The Crúbadán Project: Corpus building for under-resourced languages

Kevin P. Scannell [1]
Saint Louis University

Abstract

We present an overview of the Crúbadán project, the aim of which is the creation of text corpora for a large number of under-resourced languages by crawling the web.

Keywords : web crawling, corpus, corpora, minority languages, under-resourced languages, spell checking, language recognition.

1. Introduction

1.1. Background and Goals

Only a very small number (perhaps thirty) of the world's 6000+ languages currently enjoy the benefits of modern language technologies such as speech recognition and machine translation. A slightly larger number (less than 100) have managed to assemble the basic resources needed as a foundation for advanced end-user technologies: monolingual and bilingual corpora, machine-readable dictionaries, thesauri, part-of-speech taggers, morphological analyzers, parsers, etc. (in short, the elements of a so-called Basic Language Resource Kit (BLARK) as in (Krauwer 2003)). The remainder (certainly more than 98% of the world's living languages) lack most, and usually all, of these tools, and we therefore refer to these as *under-resourced languages*.

Since 2000, the author and his collaborators have been engaged in the development of freely-available language kits for a large number of under-resourced languages. The lack of funding opportunities or commercial interest in this work has led to an approach based on certain principles that offer maximal "bang for the buck": monolingual and parallel corpora harvested by web-crawling, language-independent tools when possible, an open-source development model that leverages volunteer labor by language enthusiasts, and unsupervised machine learning algorithms.

The present paper discusses The Crúbadán Project, [1] which implements the first of these principles. The value of the web as a source of linguistic data has been widely recog-

[1] Department of Mathematics and Computer Science, Saint Louis University, Missouri, USA, `scannell@slu.edu`.

[1] See `http://borel.slu.edu/crubadan/`.

nized for nearly a decade (Resnik 1999), (Kilgarriff 2001), (Kilgarriff & Grefenstette 2003), and several authors have addressed the particular importance that “Web as Corpus” (WAC) research holds for under-resourced languages (Ghani *et al.* 2001), (Ghani *et al.* 2005), (de Schryver 2002).

In (Kilgarriff & Grefenstette 2003), it is argued that the notion of a corpus should be “reclaimed” from, among other things, the connotation of representativeness that it has acquired over the years. We believe this is particularly so for the under-resourced languages that form the focus of our work, as the rules of the game are completely different in this case. Indeed, for many of the Crúbadán languages, the few dozen documents retrieved by the crawler may very well represent the totality of all electronic documents in existence, and therefore the notion of assembling a representative corpus (without resorting to old-fashioned methods such as keyboarding or scanning) for such languages is absurd.

In any case, having accepted this inclusive definition of “corpus”, we can claim to have created corpora for more than 400 languages.[2] More significantly, we have used these corpora, in collaboration with native speakers, in the creation of new language technologies for almost 30 languages. Most of the 400+ corpora lack any linguistic annotation for the simple reason that the tools for performing such annotations do not yet exist (see (Rayson *et al.* 2006) and (Baroni & Kilgarriff 2006) for recent work on annotating web corpora for major languages). We have, however, succeeded in bootstrapping part-of-speech taggers for a small number of languages; this is discussed below in §3.1. We should also mention that, in addition to our own work, the Crúbadán data have been provided to many other research groups and individuals working independently on open source projects on behalf of under-resourced languages.

Many of the core ideas in this paper are well-known and we hold no pretensions to originality. In technical terms as well, much of the functionality of our crawler is now implemented and easily available via the open source BootCaT tools (Baroni & Bernardini 2004). Therefore, while we will sketch the basic design of the crawler and offer some implementation details that we hope will be found useful by others working in this area, we skirt over many of the complex issues involved in WAC research (evaluating corpus composition and representativeness, random generation of seed URLs, duplicate stripping, dissemination issues) and recommend (Ciaramita & Baroni 2006), (Evert 2006), (Sharoff 2006b), (Sharoff 2006a) as starting points for readers interested in exploring these issues more deeply. Our focus will instead be on topics particularly pertinent to under-resourced languages, including the sociological aspects of the project which make it somewhat unique in language-processing circles. We expect that community-based approaches like ours will be broadly applicable in trying to break the data bottleneck in NLP applications, especially for minority and under-resourced languages.

[2] See `http://borel.slu.edu/crubadan/stadas.html`.

