OPUS - an open source parallel corpus

http://logos.uio.no/opus/

Jörg Tiedemann
Department of Linguistics
Uppsala University
Box 527
SE-751 20 Uppsala, Sweden
joerg@stp.ling.uu.se

Lars Nygaard
Tekstlaboratoriet HF
University of Oslo
Postboks 1102 Blindern
0317 Oslo
lars.nygaard@ilf.uio.no

1 Introduction

Parallel corpora are useful in a wide variety of research areas, particularly in machine translation and lexicography. However, parallel corpora have been few, often unrepresentative, and not generally available. The aim of the OPUS project is to provide a public collection of parallel corpora which can be freely used and distributed. This makes it possible for everyone to run experiments on bitexts and their results can be easily compared.

We base our corpus collection on open source documentation and their translations. Many open source projects include a large amount of textual data and invite people around the world to localize products and their documentation. Similarly to the software itself, the entire documentation is freely available and may be used in any way by anybody.

The idea of using translated text which can be collected from the web is not new (see e.g. [Res98]). In OPUS, we download translated texts from the web, convert and align the entire collection, add linguistic data, and provide the community with a publicly available parallel corpus. OPUS is based on

open source products and will also be delivered as an open source package along with tools for creating and searching parallel corpora.

2 Basic principles

2.1 Text collection

The main goal of OPUS is to collect an extensive number of translated documents in a large variety of languages from sources which are freely available on the web. Currently, we concentrate on a specific domain, namely software documentation of open source projects. However, other kinds of texts will also be included such as translated documents which are placed in the public domain by governments, international organizations and news agencies. Many projects include localization efforts for translating manuals, tutorials and other documentation. Open source projects are often international efforts and many developers and users contribute by adding and correcting information in their native language. Open source projects are well suited for our purposes as they tend to use international standards such as Unicode[1], XML (e.g. DocBook[2]), HTML and XHTML[3]. We tried to locate as many resources as possible which fit into our collection. So far, we searched manually for translation projects on the web. In the future, we plan to use automatic techniques for the location of translated texts which can be downloaded from the web. Each text collection which is added to the corpus is stored with the same file structure as in the original package with all its sub-directories and separate documents. This makes it easy to identify the origin of each text segment and to use parts of varying sizes.

2.2 Corpus encoding

We use very simple principles for the encoding of the documents in our corpus. All textual data are stored in standalone XML documents using Unicode (UTF-8). Generally, we want to maintain the existing markup as much as possible. Therefore, we decided to use stand-alone XML documents without restricting the corpus markup to elements defined in a specific document type definition (DTD). However, additional markup is added according to the corpus encoding standard for XML (XCES)[4]. Hence, corpus encoding

[1] http://unicode.org/
[2] http://www.docbook.org/
[3] http://www.w3.org/MarkUp/
[4] http://www.cs.vassar.edu/XCES/

consists of two tasks: The conversion of the original document format to valid XML (if necessary) and the conversion of the original character encoding format to Unicode UTF-8. For these two tasks we apply standard tools which are freely available such as *recode* (converts between many character encoding standards) [Pin00] and *tidy* (validates and pretty-prints HTML and XML files, converts HTML to XHTML) [Rag03]. Furthermore, we apply the Uplug toolbox which can be used to add basic markup, sentence splitting, tokenisation, and some simple XML processing [Tie02].

2.3 Alignment

Another goal of OPUS is to provide *parallel* corpora which presupposes an alignment of translated documents of some kind. We use a length-based approach for sentence alignment based on the algorithm and software by Gale&Church [GC93]. Links between sentences are stored in external XML files using the XCES format. Automatic sentence alignment is robust and earlier studies have shown a reasonable accuracy of length-based approaches for many language pairs. However, automatic alignment approaches do not produce results with absolute accuracy. A common problem of length-based approaches is the possibility of follow-up errors, which may be caused by, for instance, missing segments in one of the translations. Follow-up errors can be avoided using hard boundaries which are anchor points in the parallel documents. This is one reason why we maintained the document structure from the collection of original documents in our corpus. Each file in each sub-directory is aligned separately with its translation to keep alignment errors at a low level. In other words, the beginning and the end of each file function as hard boundaries for securing a high quality of the sentence alignment.

Another general principle in OPUS is that we align corresponding documents for all possible language combinations. In other words, we align all pairs of translations of a common original with each other. Thus, we obtain a large variety of language pairs in aligned documents. This provides a valuable resource for cross-lingual investigations.

