysis of meaning—so, for example, collocations for the bird *crane* will be mixed in with collocations for the machine *crane*—it has the merits of very wide coverage and very large numbers of collocations and examples. A first prototype of this service is available at *http://forbetterenglish.com*

8. Summary

Dictionary users like lots of examples and, if the product is electronic, there is no space constraint blocking publishers from providing them. The constraint then becomes: how expensive are they to prepare? We developed a method for automatically judging which sentences were good candidate dictionary examples. Working together with a team of lexicographers, we applied the method on a large scale to provide 8,000 additional example sentences for collocations for an electronic version of the Macmillan English Dictionary. The method greatly speeded up the process.

Looking ahead, we see various ways in which we could improve the algorithm for judging goodness, and also various models for how the technology could change how language learners interact with corpora. If we can automatically arrange for learners only to be shown corpus sentences that they can easily read and understand, then the greatest barrier to direct learner interaction with the corpus is removed.

References

Atkins, S.; Rundell, M. (2008). *Oxford Guide to Practical Lexicography*. Oxford: Oxford University Press.

Baroni, M.; Bernardini, S. (2004). BootCaT: Bootstrapping Corpora and Terms from the Web. Proceedings, Language Resources and Evaluation Conference.

CEF (2001). Common European Framework of Reference for Languages [on-line]. URL: *http://www.coe.int/t/dg4/linguistic/CADRE_EN.asp*

Collins-Thompson, K.; Callan, J. (2004). *A language-modelling approach to predicting reading difficulty*. Proc NAACL-HLT. Boston.

DuBay, W. (2004). *The principles of readability.* Costa Mesa: Impact Information.

Ferraresi, A. et al. (2008-9). "Introducing and evaluating ukWaC, a very large web-derived corpus of English". In *Proceedings, 4th WAC workshop, LREC, Marrakech, Morocco*.

Flesch, R. (1948). "A new readability yardstick." *Journal of Applied Psychology* 32. 221-233.

Kilgarriff, A. (2006). "Collocationality (and how to measure it)". In *Proc EURALEX, Torino, Italy.*

Kilgarriff, A.; Grefenstette, G. (2003). Introduction to a Special Issue on Web as Corpus. *Computational Linguistics* 29 (3).

Kilgarriff, A. et al. (2004). "The Sketch Engine". In *Proc. EURALEX, Lorient, France*.

Kotani, K. et al. (2008). "EFL Learner Reading Time Model for Evaluating Reading Proficiency". In *Proc CICLING, Haifa, Israel*.

Laufer, B. (1992, 2007). "Corpus-based versus lexicographer examples in comprehension and production of new words". In *Proc EURALEX 1992, Helsinki*. Reprinted in Fontenelle (ed.) 2007. *Practical Lexicography: A Reader.* Oxford University Press.

Macmillan English Dictionary for Advanced Learners. 1st and 2nd ed.. London: Macmillan, 2002, 2007.

Schwarm, S.; Ostendorf, M. (2005). "Reading Level Assessment using Support Vector Machines and statistical language models". In *Proc ACL, Ann Arbor, USA*.

Sinclair, J. et al. (1998). Language independent statistical software for corpus exploration. *Computers and the Humanities* 31. 229-255.

Someya, Y. (2000). *Word level checker: Vocabulary profiling program by AWK*. [on-line] URL: *http://www.cl.aoyama.ac.jp/english/newSite/wlc/index.html* [Access date: 28 March 2008].