1.2. Brief History

The roots of the project stretch back to 1999 when the author began creating the first spell checker for the Irish language. The original version of the "crawler" could recursively download complete web sites (or the documents below a specified root directory), convert them to plain text, tokenize, and create a frequency list for use in enhancing the spell checking database.

By 2003, this had evolved into a true web crawler, with a language identification module trained for the six Celtic languages. In the summer of 2004, many new language models were trained (using the techniques discussed below in §2.1.) and a major web-crawl was undertaken that targeted 144 under-resourced languages. At this point the project was dubbed *An Crúbadán*.[3]

In early 2007, in preparation for the present conference, an additional 200 models were trained (bringing the total to 416 languages) and all of the corpora were recrawled. The focus on under-resourced languages means that the amount of data in question is surprisingly small; in this latest crawl we have visited about two million URLs, resulting in the addition of approximately 350 000 documents to the corpora.

2. The Crawler

2.1. Training New Languages

As we discuss in §2.3. below, the default behavior of the crawler is to use simple character trigrams for language recognition. Therefore, training a new language model amounts to nothing more than collecting a sufficient amount of plain text from which reliable trigram statistics can be gathered. The amount of text required varies greatly from language to language, depending primarily on whether or not there are other languages that have similar trigram profiles.

Each language has some additional metadata that must be provided manually: the name of the language in English, the ISO 639-3 code, a flag indicating whether the language is under-resourced, and a list of "polluting" languages (languages one might expect to see frequently in boilerplate text in documents that are otherwise written in the target language; French is a polluter of Lingala, Spanish is a polluter of Basque, etc., and English is set as a polluting language by default). Marking a language as "under-resourced" is mostly impressionistic[4] and is used primarily for reporting purposes on the project web site. After the above-mentioned fields are set, the ISO 639-3 code is used to gather, automatically, additional metadata by screen-scraping the Ethnologue web site[5] (for alternate language names, linguistic classification, countries in which the language is spoken, etc.).

[3] The name means "the crawler" or "the crawling thing" in Irish. The root word is *crúb* ("paw"), which lends the appropriate connotation of unwanted pawing, as in *Ná leag do chrúba orm*, roughly "Get your paws off me".

[4] See (Streiter *et al.* 2007), (Maxwell & Hughes 2006), or (Berment 2004) for discussions of how this notion might be quantified.

[5] See `http://www.ethnologue.com/`.

Many languages have not fully embraced the use of Unicode, and this can be either a minor annoyance (when a standard 8-bit encoding is used but is not indicated correctly in HTML metadata) or a major annoyance (when special 8-bit encodings are used in conjunction with language-specific fonts). In the latter case, we have relied on input from native speakers in order to map the encodings that exist in the wild to standard UTF-8.

Mongolian provides a nice, simple example. Most documents on the web are encoded, at least according to the metadata, as CP-1251. But according to CP-1251, decimal byte values 170, 175, 186, and 191 correspond to Unicode codepoints U+0404, U+0407, U+0454, and U+0457, respectively. In Mongolian documents, however, these bytes are intended to represent codepoints U+04E8, U+04AE, U+04E9, and U+04AF, and the conversion is handled simply by having an appropriate Mongolian font installed when reading CP-1251 documents.

Consider also Polynesian languages such as Hawaiian that have macrons over vowels and the “okina” (glottal stop). Many legacy texts either leave out these special characters entirely, or simply encode them in Latin-1, with no expectation that the macrons will be rendered correctly on the screen. We have seen each of Á, À, Â, Ã, Ä used in this way, as well as a bevy of different characters for the okina. It is interesting to note that several Hawaiian-speaking contacts have suggested that documents not encoded correctly in UTF-8 be left out of the corpus, even though they could easily be converted; the expectation is that these will not be carefully-edited texts, and are more likely to contain misspellings, poor grammar, etc.

Irish offers the best (and an almost absurd) example. In the days before 8-bit email, Irish speakers used to indicate acute accents on vowels with a forward slash following the vowel: “be/al” for “béal”, etc., and this habit persisted well into the 2000’s. As it turns out, certain mailing list archives hosted at `listserv.heanet.ie` form the single largest repository of Irish language text on the web (and therefore, presumably, the largest Irish text collection in history), but these texts are basically invisible to web crawlers and search engines like Google that do not take these conventions into account (Google indexes a word like “be/al” as two words: “be” and “al”).

The vast number of undocumented encoding schemes of this kind illustrates the importance of collaboration with native speakers for a project of this kind. Indeed, we claim that any effort to crawl the web for a large number of languages without attempting to harness the collective knowledge of many language experts, either via direct collaboration or through a large database in the style of XNLRDF (Streiter & Stuflesser 2005), is doomed to failure.