2.4 Linguistic markup

Sentence alignment presupposes sentential markup. A simple sentence splitter from the Uplug toolbox is used to add sentence boundaries to documents in the corpus. Furthermore, documents are tokenized using either language specific tokenizers or the default tokenizer from the Uplug toolbox. As men-

tioned earlier, we keep all pre-existing markup in the documents which helps to improve the quality of the sentence splitter and tokenizers. Markup such as paragraphs, headers, tables and lists give useful cues for the segmentation of texts into sentences, words and other tokens.

The project also aims at adding linguistic information to the corpus. For this purpose, we applied available tools for language specific markup. In the current version of OPUS we used part-of-speech taggers (TnT, TreeTagger, Grok, ChaSen) and a shallow parser (Grok). Grok [Bal01] and ChaSen [MKY+00] are freely available from the web. Grok is an implementation of the OpenNLP interfaces and comes with modules for tagging and chunking English texts. Both modules are trained on the Penn Tree Bank using the Penn tagset [MSM93]. ChaSen is a tokenizer and morphological analyzer for Japanese. It provides several kinds of linguistic information such as readings, parts of speech, and base forms. TnT [Bra00] and the TreeTagger [Sch94] are freely available for research purposes and their usage on OPUS files has been granted by the authors. They come with ready-to-use modules for tagging English and German (TnT & TreeTagger) and for French and Italian (TreeTagger only). Furthermore, the TreeTagger also comes with a lemmatizer for all supported languages. Both taggers can be trained on other material. A module for tagging Swedish with TnT trained on the SUC corpus has been provided by Beáta Megyesi [Meg01]. The Uplug toolbox is used to convert between input and output formats which are used by these tools and, finally, to produce the markup in the corpus format that is used by OPUS.

3 OPUS v0.1

In the current stage, we focus on a specific domain, namely software documentation, which can be found in open source projects. OPUS consists so far of the documentation of the office package OpenOffice.org[5] with its original collection of 2014 files in English and five collections of translated texts (French, Spanish, Swedish, German, and Japanese). The English part comprises about 500,000 words. Not all files have been translated yet. Each translation except the Japanese part contains between 400,000 and 500,000 words. The Japanese text contains about 270,000 lexical units which have been identified by means of the morphological analyzer (ChaSen). The entire corpus includes about 2.6 million words in its current version. Table 1 summarizes some characteristics of the OpenOffice.org corpus.

[5] http://www.openoffice.org/

language	nr files	nr words
English	2,014	478,654
French	1,739	496,777
Spanish	1,738	491,426
Swedish	1,739	403,195
German	2,014	474,436
Japanese	1,739	267,656
	10,983	2,612,144

Table 1: The OpenOffice.org corpus.

4 Availability

The corpus is available from the OPUS home page[6]. There is also a CVS archive[7] which is accessible from this web page. The complete corpus can also be downloaded as a whole in compressed GNU archive files. Furthermore, we converted sentence aligned bitexts for each possible language combination to HTML encoded documents which can be browsed from the OPUS home page. An example of a Spanish-Japanese alignment sample can be found in figure 1 below.

The sentence-aligned OpenOffice.org corpus (except the Japanese part) has also been indexed by the Corpus Work Bench [Chr94]. The index is accessible via web-interfaces[8] and can be searched for multiple languages in parallel[9]. The Corpus Work Bench comes with a powerful query engine, the Corpus Query Processor (CQP) which is used by our interfaces. CQP allows the search for words, patterns and any kind of lexical annotation. OPUS annotations such as part-of-speech tags can be queried in a very efficient way using the powerful CQP query language.

5 Future work

OPUS is meant to be open for extensions and is open to contributions from the community. In particular, we would like to add tools such as tokenizers, taggers and chunkers for the languages which are represented in the corpus. We are currently working on adding further material. The contents of the

[6] http://logos.uio.no/opus/
[7] CVS is a widely used version control system.
[8] http://logos.uio.no/opus/search.html
[9] http://logos.uio.no/opus/oo.html

