A framework for real-time dictionary updating

Cédrick Fairon[1], Sébastien Paumier[1,2]

[1] Centre de traitement automatique du langage, UCLouvain, Place Blaise Pascal 1, B-1348 Louvain-la-Neuve
[2] Institut Gaspard Monge, Université de Marne-la-Vallée, 5bd Descartes, F-77454 Champs-sur-Marne
cedrick.fairon@uclouvain.be, paumier@univ-mlv.fr

Abstract

We present a framework that combines a web-based text acquisition tool, a term extractor and a two-level workflow management system tailored for facilitating dictionary updates. Our aim is to show that, thanks to such a methodology, it is possible to monitor data sources and rapidly review and code new dictionary entries. Once approved, these new entries can feed in real-time client dictionary-based applications that need to be continuously kept up to date.

1. Introduction

Automatic lexical acquisition from corpora has been used for several years as a method for building and/or updating electronic dictionaries (Atkins, 1992; Boguraev, 1996; Fairon, 2002; Evert, 2004; etc.). A common limitation of this approach is that once terms have been extracted from a corpus, the corpus itself becomes valueless, and new corpora must be found. This issue is not unique to lexical acquisition activities, but is common to most corpus-based studies in Natural Language Processing and in applied linguistics. It has lead in the last few years to an increasing demand for corpus development and many researchers have investigated how to automate text collection for language studies. The Web is of course seen as a very promising source (it is freely accessible, it is very large and contains all text types) even if it poses many challenges (Kilgarriff and Grefenstette, 2003).

In this paper, we will discuss the integration of a lexical extractor into a framework that combines a corpus acquisition tool (able to automatically gather new textual data from web sources) with various tools enabling the manual review process of term candidates to be quicker and easier. The focus of our discussion will not be on the lexical extractor (in fact any extractor could be plugged into the system) but rather on the workflow itself and its several steps: acquisition, extraction, review, coding.

2. Overview of the system

2.1. Text acquisition

Two different text acquisition tools provide the extractor with a continuous flow of data taken from online text sources. The first one is GlossaNet[1] (Fairon, 1998), a system that downloads newspapers on a daily basis and turns them into ready-to-use corpora. As these corpora change over time we refer to them as "dynamic corpora"[2]. GlossaNet retrieves the texts on the Web using a crawler that is bound to a predefined set of web domains. The second tool is Corporator (Fairon, 2006), an innovative program that gather texts by downloading articles referenced in RSS news feeds[3]. The main advantage of this technique over the first one is that RSS feeds give access to pre-classified documents, so that it is easy to build specialized corpora (by theme, genre, level of language, etc.). GlossaNet and Corporator have in common to be bound to predefined lists of sources (this particularity distinguishes these systems from the more popular "wide crawling" approach[4]). This hand selection of Web domains may look like a limitation, but as far as dictionary updating is concerned, it is a great asset. In fact, we can select "trusted" sources releasing text of constant quality. Newspaper Web sites are interesting for several reasons:

- they publish texts that have been reviewed through a traditional editorial process ensuring a certain level of language quality;
- they are a great source of new terms, neologisms, names, etc.
- as they are available all around the world, it is possible to monitor how new terms or expressions are spreading geographically (for an illustrative study, see Fairon & Singler, 2006a)

Freshly acquired data are then passed through the extractor.

2.2. Lexical extraction

The lexical extractor was developed using the programs and resources of Unitex[5] (Paumier, 2003). It is designed for identifying simple and compound words

[1] There is a free online interface that enables users to query GlossaNet corpora and build concordances: http://glossa.fltr.ucl.ac.be.

[2] This approach has some similarities with the concept of "monitor corpus" proposed by A. Renouf (1992) in the AVIATOR project which aimed at monitoring language changes over time.

[3] Really Simple Syndication (RSS) is a XML format used for easing data interchange between Web sites. It is very popular on newspapers web sites where it is used for publicizing news article by themes of other classifications. For more information about RSS, see Fairon 2006 or read the New York Times site: http://www.nytimes.com/services/xml/rss/index.html.