The majority of training texts come from three sites: the Wikipedia,[6] the Jehovah’s

[6] See `http://meta.wikimedia.org/wiki/Complete_list_of_language_Wikipedias_available`, which lists 251 languages as of 22 April 2007. Standard practice on the Wikipedia site is to encode all documents as UTF-8, but this is not always the case, even when the HTML metadata indicates as much, so care is needed when using these texts for training purposes.

Witnesses web site, [7] and the United Nations' Universal Declaration of Human Rights site. [8] The training texts from these three sites were cleaned using *ad hoc* methods suited to these sites. Many other languages were trained using texts provided directly by native-speaking contributors. In cases where an open source spell checking package (hence a word list) was already available, it was possible to generate search engine queries directly (see §2.2. below), and when the spell checker was known to be sufficiently reliable, it could be used directly for language identification purposes, bypassing the trigram approach (and the need for training data) completely.

Next, instead of using these texts directly for the trigram statistics, we perform some additional processing. A word frequency list is generated, and then several filters are applied in an attempt to produce a clean word list. For example, we remove words containing characters not usually appearing in the target language, words with no vowels (when this makes sense), words with the same character appearing three or more times in a row, words with a capital or titlecase character appearing after the first character, words that appear in the word list for a polluting language, and words that contain improbable trigrams (at later stages, after the statistics are available). Also, since it is extremely common in web corpora for diacritics to be omitted, we have found it useful to remove ASCII-only words (like "beal") if a version with diacritics ("béal") appears with higher frequency. Additional language-specific filters can be applied when a native-speaking contact is available, and these can be very powerful – e.g. Hawaiian does not allow two consecutive consonants and Malagasy has similar constraints that allow for very efficient filtering. The end result is a word list we call (imprecisely) the *lexicon*. The trigram statistics used for language recognition are collected from the subcorpus of words appearing in the lexicon.

Three final bits of language metadata are gathered, based on the training texts. First, the trigram vector for the language is compared with every other language in the database, and a list of *nearby languages* is created. Second, one or two "stopwords" are extracted from the frequency list to be used in search engine queries as the crawler runs (as described in §2.2.). When no native-speaking contact is available, this is done automatically by selecting the highest frequency words that do not appear as a high frequency word in another language in the database (in cases where it is difficult to find good stopwords, one can restrict to nearby languages plus the sixty or so that are *not* marked as under-resourced). Third, a list of characters appearing in the texts is created to be used for tokenization purposes. Getting the tokenization correct is very much language-dependent and we often rely on native speaker input for refining this part of the software.

[7] See `http://www.watchtower.org/languages.htm`, with 310 languages as of 22 April 2007. The documents for some languages are given in PDF, presumably when there is a concern that visitors to the site will not have the necessary fonts installed to view UTF-8-encoded HTML. Various Crúbadán contributors have also reported quality issues with the translations, and while these do not seem to be serious enough to affect their usefulness for language recognition, one should be cautious when dealing with languages for which these texts make up the majority of the web presence.

[8] See `http://www.unhchr.ch/udhr/navigate/alpha.htm`, which has 331 languages listed as of 22 April 2007, though, like the Watchtower site, many of these are given as PDF files and cannot easily be converted to plain text. Some of these are now available from `http://udhrinunicode.org/`.

2.2. Basic Design

The crawler focusses on one language at a time. A reasonable alternative would have been to crawl the web very broadly and categorize each downloaded document using the language recognizer, but this is clearly inefficient if one cares primarily about finding texts in languages that do not have a large presence on the web.

Search engine queries are generated by OR'ing together randomly chosen words from the so-called "lexicon" (discussed above), and then AND'ing at least one "stopword". A typical query for Irish might look like this:

```
agus AND sainchomhairle OR ndamhsa OR oirfidigh OR caillteacha OR rancás
```

where "agus" (En. "and") is the stopword. It is the fourth most common word in Irish and so appears in any document of non-trivial size, yet it does not appear commonly in any other language with the exception of Scottish Gaelic.

