N. Klyueva, O. Bojar

UMC 0.1: CZECH-RUSSIAN-ENGLISH MULTILINGUAL CORPUS[1]

1. Introduction

The number of parallel corpora has been growing recently, since they represent a valuable resource of linguistic information. They can serve for various needs of theoretical and computational linguistics as well as for natural language processing applications. Herein, we present the UMC (UFAL Multilingual Corpus) – a multilingual parallel corpus of texts in Czech, Russian and English languages with automatic pairwise sentence alignments.

UMC is closely related to CzEng, a Czech-English corpus released by our department and successfuly employed in a machine translation (MT) competition.[2] The primary aim of UMC is to extend the set of languages covered by our corpora mainly for the purposes of MT. In the present work we will pay special attention to the Czech-Russian pair.

As a starting point, we have chosen only one web source to download our texts and up to now we were able to obtain over 1.7 million words in each of three languages. In future releases, we are planning to include parallel texts from additional sources.

2. Existing parallel corpora for the languages

1 The work on this project was supported by the grants GAAV CR 1ET201120505, FP6-IST-5-034291-STP (EuroMatrix), MSM0021620838, grant of Ministry of Education of the Czech Republic No. ME838 and NSF PIRE grant #0530118.

2 **http://www.statmt.org/wmt08**

One can find several Czech-English, Russian-English corpora, as the respective parallel texts can be easily obtained from the web and other sources. For example:

- **CzEng**[3] – a parallel corpus of Czech-English texts, which contains about 20 million (for Czech) and 23 million (for English) words including punctuation marks.
- **Russian-English corpus**[4] contains 2345 BBC new articles.
- The parallel Russian-English corpus is also currently developed within the project **RNC**[5] - Russian National Corpus

The situation is different for the pair Czech-Russian, since less parallel texts exist in a machine readable form in those languages. Still, we are aware of at least two parallel corpora of belletristic texts, though considerably smaller in size:

- **RPC**[6] (Regensburg Parallel Corpus) has 11 belletristic texts in both Czech and Russian and above all in some other Slavic languages.
- The project **Intercorp**[7] also include belletristic texts in the two languages.

To the best of our knowledge there are no Czech-Russian corpora downloaded from the Internet.

3. Data sources

Collecting parallel texts meets such challenges as copyright, translation quality and representativeness of the language. The problem of copyright is decided by contacting the page editor asking for a licence agreement for educational purposes. It is more complicated with a translation quality, because when downloading

3 **http://ufal.mff.cuni.cz/czeng/**
4 **http://l2r.cs.uiuc.edu/~cogcomp/Data/Temporal/temporal.html**
5 **http://www.ruscorpora.ru/**
6 **http://www.uni-regensburg.de/Fakultaeten/phil_Fak_IV/Slavistik/RPC/index.html**
7 **http://www.korpus.cz/intercorp/**

automatically huge amount of texts, they can not all be checked, so we look only at the extralinguistic factors.

Let us look on the texts in both Czech and Russian, that we can come across in the Internet.

The greatest part of them probably belongs to the tourism industry as many hotels, restaurants, tourist sites are advertising their services both in Czech and Russian. The disadvantages are that the texts are generally short and the translation quality is doubtful.

Technical texts present the second, more reliable and broad group, but their representativeness is low, as they contain lots of technical terminology and general language usage is limited. On the other hand, those type of texts are most suitable as a limiting domain for MT, as the language is strict and metaphorical use of language is rare. In most cases the pivot language is English, and the texts are translations from English to Czech and from English to Russian.

Newspaper articles are written in language rich with metaphors sometimes with tricky constructions, which can be translated differently in different languages. However, the language of news covers the most essential part of standard language use, so we have chosen to use the news articles in the first phase of the experiment.

As it was mentioned, all the texts were downloaded from a single source — The Project Syndicate[8], which contains a huge collection of high-quality news articles and commentaries. We were given the permission to use the texts for research and non-commrcial purposes.

Our next step is to expand the corpus with KDE documentation in corpus OPUS[9] and probably with some other manuals to operating systems and computer applications.

4. Document processing

8 **http://www.project-syndicate.org/**

9 *Teidmann J. and Nygaard L.* The OPUS Corpus – parallel and free. // In Proceedings of the Fourth International Conference on Language and Evaluation (LREC-04). 2004.

Texts were downloaded with the help of tools developed under the project CzEng. The total amount of downloaded documents is 2.186 in each of the three languages.

In the table 1 we summarize the size of the corpus in the terms of words, tokens and sentences.

Table 1. Corpus Statistics

	Czech	**Russian**	**English**
words	1,747,997	1,815,550	1,920,164
tokens	2,022,990	2,152,326	2,255,901
sentences	96,335	101,528	97,250

4.1. Transforming formats

HTML files are transformed into text documents by extracting text paragraphs from pages. The original pages do not include pictures, tables or mathematical formulas, so the process is rather straightforward. Still there is some noise in the data, and we will be improving the cleaning algorithm. Unlike the project CzEng, where the preferation was given to the XML storage format, in UMC we use plain text format so far.

