Exploiting of the timing information in subtitle-like parallel multilingual data

Marek Boháč and Michal Rott

Technical Univerzity of Liberec
Institute of Information Technology and Electronics,
Studentská 2/1402, 461 17 Liberec, Czech Republic
{marek.bohac,michal.rott}@tul.cz

Abstract

The demand for making documents accessible in multiple languages increases, as the globalization of our society continues. Typical example is provided by the European Union, whose institutions produce vast amounts of documents, all of them being translated into all official languages of the member countries. When the document is not just a text, but it has a multimedia content, a need arises to provide the translation with correct timing information (to enable navigation in indexed archives). The timing is as time-consuming task as the translation itself, so it is reasonable to exploit existing timing information (from already completely processed languages) to spare human-time when other languages are being processed. We addressed this task in our previous work (Boháč et al., 2015) where we developed a scheme able to pair subtitles between two languages. In this paper we address the question if the using of multiple source languages can improve the processing of one target language documents. We also provide a deeper insight in some errors often occurring in the transcription timing. For some of them we propose automatic solution, namely for the additive offset.

1. Introduction

In the last decade our society turns into the Information Society. The production of various documents greatly increases. There are so many information sources, that single human is not able to follow all the main news. This brings focus to two main tasks: i) building of media monitoring systems able to find documents of interest (Cook et al., 1999; Heeren et al., 2008) and ii) construction of systems which group the corresponding data from different sources and languages (EuroNews, 2015). One of the most discriminating factors is the number of languages in which the documents of interest are produced. Only the frame of the European Union covers 24 official languages. There are two possible ways how to break the language barrier.

The first one relies on machine translation applied straightforwardly between a pair of languages. This group of solutions may work well if both languages are members of the same language family. An example is shown in (Lyu et al., 2008) where the authors find best fitting document transcription (in Mandarin) for the spoken news (in Taiwanese). The problem comes when the languages have different origin and thus the word order strongly differs. The impact of such effects was shown in our recent work (Boháč et al., 2015), where the different nature of target and source language caused many troubles.

The second way to break the language barrier comes from the idea that more language variants of the same source document may exist in parallel. The European society not only brings the complication with many languages but also provides a solution. All the official documents are correctly translated (EuroArchive, 2015), the shared culture creates subtitles for the same series and movies and important political and social speeches are broadcast with translations in the main languages (languages with the biggest usage in the Europe). This provides a great opportunity for less-resourced (less used) languages to find ways how to utilize these already existing parallel multilingual data for their own purposes.

These application areas may be processing of official European documents (e.g. subtitling of official speeches in the European Parliament and similar institutions), which is the first step to enabling a sophisticated search for wide public. The second purpose may be creation of subtitles or closed captions (needed for hearing-impaired people), where the work is usually done by a volunteer community. This approach may also be used by the scientific community dealing with the spoken word processing. For them, the lack of training acoustic data is usually the bottleneck constraining application of many technologies in the real life. Reduction of human work is therefore welcome.

The first and second mentioned applications are quite clear. If we can extract timing information for our subtitles from already existing data, we can greatly reduce the time demands (cut the costs). It would be also wonderful if we were able to create one compact subtitle pack for one document, which would be suitable for all the potential users (deaf people). This could speed up production of new subtitles in new languages (so the volunteer community would be able to make more translations instead of timing the translated ones).

The speech processing applications are a bit wider field. Well-timed subtitles can be used as a source of training (testing) data for voice activity detectors, or as a source of rarely observed non-speech events (laughing, shooting, car crash sounds etc.). If we can find dabbed data, it can be used to train GMM models for language identification task. And finally a combination of film dabbing and its translation together with timed subtitles in other languages can be used as good acoustic model data source in under-resourced languages (Nouza and Boháč, 2011; Nouza et al., 2012; Boháč et al., 2014).

The ultimate goal can be defined as creating a media monitoring system able to cluster the documents of the same content (in different languages) together, followed by a subsystem working inside the topic cluster which would provide subtitles for the multimedia sub-content of the cluster. These tasks are partially solved (EuroNews, 2015; Lyu et al., 2008), but no solution incorporates both.