Using stopwords in this way leads to very high precision in terms of retrieving documents that are actually written in the target language. Extensive tests for Irish confirm this, with queries of the above form returning Irish documents with precision exceeding 98%. Over the long term, the recall is excellent as well, which is not surprising since, given any particular Irish language document you might hope to retrieve, one can easily imagine producing a large number of queries of the above form such that the desired document appears in the top ten results returned by Google. Note that the high precision for Irish is really a measure of the effectiveness of the particular stopword "agus", and for other languages it is sometimes more difficult for find suitable candidates for stopwords. For example, for 121 of the 416 Crúbadán languages (29%), none of the top 10 most frequent words have four or more letters.

The randomly generated queries are passed to the Google API [9] which returns a list of URLs of documents potentially written in the target language. These are downloaded (using the standard Linux tool `wget`) and converted into plain text, encoded as UTF-8. For the conversion to plain text, we have had the most success with the open source programs `vilistextum`, [10] `pdftotext`, [11] and `wvText`. [12] As discussed above, for certain languages the correct conversion to UTF-8 requires some pre- or post-processing.

After this is complete, the language recognizer (§2.3.) is applied to the plain-text candidate document. If it is deemed to have been written in the target language, then it is added to the corpus, and all URLs appearing in the document (either as hypertext links or in running text) are added to the list of "pending" URLs. If it is deemed to have been

[9] Since it appears Google is no longer offering new keys for its search API, finding a reliable alternative has become a more pressing issue for WAC research. We have experimented with other search engines, via the `WWW::Search` Perl modules, with mixed success.

[10] For converting HTML to plain text. It is available from `http://bhaak.dyndns.org/vilistextum/`.

[11] For converting PDF files. This is part of the `xpdf` package; see `http://www.glyphandcog.com/Xpdf.html`. We also use `pstotext` for PostScript files.

[12] For converting Microsoft Word documents. See `http://wvware.sourceforge.net/`; the `wv` library is now integrated into the `abiword` program.

written in a nearby language, the URL is added to a list of seed URLs for that language, to be used later, when the crawler is targeting that nearby language. In all other cases, the document is simply discarded.

For languages flagged as "under-resourced", this process continues until the collection of pending URLs is depleted, at which time the crawl can be terminated, or else a new set of search engine queries can be generated from the new, larger corpus. One important note for under-resourced languages is that true crawling (i.e., following links versus relying only on URLs from search engines) is absolutely essential in order to maximize the size of the corpus. For Irish we have found well over 125 000 documents online, and searching for a random sample of these with Google suggests that only about 90% are indexed by Google.

When the crawl is complete, some housecleaning is performed: duplicate documents are removed from the corpus, a list of "unproductive" top-level domains (many hits but no documents in the target language) is produced, the frequency list is rerun, the filters discussed above are applied to generate a new lexicon, and, finally, the trigram statistics are updated.

2.3. Language Recognition

The software measures the similarity between documents A and B (where one or both of the documents might consist of the existing corpus for a language) using the cosine of the angle between vectors representing the documents in the space of character trigrams, which we denote $c\theta(A,B)$. Just this simple approach is sufficient for distinguishing the vast majority of language pairs in our database; [13] a nice survey of alternate approaches is found in (Hughes *et al.* 2006).

There are some subtle questions regarding language recognition that we will not treat in detail for reasons of space. First is the granularity at which language recognition should be performed. Generally speaking, we work at the document level, but for certain languages of special interest (Irish) we have extracted paragraphs from HTML documents (see also (Zuraw 2006) for interesting remarks, in the context of an under-resourced language, on the value of retaining documents containing even small snippets in the target language). Second, the language recognition threshold is very much language-dependent and requires occasional tuning based on a number of factors. The most important factor, of course, is whether there are languages with very similar trigram profiles in the database. One also has the ability to filter out "low quality" documents by setting the threshold at a high level (say, more than 0.85), but this is counter to our goals when working with under-resourced languages, and we generally set the cutoff to the lowest value possible so that misclassifications are avoided. As an example, for Yoruba, the closest language in the database (Sango) has cosine measure 0.460, so we are able to use a cutoff of 0.50, and this is low enough that a large number of Yoruba documents (e.g. those missing diacritical marks) are found which would not otherwise have been

[13] The complete table of $c\theta(A,B)$ values can be found at `http://borel.slu.edu/crubadan/table.html`.

included in the corpus.

