From Subtitles to Parallel Corpora

Mark Fishel,[γ] Yota Georgakopoulou,[δ] Sergio Penkale,[χ] Volha Petukhova,[φ] Matej Rojc,[ξ] Martin Volk,[γ] Andy Way[χ]

[γ] Institute of Computational Linguistics, University of Zurich, Switzerland
[δ] Deluxe Digital Studios, UK [χ] Applied Language Solutions Ltd., UK
[φ] Human Speech and Language Technologies, Vicomtech, Spain
[ξ] Laboratory for Digital Signal Processing, University of Maribor, Slovenia

[γ] {fishel,volk}@cl.uzh.ch [δ] yota.georgakopoulou@bydeluxe.com
[χ] {sergio.penkale,andy.way}@appliedlanguage.com
[φ] vpetukhova@vicomtech.org [ξ] matej.rojc@uni-mb.si

Abstract

We describe the preparation of parallel corpora based on professional quality subtitles in seven European language pairs. The main focus is the effect of the processing steps on the size and quality of the final corpora.

1 Introduction

The present user study is a part of the SUMAT project,[1] which aims at developing an online machine translation (MT) service for subtitles. The project employs the paradigm of statistical MT, which means that large datasets are required for training translation models.

The training data was provided by professional subtitle companies, which create and translate subtitles for movies, TV shows and other video material; they are also the future users of the translation systems planned in the project.

In this paper we will focus on the preparation of parallel corpora on the basis of the provided data. We will describe in detail the problems that arose while producing ready, clean, usable datasets from raw subtitle files, discuss our solutions to those problems and their effect on the size and quality of the final datasets.

2 General Description

The project plans include translation between seven language pairs: English–Dutch, English–French, English–German, English–Portuguese, English–Spanish, English–Swedish and Serbian–Slovenian. Additional monolingual data was provided for language models, but in this paper we will focus on handling parallel data.

Previous work on subtitle translation (Armstrong et al., 2006; Volk et al., 2010) has demonstrated that subtitle-by-subtitle translation can be successful; there are also examples of sentence-based translation for subtitles (Tiedemann, 2009). Sentence-based translation can be linguistically motivated, but just like any other merging/splitting of the subtitles, it introduces additional pre-processing and post-processing steps, which are additional potential sources of error. In the SUMAT project we will compare the different approaches in terms of the final translation quality, but this user study is limited to subtitle-based processing only.

The subtitle companies provided the subtitle files with their original names (following a variety of naming conventions) and for the most part – in their original format. All files were accompanied by their genres and domains. Automatic processing therefore had to start with systematic file renaming, and subsequent format conversion; the following steps were language identification, document alignment, subtitle alignment and finally tokenization and lower-casing. All of these steps are described in more detail in the following sections.

3 Format Conversion

The subtitle files supplied by the subtitle companies included a text-based format, colloquially called *the .txt format* and several binary formats: STL,[2] 890,[3] PAC[4] and the o32/s32/x32 format group.[5] We implemented file format converters for

[1] http://www.sumat-project.eu

[2] http://www.ebu.ch

[3] http://www.cavena.se

[4] http://www.subtitling.com

[5] http://www.softelgroup.com

Format	success rate	#files	#subs ($\cdot 10^3$)
TXT	99.6%	18 381	9 031.1
STL	99.9%	5 074	1 434.5
890	99.1%	1 469	269.2
PAC	98.2%	3 940	1 528.3
Total	99.4%	28 864	12 263.0

Table 1: Format conversion success rates in the raw dataset and the resulting number of files and subtitles after conversion.

all of them, except the latter group, which turned out to be too complicated to support without format specifications and was manually converted to .txt.

The binary formats supported text positioning, coloring, fonts, etc. Although the final translation systems have to preserve such formatting, it will be handled separately from translation. We thus discarded all formatting information in the training data and selected .txt as the common format.

The portion of the data in the .txt format thus required only encoding normalization; the main problem turned out to be the ambiguity of this format, as several different formats were grouped under a common name. All differences occurred in the subtitle time stamps: in addition to the usual index, starting and ending time, some files specified the subtitle duration (sometimes omitting the ending time), or preceded time codes with *TIMEIN* and *TIMEOUT*. A small amount of the files were missing some necessary information (e.g. only the starting time with no duration or ending time).

In contrast, the binary formats have a fixed text encoding. The main problem was caused by the formats without open specifications (890 and PAC), which have custom encoding tables for non-ASCII characters (diacritics, specified after the "carrier" letters, custom characters like non-Latin letters, copyright symbols, custom quote marks, etc.) and were reverse-engineered to implement format conversion.

Table 1 presents format frequencies in the dataset, conversion success rates and results; only 0.6% of the files were lost during this step.