In this paper we focus on the task of processing multilingual timed speech information to obtain timing information for the text in another language. We also provide some more insight in the crucial transcription-timing issues.

In the next section we explain the proposed method and summarize the most important issues of the subtitle timing. The experimental evaluation is provided in section 3. In the last section we provide conclusions and future work.

2. Proposed Method

2.1. General notes to transcription timing

Before we propose our solution for the subtitle timing it is essential to state some basic assumptions about the subtitling and associated sources of inaccuracies. Some of the problems come from the fact, that the authors of our experimental data are not professionals but fan-community members, so they do not follow standards as they are set in big companies[1,2]. This leads to insertions of such subtitles like nicknames of the subtitle authors, short advertisements. Different marking of nonspeech events ranges from [: music :] to ♪...♪ or is completely omitted. Subtitles also differ in the way of marking speaker turns in longer subtitles ("–") or marking of interrupted utterances ("...").

Second important source of inaccuracies is the manual timing of the subtitles. It naturally leads to some variance in the timing as can be seen in Fig. 1. The image shows the histogram of time-differences between original subtitles and subtitles with carefully corrected timing. For this process we used universal transcription software NanoTrans (Šeps, 2013). However some subtitles are shifted, the subjective quality of the original subtitles is very high.

Last source of errors comes from the data coupling. It often happens that the version of subtitles does not fit the media file – the times may be affected by additive offset (usually caused by different length of the signature tune etc.). Another source of errors comes from presence of advertisements in some of the media versions (which shifts following subtitles) or some scenes may be deleted because of different legislative frame in some countries (e.g. Nazi symbols in Germany). Histogram of such subtitle timing differences is shown in Fig. 2. It shows us standard distribution of timing differences in one part of the media file and a second peak caused by a deleted scene causing approx. 2s delay of subtitles in the second part of the episode.

2.2. Additive offSet detectors

As mentioned in the previous paragraph, one of the main challenges in exploiting existing subtitle timing is it's additive offset. To deal with it we propose two methods for automatic determination of offset between two subtitles. The first approach assumes there is a strong correlation in time structure of speech and nonspeech regions of the subtitles (see 2.2.1.). The second approach pairs the utterances by its translated textual content (see 2.2.2.).

[1] *http://www.ofcom.org.uk/static/archive/itc/itc_publications/codes_guidance/standards_for_subtitling/subtitling_1.asp.html*

[2] *http://www.bbc.co.uk/guidelines/futuremedia/accessibility/subtitling_guides/online_sub_editorial_guidelines_vs1_1.pdf*

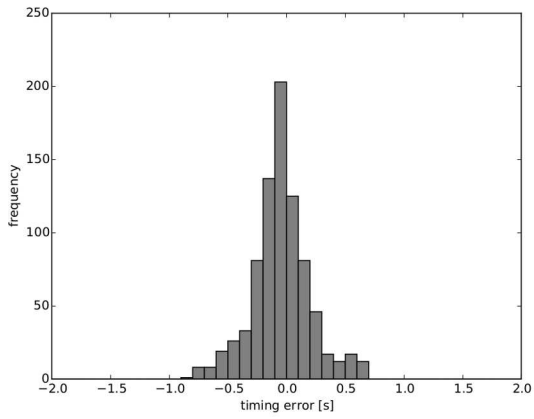

Figure 1: Histogram of the subtitle beginning differences between manually timed subtitles and reference subtitles.

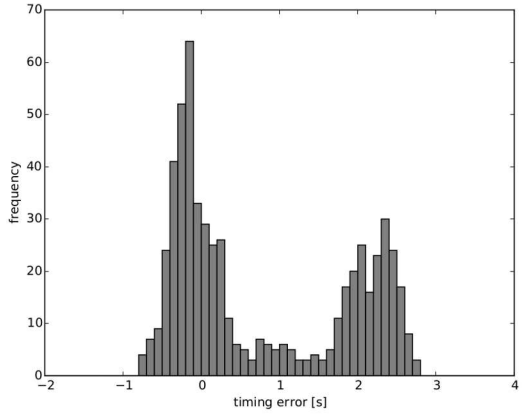

Figure 2: Example of timing error histogram.