In certain problematic cases, we augment the language recognizer with a naive Bayes classifier that works at the level of words. Examples where additional help has been required are the dialects of Ladin (Badiot, Fascian, Gherdina, and Standard Ladin) and Occitan (Languedocien, Provençal, Gascon, Limousin). In these and similar cases, we had originally trained the crawler to recognize the language in the broad sense ("Ladin", or "Occitan"). Then a list of URLs (on the order of 100-500) of harvested documents was provided to an expert who manually classified them according to dialect, and these were used to bootstrap the Bayesian classifier. Dialects are not the only issue: Cornish, as an example, has at least three competing orthographies and it would be useless for any computational purpose to mix corpora for the three. And of course certain language pairs are as difficult (or more difficult) to distinguish than even some dialects, for example Zulu–Xhosa, Danish–Norwegian Bokmal, and Indonesian–Malay.

3. Applications

3.1. Corpus to Spelling and Grammar Checking

The most satisfying applications of the Crúbadán corpora have been to the most severely under-resourced languages, in particular, those languages lacking even a simple word list.

In §2.1. above, we discussed our algorithm for filtering a raw frequency list in order to generate a (mostly) clean "lexicon" from which our trigram statistics are gathered. To create a *completely* clean (not just statistically clean) word list, we must rely on human editing. Our approach is to first provide the statistically-cleaned lexicon to a native-speaking volunteer – since this list generally contains few errors, the editing goes quite quickly. Then, excerpts from the output of the various filters are examined, and any correctly-spelled words are added to the official cleaned list. New trigram statistics are created based on this editing, and new excerpts are produced for editing. This process continues until the word list is large enough for reasonable spell checking (a recall of 85% of words in typical documents is a reasonable target, but this varies widely according to the morphological complexity of the language).

We have created new open source spell checkers for the following languages using this approach: Azerbaijani, Chichewa, Frisian, Hiligaynon, Kashubian, Kinyarwanda, Kurdish, Malagasy, Manx Gaelic, Mongolian, Scottish Gaelic, Setswana, Tagalog, and Tetum. [14]

Once a clean word list is in place, the next step is to work on morphological analysis, at least to the extent that it is supported by existing open source tools like Hunspell. [15] Creating an "affix file" for Hunspell is quite easy, and while the result is not as powerful as a full transducer, the construction can be done easily by an individual with no linguistic training. The affix file allows simple morphological analysis, and also allows

[14] The truly hard work was done by our collaborators; see the Acknowledgements below.

[15] See `http://hunspell.sourceforge.net/`.

the construction of a (partially) part-of-speech tagged word list. Volunteers can finish tagging the word list manually. Finally, Brill's unsupervised learning algorithm (Brill 1995) can then be used to train a reasonably reliable part-of-speech tagger.

3.2. Lexicography

During 2004 we collected over 100 million words of Welsh, and about half of this text was provided to the University of Wales Welsh Dictionary project.[16] Andrew Hawke emphasized to us at this early stage the value of inclusiveness when corpora are collected for lexicographical purposes (for fear that interesting words might be discarded when boilerplate text or near-duplicates are stripped), and this has guided our actions since.

One benefit of working with under-resourced languages is that they are only rarely the target of "WAC spam" – documents not written by humans who speak the target language but instead generated automatically by a computer one way or another. We encountered a small amount of WAC spam while developing the Welsh corpus (apparently generated by a dim-witted word-for-word machine translation program) and we have seen some in Irish also (an *n*-gram word model). In each case it was a simple matter to write a language-specific filter to detect these, but creating a language-independent filter, or filters for 400+ languages will be a major obstacle.

3.3. Other Applications

We have provided data (sentences, frequency lists, language identification data) to several dozen other projects. These projects involve everything from lexicography, morphology, and diacritic replacement (Wagacha *et al.* 2006) to machine translation, word sense disambiguation, and thesaurus construction. We will continue to share the data with research groups that release their own software under an approved open source license.[17]

Acknowledgements

The Crúbadán project owes its success to the volunteer efforts of more than 75 contributors from over 40 countries; the full list is available from the project web site.[18] We hope that relegating their names to the web site in this way does not appear to diminish our gratitude for their tireless work and enthusiasm. The project was undertaken (and continues) with no external funding, though the author is grateful to his home institution for a well-timed sabbatical during which the majority of the development was done.

References

BARONI M. et BERNARDINI S. (2004), "BootCaT: Bootstrapping Corpora and Terms from the Web", in *Proceedings of LREC 2004,*Lisbon, Portugal.

BARONI M. et KILGARRIFF A. (2006), "Large linguistically-processed Web corpora for multiple languages", in *Proceedings of the 11th EACL Conference,*Trento, Italy.

[16] See `http://www.aber.ac.uk/~gpcwww/gpc_visi.htm`.