4.2. Tokenization and segmentation

We implemented a simple yet powerful data-driven trainable tokenizer and segmenter, inspired by the tool TextSeg[10]. Our tokenizer proceeds in three separate steps: (1) create "rough" tokens,

[10] *Bojar O., Janíček M, Žabokrtský Z., Češka P., Beňa P*. CzEng 0.7: Parallel Corpus with Community-Supplied Translations. // In Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC- 08). 2008.

mark choice points, (2) decide what happens at the choice points, preserving rough token count, and (3) interpret the decisions, possibly joining some of the rough tokens, never further increase the number of tokens. This separation allows us to keep the steps (1) and (3) very simple, deterministic and based on a very small context of at most two neighbouring tokens while all the decisive power in step (2) is left to a machine learner that is trained on manually annotated data.

In the first step of "rough tokenization" we break text into tokens at every white space, word boundary (between a letter and a non-letter), digits boundary (between a digit and a non-digit) and after every other single symbol. Moreover, some language-dependent tokenization exceptions allow to split words, such as the English "don't" into "do n ' t". We insert special tokens to denote original line breaks (
), paragraph breaks (<P>), as well as positions where there was no space, but we split the token (<D>).

Another special symbol <mayS> is deterministically inserted at all places where a new sentence might start (e.g. after a full stop, question mark or paragraph break).

In cases where tokens originally delimited by some white space may have to be joined (e.g. space-delimited numbers over one thousand), we insert a special token <mayjoin>. These places are again deterministically identified by a set of language-dependent rules.

For example, the text "*Don't send $5 000.00.*" becomes "*Do <D> n <D> ' <D> t send $ <D> 5 <mayjoin> 000 <D> . <mayS> <D> 00 <D> . <mayS>*".

The second step makes all necessary decisions while preserving the number of rough tokens. We apply a machine learner to change <D> or <mayjoin> into <joinD> at places where a token should not have been split, e.g. in "*n't*" or "*5000*" and to change <mayS> into <S> at places where the sentence indeed ends. The machine learner is trained on language-specific manually annotated data demonstrating precisely these decisions. To speed up the manual annotation, the

annotators are provided with a simplified tokenized text (no special tokens) where a space indicates every proposed token boundary and a new line indicates every proposed sentence boundary. All they have to do is to remove spaces where tokens should not have been split and to remove new lines where sentences do not end.

One of the main advantages of this step is that we can easily train our tokenizer to follow rules exhibited in other corpora, we need just the original text version and the intended tokenized version with every sentence on a new line. Only a few of the aforementioned exceptions for joining at whitespaces and splitting within words may have to be added.

We use the MAXENT learner developed by Le Zhang[11] and our features include specific tokens from a window of 14 tokens around each choice point as well as regular expressions describing the tokens (e.g. a number, capitalized word etc.) and features checking for membership in various lists (e.g. common abbreviations or month names to improve sentence-boundary detection).

Given enough training data, our implementation of the tokenizer is able to correctly identify sentence boundaries even in complicated cases such as an abbreviation followed by a full stop, i.e. 'и т.п.'. However, there are cases where the disambiguation requires to consider a broad context, cf. 'г. Москва' (город, city) and 'в 1995 г.' (год, year). Here the abbreviation 'г.' may signalize a sentence boundary in both cases and further morphological or even syntactic analysis would be necessary to resolve the issue. Our tokenizer can only learn to slightly prefer sentence breaks in the latter case based on the presence of a four-digit number and to slightly disprefer sentence breaks in the former case, if we supplied it with a list of city names.

The final step of the process is to interpret and remove all auxiliary tokens from the stream. Rough tokens are joined at <joinD> and sentences are split at <S>.

11 *Le Zhang*. 2004. Maximum Entropy Modeling Toolkit for Python and C++. Reference Manual.

4.3. Alignment

In CzEng and in UMC the texts are aligned only at sentence level using the hunalign tool[12]. We did not use any additional dictionary, the dictionary was learnt automatically by the tool. In the table 2 we show the statistics about the alignment - how many sentences were aligned 1 to 1, 1 to 2 and so on in all the three languages.

Table 2. Distribution of the alignment types

1-1	**2-1**	**0-1**	**1-2**	**1-0**	**Others**
259599	9074	8551	7686	834	2434
90.1 %	3.1 %	3.0 %	2.7 %	0.3 %	0.8 %

In the future we will provide manual analysis of sentence alignment quality. One of our working hypothesis now is that the alignment accuracy for Czech and Russian, which are closely related languages, is better than that for English and Czech.

5. Conclusion and future work

We have presented the first release of UFAL Multilingual Corpus (UMC 0.1). The corpus contains Czech, Russian and English texts aligned at sentence level for each language pair. We will soon experiment with machine translation between Czech and Russian, to continue the early attempts in the 80's[13] as well as other language pairs.

---

12 **http://mokk.bme.hu/resources/hunalign**

13 *Karel Oliva.* 1989. A Parser for Czech Implemented in Systems Q, in Explizite Beschreibung der Sprache und automatische Textbearbeitung.

This experiment will be also interesting when comparing paradigms used for machine translation between closely related languages. As it was suggested in the earlier project Česílko[14] (MT between Slavic languages, mainly linguistically-based approach), it is not necessary to use such a powerful tool as statistics to reach sufficient translation results, for example, between Czech and Russian or Czech and Polish. Training a statistical MT system on UMC Czech-Russian data will provide us with the opportunity to compare the strategies.

However, the corpus is released as a freely available resource for any kind of linguistic experiments.

[14] *Hajič, J., Homola, P. and Kuboň, V.* : A simple multilingual machine translation system. In *Proceedings of the MT Summit IX*, New Orleans, 2003.