4 Language Identification

Automatic language identification was required to check whether every subtitle file indeed contained subtitles in the specified language pair and to steer document alignment.

We performed language identification using the Lingua::Ident package,[6] which implements a character trigram probability-based algorithm. The OpenSubtitles v.2 corpus (Tiedemann, 2009) was used to estimate the language signatures.

During data acquisition it turned out that some subtitles in languages unconnected with the project had ended up in the dataset, the most frequent of which were Italian and Danish; to detect such files separately, corresponding signatures were added.

After manual inspection of the language identification results, we determined that the majority of languages was identified correctly. The only small problem consisted of a couple of dozen files with gibberish or unconventional content (like "asdfasfd", "qwertyqwerty", "whoop whoop! shhhufff! ding dong!") and empty files.

The results of language identification against the manually specified languages or language pairs are presented in Table 2. Comparing the number of subtitles in the correctly placed files to the conversion results, the total subtitle loss at this point is around 95 000 subtitles, or 0.8% percent of the converted subtitles. However, given the different number of files in the two languages of every language pair, further loss is going to be greater.

5 Document Alignment

The next step was to identify pairs of subtitle files (documents) that were translations of each other. The fastest way to perform document alignment is based on the file names, since this does not involve reading the contents of the files. For that we collected and documented the file naming conventions in the dataset, discovering the following patterns:

- file names of the aligned pair differing only in the language (e.g. "Movie_Title_**en**.txt" and "Movie_Title_**fr**.txt")
- file names starting with the same 4-to-5-digit ID (e.g. "**12345**_en.txt" and "**12345**nl6.txt")
- file names containing the same 9-symbol ID (digits and capital letters), followed by a 3-character language code (e.g. "Deutsche Titel_**AXGM0102A_DEU**.PAC" and "English Title-**AXGM0102A_ENG**.PAC")

[6] http://search.cpan.org/~mpiotr/Lingua-Ident-1.7/

Manually Specified	Automatically Identified	#files	#subs ($\cdot 10^3$)	Manually Specified	Automatically Identified	#files	#subs ($\cdot 10^3$)
English–Dutch	English	1 606	863.0	English–Spanish	English	1 694	849.0
	Dutch	1 617	833.9		Spanish	1 711	851.3
	Other	8	3.5		Other	6	2.3
English–French	English	2 369	1 066.4	English–Swedish	English	1 100	636.5
	French	2 376	1 067.7		Swedish	1 157	635.5
	Other	20	7.2		Other	10	5.6
English–German	English	6 919	1 958.7	Serbian–Slovenian	Serbian	402	233.7
	German	5 124	1 884.7		Slovenian	391	175.1
	Other	14	2.9		Other	2	1.5
English–Portuguese	English	1 145	560.3	**Total**	Correct-1	15 235	6 167.5
	Portuguese	1 142	552.4		Correct-2	13 518	6 000.6
	Other	4	1.7		Other	64	24.8

Table 2: Results of automatic language identification, contrasted with manually specified language pairs; the "Other" languages do not include Italian and Danish, as these are not covered in the SUMAT project.

- 8-symbol file names starting with the same movie ID (4 letters) and a 2-character language code (e.g. "MISS**EN**DC.txt" and "MISS**NL**DV.TXT")

Even while comparing file names, it is inefficient to try to align a document to all other documents, so we trimmed the search space by comparing only files within the same genre and domain.

After the initial file name-based processing, 52.1% of the subtitle files specified as parallel were identified as such. We processed the remaining files with a time code similarity-based approach to document alignment: two documents are considered parallel if at least 90% of the time codes correspond to each other.[7]

As a result of joint file name- and subtitle-based processing, we discovered alignments for 68.6% of the documents. We processed the remaining third of the dataset manually, which resulted in detected file pairs for 83% of all the files specified as parallel; the remaining 17% were added to the corresponding monolingual datasets.

The resulting numbers of aligned document pairs and subtitles are summarized in Table 3; the coverage of document alignment in terms of subtitles is 87.9% of the converted parallel dataset.

Manual reviewing of the unaligned files, initially specified as parallel, revealed that a large amount of the files were missing their counterpart.

Another problem with document alignment arose from subtitle files, which were translated and saved in parts, indicating a many-to-one document correspondence; these occurred in the English–German language pair. As a result only the first (English) part of the translation was aligned with the full (German) document, putting the other parts into the monolingual datasets. This reflects negatively on the number of subtitles in this language pair after document alignment.