[17] See `http://www.opensource.org/licenses` for a complete list.

[18] See `http://borel.slu.edu/crubadan/stadas.html`.

BERMENT V. (2004), *Méthodes pour informatiser des langues et des groups de langues peu dotées*, PhD thesis, Université Joseph Fourier.

BRILL E. (1995), "Unsupervised Learning of Disambiguation Rules for Part of Speech Tagging", in Yarowsky D. & Church K. (Eds), *Proceedings of the Third Workshop on Very Large Corpora*,Cambridge, Massachusetts.

CIARAMITA M. et BARONI M. (2006), "Measuring Web-Corpus Randomness: A Progress Report", in Baroni M. & Bernardini S. (Eds), *WaCky! Working Papers on the Web as Corpus*, GEDIT, Bologna.

DE SCHRYVER G.-M. (2002), "Web for/as Corpus: A Perspective for the African Languages", in *Nordic Journal of African Studies*, n^{o} 2, vol. 11.

EVERT S. (2006), "How Random is a Corpus? The Library Metaphor", in *Zeitschrift für Anglistik und Amerikanistik*, n^{o} 2, vol. 54.

GHANI R., JONES R. et MLADENIĆ D. (2001), "Mining the Web to Create Minority Language Corpora", in *Proceedings of the 10th international conference on Information and knowledge management*,Athens, Georgia : 279–286.

GHANI R., JONES R. et MLADENIĆ D. (2005), "Building Minority Language Corpora by Learning to Generate Web Search Queries", in *Knowledge and Information Systems*, n^{o} 1, vol. 7.

HUGHES B., BALDWIN T., BIRD S., NICHOLSON J. et MACKINLAY A. (2006), "Reconsidering Language Identification for Written Language Resources", in *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC2006)*,Genoa, Italy : 485–488.

KILGARRIFF A. (2001), "Web as corpus", in *Proceedings of the Corpus Linguistics 2001 Conference*, Lancaster University : 342–344.

KILGARRIFF A. et GREFENSTETTE G. (2003), "Introduction to the Special Issue on the Web as Corpus", in *Computational Linguistics*, n^{o} 3, vol. 29.

KRAUWER S. (2003), "The Basic Language Resource Kit (BLARK) as the First Milestone for the Language Resources Roadmap", in *Proceedings of the International Workshop "Speech and Computer", SPECOM 2003*,Moscow, Russia.

MAXWELL M. et HUGHES B. (2006), "Frontiers in Linguistic Annotation for Lower-Density Languages", in *Proceedings of the COLING/ACL2006 workshop "Frontiers in Linguistically Annotated Corpora"*,Sydney : 29–37.

RAYSON P., WALKERDINE J., FLETCHER W. H. et KILGARRIFF A. (2006), "Annotated web as corpus", in Kilgarriff A. & Baroni M. (Eds), *Proceedings of the 2nd International Workshop on Web as Corpus (EACL06)*,Trento, Italy : 27–34.

RESNIK P. (1999), "Mining the web for bilingual text", in *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99)*,College Park, Maryland : 527–534.

SHAROFF S. (2006a), "Creating General-Purpose Corpora Using Automated Search Engine Queries", in Baroni M. & Bernardini S. (Eds), *WaCky! Working Papers on the Web as Corpus*, GEDIT, Bologna.

SHAROFF S. (2006b), "Open-source corpora: Using the net to fish for linguistic data", in *International Journal of Corpus Linguistics*, n^{o} 4, vol. 11.

STREITER O., SCANNELL K. et STUFLESSER M. (2007), *Implementing NLP Projects for Non-Central Languages: Instructions for Funding Bodies, Strategies for Developers*, To appear in *Machine Translation*.

STREITER O. et STUFLESSER M. (2005), "XNLRDF, the Open Source Framework for Multilingual Computing", in Ties I. (Ed), *Proceedings of the conference "Lesser Used Languages and Computer Linguistics"* : European Academy,Bozen-Bolzano, Italy : 189–207.

WAGACHA P. W., DE PAUW G. et GITHINJI P. W. (2006), "A Grapheme-Based Approach for Accent Restoration in Gĩkũyũ", in *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC2006)*,Genoa, Italy : 1937–1940.

ZURAW K. (2006), "Using the Web as a Phonological Corpus: a case study from Tagalog", in Kilgarriff A. & Baroni M. (Eds), *Proceedings of the 2nd International Workshop on Web as Corpus (EACL06)*,Trento, Italy : 59–66.