[4] See for instance the WaCky Project: http://wacky.sslmit.unibo.it

[5] Unitex is an Open Source linguistic development platform which is based on the DELA resources (a group of large coverage electronic dictionaries first developed by the LADL and its partners under the direction of Maurice Gross and now being maintained at the University of Marne-la-Vallée in collaboration with various European laboratories. See Courtois, 1990 and http://www-igm.univ-mlv.fr/~unitex/)

matching given morphological rules and syntactic patterns. Although it is an important part of the system, this program will not be the centre point of our discussion as we will focus on the general architecture and on the review process (we do emphasise that any other term extraction software could be used in place of this program).

Extracted candidate terms are stored in a database together with the context in which they occurred (under the form of KWIC concordances) and some metainformation (date, name and location of the source in which it was found). The absolute frequency of each term is also recorded and of course updated every time new occurrences of the term are found. As new corpora are automatically fed to the system every day, the counting continues, until a decision is made to accept or reject the candidate. Of course, only candidates that occur above a given threshold will be presented to the reviewer.

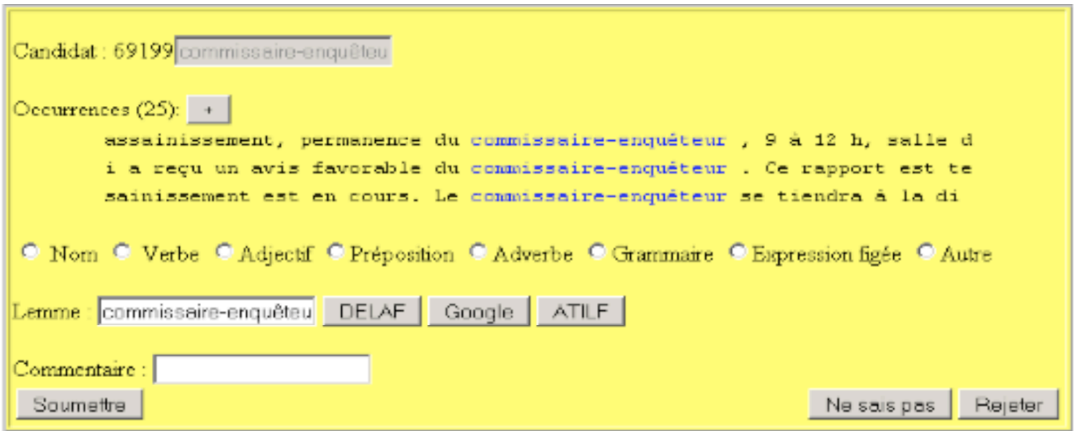

Figure 1: First level review

2.3. Still counting...

In a basic approach to term extraction, the program analyses a text and gives scores to extracted candidates. Lowest scored terms are then simply ignored[6]. In our framework, we keep each candidate as long as it has not reached the minimum threshold that makes it reviewable. As a consequence:

-it is possible for an infrequent word to reach the minimum score even after a certain period of time;

-the system will not ignore a term that is a hapax legomenon in each source but appears in several sources.

Another advantage of the storage/threshold combination is that we can sort the review list by scores to point out words that suddenly appear in news. For instance, if a new molecule against bird flu is found, all the newspapers will mention it, and the name of this molecule will instantly reach the top of the review list, which may be very important in a real-time perspective, as suggested in our title.

However, this system will also select words that have a short lifetime for fashion reasons and users may not want to accumulate such deprecated words in their dictionaries (for example, in a speech processing system or in a spellchecker, it is important to keep the size of the lexicon under control because if it grows too much higher noise and lower performance may result). In order to deal with this problem we could monitor word frequencies over time so that we can detect when a word falls into disuse: when a term that occurs with a high frequency is selected, we check periodically if it continues to appear in newspapers. If a word that was very frequent at some point is not used any more (or only occasionally) over a given period of time, we can decide that this word is obsolete and remove it from dictionaries. On the other hand, if a word still occurs several months after it first appeared, then we could consider it as a new permanent word and stop checking its relevancy. The length of the reference period (weeks, months, years) depends on the application that uses the dictionary.