[7] see the next section on subtitle alignment for more details

Language pair	#file pairs	#subs ($\cdot 10^3$)
English–Dutch	1 530	831.9 / 801.2
English–French	2 232	989.4 / 989.5
English–German	4 009	1 337.3 / 1 520.2
English–Portuguese	1 126	544.8 / 547.0
English–Spanish	1 641	810.9 / 811.9
English–Swedish	1 055	609.1 / 594.3
Serbian–Slovenian	380	219.1 / 169.7
Total	11 973	5 342.6 / 5 433.9

Table 3: Document alignment results: the number of file pairs and subtitles per language pair.

6 Subtitle Alignment

The main state-of-the-art work on subtitle alignment (Tiedemann, 2007, 2009) aligned corpora at the sentence level, so we had to come up with an approach of our own to align subtitles.

The main assumption in the planning phase of the SUMAT project was that almost all translated subtitles would have directly matching time codes, which would make subtitle alignment trivial. It turned out, however, that several issues made this task more "interesting": some companies translate subtitles without preserving the time code template, which results in more loose translations and many-to-one correspondences between subtitles. Also due to a different movie cut or version, portions of the translated subtitles can be missing and

Language pair	#file pairs	#sub pairs ($\cdot 10^3$)	#tokens ($\cdot 10^6$)
English–Dutch	1 515	688.7	6.89 / 5.75
English–French	2 202	944.1	9.33 / 8.72
English–German	3 841	954.9	9.20 / 8.01
English–Portuguese	1 123	523.4	5.16 / 4.60
English–Spanish	1 613	779.5	7.59 / 6.83
English–Swedish	1 047	577.5	5.87 / 4.86
Serbian–Slovenian	380	111.9	1.25 / 1.50
Total	11 721	4 580.0	45.29 / 40.27

Table 4: Subtitle alignment results: the number of aligned file pairs, subtitle pairs and tokens per language pair in the final corpora.

subsequent portions shifted.

To account for these complications, we designed a dynamic programming algorithm, based on subtitle shift similarity: subsequent subtitle alignments with a certain shift are endorsed if the shift stays almost constant. The same algorithm checks for many-to-one matches; merging is achieved by using the starting time code of one subtitle and the ending time code of a subsequent subtitle.

To assess the quality of the alignments, we aligned small held-out datasets of approximately 500 parallel subtitles per language pair manually. The average precision and recall of the alignments were 0.94 and 0.91, respectively.

As a final step we tokenized the aligned subtitles and converted them to lower-case. Serbian and Slovenian data was tokenized with a tool from the PLATTOS system (Rojc and Kacic, 2007) and the remaining data with the Moses toolkit[8] tokenizer.

The resulting sizes of the final parallel corpora are presented in Table 4. According to the numbers the final corpora constitute a total of 85.0% of the document-aligned dataset and 74.7% of the unaligned, converted dataset. However, this estimate is overly pessimistic, since many subtitles were merged as a result of 1-to-N subtitle alignment. Data loss rates per language pair range from over 50% (German, Serbian) to 5% (Portuguese), although these estimates are exaggerated as well; it is important to note that the different rates per language are caused by the characteristics of the supplied subtitles, and not the language itself.

7 Conclusions

The SUMAT project has started by turning raw subtitle files into clean parallel corpora, usable for training statistical translation models. We have described the problems that were encountered during the preparation of the files as well as our solutions.

The total data loss from raw subtitle files to final parallel corpora is below 25% and the corpus sizes are mostly sufficient for training translation models.

The main reason for data loss is human error, manifesting as incorrectly specified subtitle language pairs and file format inconsistencies. Added to this, the subtitle alignment algorithm was unable to fully cope with loose translations and subtitle time correspondences.

The next step in the project is training the baseline MT systems for all translation directions, thus evaluating the collected datasets in practice.

[8] `http://www.statmt.org/moses`

References

Armstrong, S., C. Caffrey, M. Flanagan, D. Kenny, M. O'Hagan, and A. Way. 2006. Leading by example: Automatic translation of subtitles via EBMT. *Perspectives*, 14(3):163–184.

Rojc, M. and Z. Kacic. 2007. Time and space-efficient architecture for a corpus-based text-to-speech synthesis system. *Speech Communication*, 49(3):230–249.

Tiedemann, J. 2007. Improved sentence alignment for movie subtitles. In *Proceedings of RANLP-07*, pages 582–288, Borovets, Bulgaria.

Tiedemann, J. 2009. News from OPUS – a collection of multilingual parallel corpora with tools and interfaces. In *Proceedings of RANLP-09*, pages 237–248, Borovets, Bulgaria.

Volk, M., R. Sennrich, C. Hardmeier, and F. Tidström. 2010. Machine translation of TV subtitles for large scale production. In *Proceedings of the 2nd Joint EM+/CNGL Workshop on "Bringing MT to the User"*, pages 53–62, Denver, CO.