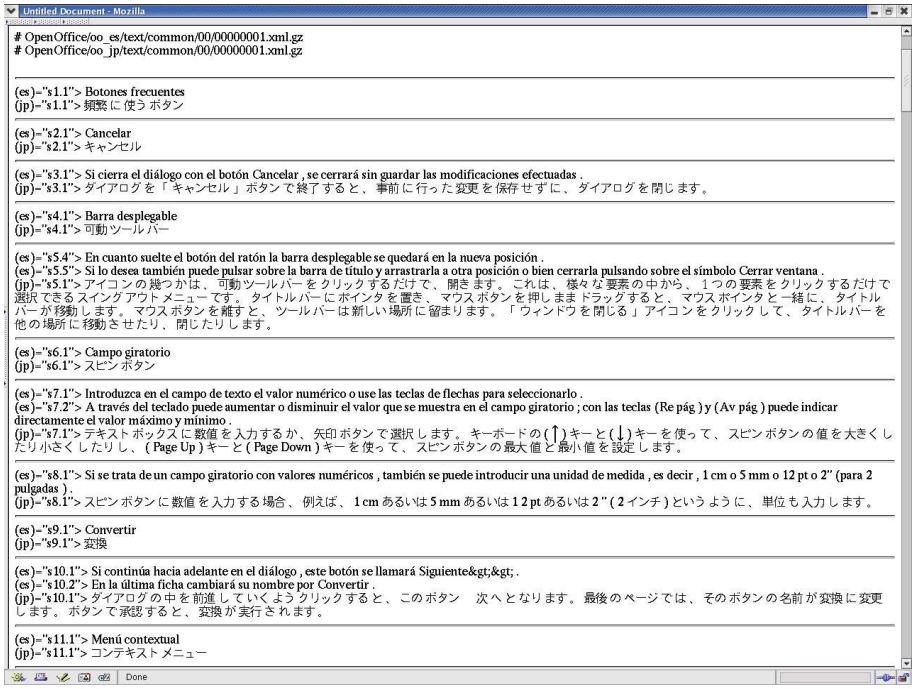

Figure 1: A Spanish-Japanese alignment example.

collection will be updated continuosly, and updates will be announced on the project web page. We also want to add further search facilities to be used for exploring the corpus on-line. Additional markup in the corpus is another goal in the project. For instance, we would like to add part-of-speech information to a larger variety of languages. Furthermore, we would like to process the corpus with word alignment tools in order to extract multi-lingual domain-specific term databases, which can be made available to the community, for instance, for the localization of open-source software. We also work on tools for the automatic retrieval of translation data from the web using web crawlers and other techniques. This supports the main goal of OPUS, i.e. to provide an extensive parallel corpus with a large number of languages included.

References

[Bal01] Jason Baldrigde. Grok - an open source natural language processing library, 2001.

[Bra00] Thorsten Brants. TnT - a statistical part-of-speech tagger. In *Proceedings of the 6th Applied Natural Language Processing Conference ANLP-2000*, pages 224–231, Seattle, WA, 2000.

[Chr94] Oliver Christ. A modular and flexible architecture for an integrated corpus query system. In *Proceedings of the 3rd International Conference on Computational Lexicography (COMPLEX)*, pages 22–32, Budapest, 1994.

[GC93] William A. Gale and Kenneth W. Church. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102, 1993.

[Meg01] Beáta Megyesi. Comparing data-driven learning algorithms for POS tagging of swedish. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 151–158, Carnegie Mellon University, Pittsburgh, PA, USA, June 2001.

[MKY+00] Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano, Hiroshi Matsuda, Kazuma Takaoka, and Masayuki Asahara. Morphological analysis system chasen version 2.2.1 manual. http://chasen.aist-nara.ac.jp/chasen/bib.html.en, December 2000.

[MSM93] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of english: the penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

[Pin00] François Pinard. The recode reference manual. http://www.iro.umontreal.ca/contrib/recode/HTML/recode.html, 2000.

[Rag03] Dave Raggett. Clean up your web pages with html tidy. http://www.w3.org/People/Raggett/tidy/, 2003. http://tidy.sourceforge.net/.

[Res98] Philip Resnik. *Machine Translation and the Information Soup: Third Conference of the Association for Machine Translation in the Americas (AMTA-98), Langhorne, PA*, chapter Parallel Strands: A Preliminary Investigation into Mining the Web for Bilingual Text. Number 1529 in Lecture Notes in Artificial Intelligence. Springer, October 1998.

[Sch94] Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, September 1994. http://www.ims.uni-stuttgart.de/ schmid/.

[Tie02] Jörg Tiedemann. Uplug - a modular corpus tool for parallel corpora. In Lars Borin, editor, *Parallel Corpora, Parallel Worlds*, pages 181–197. Rodopi, Amsterdam, New York, 2002. Proceedings of the Symposium on Parallel Corpora, Department of Linguistics, Uppsala University, Sweden,1999.