3. Review process

There are two steps in the review process:

1) extracted terms are manually sorted to remove bad candidates;

2) a linguist verifies each lemma and generates its inflected forms.

3.1. Review of term candidates

In the first step (Figure 1), candidates are presented to the reviewer in a Web-based interface with a small set of concordances (several reviewers can work at the same time, the system keeps track of their respective actions and makes sure that the same term candidates are not assigned to several reviewers). If the word is to be validated, the reviewer selects a grammatical category, provides the correct lemma and then clicks on the submit button ("soumettre" in Figure 1).

A small set of concordances is not always sufficient to decide if a candidate is acceptable or not. This is why the

[6] In more advanced approaches, low score terms can be postprocessed. For instance, if the analysis of an infrequent compound word shows that the nominal head of the compound is the same than the nominal head of another compound of the text which received a good rating, it is probably an interesting candidate. In other words, if *sulfuric acid* occurs 10 times in a text and *lactic acid* only once, the later could received a better rating than the one implied by its frequency because it has the nominal head *acid* which is also present in the more frequent term: *sulfuric acid*.

interface gives access to additional tools that can help in the decision process:

- it is possible to visualise a longer concordance with more examples (see the *plus* sign button on the interface shown at Figure 1);

- the interface also gives direct access to online dictionaries. For French, our interface gives access to two dictionaries, the French DELAF[7] and the online edition of the TLF[8] (*Trésor de la Langue Française*, Atilf). A simple click on one of these two buttons will initiate a dictionary lookup in the corresponding resource.

- another possibility is to run a query in a Web search engine (Google, in our system). The search engine gives a general idea of the candidate's frequency on the web and offers additional examples. For example, Google[9] returns 166,000 occurrences of "commissaire-enquêteur". If the reviewer still cannot decide whether the candidate is appropriate or not, he can simply postpone the decision (button "je ne sais pas"). In this case, the candidate will be stored in a temporary list and will be submitted again later on.

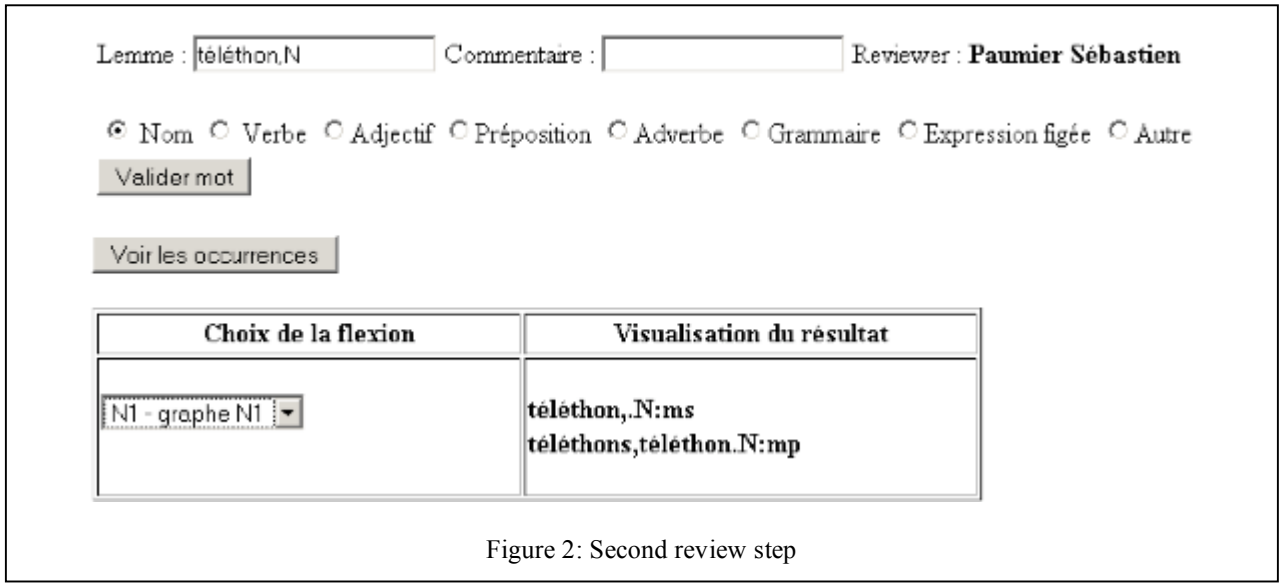

Figure 2: Second review step

If the reviewer rejects a candidate, it is also recorded and added in an anti-dictionary (a stop list) so that the system will not select it for review in the future[10].

3.2. Inflexion & coding

In the second step, a linguist checks if the candidates approved in step 1 are relevant dictionary entries, and then generates the corresponding inflected forms. A morphological tool[11] is integrated in the interface: the reviewer just has to select in a combo box the correct inflection class for a given lemma, and inflected forms are instantly generated (so that the linguist can directly see and check the output of the morphological processor). Figure 2 shows that the inflexion class N1[12] was selected for the French word "téléthon"[13] and that it led to the creation of two dictionary entries, one masculine singular (N:ms) and one masculine plural (N:mp).

When the second level reviewer clicks on the validation button, these data are saved in a database, ready to be exported to any dictionary-based application (for example, an intelligent indexation system, a speech processing system, a spellchecker, etc.).

3.3. Why a two-step procedure?

The two steps are separated for efficiency reasons: in fact, we can argue that the first level does not require the same computational and linguistic expertise as does the second. Moreover a two-step procedure can involve different people and allows a better quality control, which is essential in a real-time application (produced lexical data may be used immediately after their validation by a linguist). This quality control is best ensured if fewer people are involved in the second step than in the first one. It is indeed in step two that one can work on data uniformisation and on the coherence of the data. We consider that this task is best handled if the people working on it are not the ones who have selected the words. Of course, the second-level reviewers must provide feedback to level-one reviewers for any selection/coding problem they notice.

[7] Delaf dictionaries exist for many different languages. They are developed in the framework of the RELEX network. See http://ladl.univ-mlv.fr.

[8] http://atilf.atilf.fr/tlf.htm

[9] When passed to the search engine, the query is automatically quoted so that the engine will look for an exact match in case the query contains several words.

[10] In the case of compound candidates, some are discarded because there are free structures *(Sunday morning)* and some others because they are not valid syntactical units but errors from the extractor. If these categories were sorted, one could explore the idea of using this manually-made anti-dictionary to reject wrong analysis in a parser.

[11] We will not describe in detail the morphological generator. For a general presentation, see the Unitex manual (Paumier, 2003).

[12] As explained in Paumier (2003), the N1 category gathers masculine nouns whose plural is formed by appending a "s" to the singular form.

[13] A television program that aims at collecting funds for medical research.

4. Real-time updates

Why are we examining the possibility of real-time updating of dictionaries used by NLP applications? The reason is simple: these applications are sometimes used in a context in which the language changes rapidly and it is therefore necessary to keep the reference resources updated. The most representative example is probably the domain of news and media information: every day, new names and terms appear in the news and it therefore seems necessary to continuously adapt the lexical resources used in this framework to these developments. Interestingly, these informational texts are also published online, so it is possible to monitor them, extract new terms and update the lexical resources that will be used for analysing the very same texts.

Of course, in some situations it can be inappropriate to dynamically update the dictionary of a production application without running regression tests or without verifying that the modification has no unexpected effect on the system efficiency. This is the main risk of "real-time" dictionary updates and it must be evaluated in each particular application context.

5. Adaptation to other languages

This framework has been designed for French and is currently used for extending the lexical coverage of the French DELA dictionary. We are now working in collaboration with international partners to adapt the system to English, Greek and Portuguese (Brazil). The first review interface obviously needs to be adapted for each language. It is not a problem for the "DELAF" dictionary lookup option because DELAF dictionaries exist for many languages. Neither is it a problem for the search engine option as the Web is highly multilingual (the only adaptation consists in binding the search engine to specific domains or in specifying to the engine the language you are interested in[14]). But of course, it is more difficult to find freely accessible online resources comparable to the TLF for some languages.

6. Conclusion

The framework we have presented in this paper integrates different tools for facilitating the time-consuming task of dictionary updates. The framework is based on a text acquisition tool and on a term extraction program tailored to finding new simple words and multi-words in texts. These tools are combined with a Web-based interface that allows several reviewers to collaborate in the process of approving and coding new words to be added to a dictionary. We explain that such a system can be used for providing real-time dictionary updates by monitoring text sources representing the thematic domain covered by the dictionary. We have mentioned as an example the possibility of monitoring online newspapers in order to retrieve new terms and names that appear in the news and to add them in real time to dictionaries. But it could also be used on specialised sources for updating domain-specific dictionaries.

[14] These are two possibilities common to several search engines like Google, Yahoo or Alltheweb.

Acknowledgement

We would like to thank the CENTAL members who have contributed to the development of this system, in particular Marc Durieux and Isabelle Lecroart.

References

Atkins, B. T. S. (1992). Tools for Computer-Aided Corpus Lexicography: the Hector Project. *Acta Linguistica Hungarica*, 41, pp. 5-72.

Boguraev, B., Putejovsky J. (Eds). (1996). Corpus Processing for Lexical Acquisition. Cambridge, MA: MIT Press.

Evert, S. *et al*. (2004). Supporting Corpus-based Dictionary Updating. In *Proceedings of Euralex 2004*.

Fairon, C. (1998). Parsing a web site as a corpus. In C. Fairon (Ed.), *Analyse Lexicale et syntaxique: le système INTEX. Linguisticae Investigationes*, 22(2), Amesterdam/ Philadelphia: John Benjamins, pp. 327-340.

Fairon, C., Courtois, B. (2000). Extension de la couverture lexicale des dictionnaires électroniques du LADL à l'aide de GlossaNet. In *Proceedings of Journées internationales d'analyse statistique des données textuelles (JADT 2000)*, Lausanne.

Fairon, C, J.V. Singler, (2006a). I'm like, 'Hey, it works!': Using GlossaNet to find attestations of the quotative (be) like in English-language newspapers. In A. Renouf and A. Kehoe (Eds), *The Changing Face of Corpus Linguistics. Language and Computers*, 55. Amsterdam/New York, NY, pp. 325-336.

Fairon, C. (2006b). Corporator: A tool for creating RSS-based specialized corpora. In A. Kilgarriff and M. Baroni (Eds), *Proceedings of the Workshop Web as a Corpus*, Trento, Italy.

Kilgarriff, A. and Gregory Grefenstette. (2003). Introduction to the Special Issue on the Web as Corpus. *Computational Linguistics*, 29(3), pp. 333-348.

Paumier, S. (2003). De la reconnaissance de formes linguistiques à l'analyse syntaxique. Thèse de doctorat en informatique. Institut Gaspard Monge, Université de Marne-la-Vallée.

Renouf, A. (1993). A Word in Time: first findings from dynamic corpus investigation. In J. Aarts, P. de Haan, N. Oostdijk (Eds), *English Language Corpora: Design, Analysis and Exploitation*. Amsterdam: Rodopi, pp. 279-288.

Courtois, B. (1990). Un système de dictionnaires électroniques pour les mots simples du français. In B. Courtois, M. Silberztein (Eds), *Dictionnaires électroniques du français. Langue française*, 87, Paris: Larousse, pp. 11-22.