2.2.1. Time structure cross-correlation

The cross-correlation approach finds speech/nonspeech regions in the subtitles. This classification of the signal gives us a rectangular function of time which can be compared to the second subtitle signal (as shown in Fig. 3). We can choose if the logical one state is the presence of speech in the signal or the presence of the nonspeech (noise). The distance between the cross-correlation function (CCF) maximum value and index corresponding to the zero-shift of the subtitles is the detected offset.

The inaccuracies in the subtitle timing (as shown in Fig. 1) and presence of "floating subtitles" (e.g. advertisements or names of the subtitle makers) may lead to improper position of the CCF maximum. This problem can be partly solved by limiting the examined area in the CCF to the surrounding of the "zero-shift point" but this limitation also means we limit the detectable offset just to some range. These limitations of the approach can not fully overweight the fact it does not employ the textual content (we do not have to translate the content).

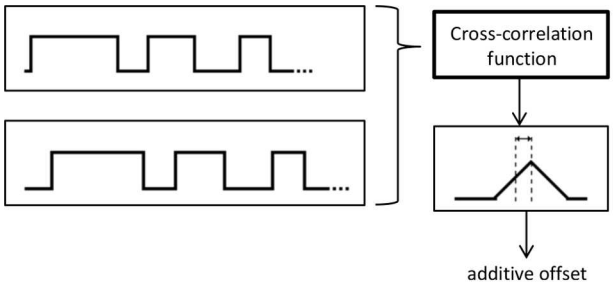

Figure 3: Cross-correlation based offset detector.

2.2.2. Textual-content based alignment

This offset detector uses the alignment between two sets of subtitles which are aligned by its textual content. The method demands the translation[3] into the same language and compares the similarity of the subtitles by the intersection of word sets of concrete utterances. The computation is derived from the dynamic programing principles (Wagner and Fischer, 1974) where the similarity metric is given by eq. 1. $|A \cap B|$ stands for the words found in both source and target utterances, while $|A|$ represents the number of words in the source utterance. Details of this alignment method can be find in (Boháč et al., 2015) and will be referred to as 1–to–1 alignment. When the corresponding utterances are known it is easy to compute the histogram of timing differences. The offset is given as the maximum histogram value in the 0.25s interval around the histogram center of gravity. This approach demands the ability to translate the subtitles but is more robust than the CCF approach and the detectable offset is not limited.

$$similarity = \frac{|A \cap B|}{|A|} \tag{1}$$

2.3. Proposed scheme

The proposed scheme (depicted in Fig. 4) starts with the conversion of all available subtitle formats (e.g.: .srt, .sub) into one general data structure unifying the encoding, time format, special symbols and marks.

The next step is the recovery of sentences which is based mostly on the punctuation and on the speaker changes in one subtitle (which are usually marked by the dash). Special symbols and content labels (e.g.: music, laughing) annotated for the hearing impaired people are cleared out. The sentence recovery is very important for the 1–to–1 alignment as shown in (Boháč et al., 2015).

The 1–to–1 alignment of the subtitles has 3 possible set ups. If we denote the language we want to time the *target* language and the other available subtitles the *source* languages we can align:

- target language – to – the source language translated to the target one (trgt↔src2trgt)
- target language translated to the source one – to – the source language (trgt2src↔src)

[3]We use the GoogleTranslate service in our experiments: https://translate.google.com/

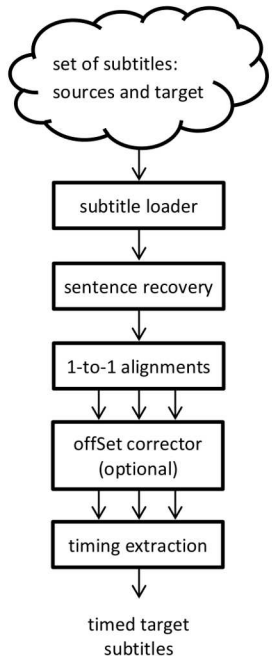

Figure 4: Scheme of the proposed method.

- target language translated to a third language – to – the source language translated to a third language (trgt2xx↔src2xx)

In our experiment the target language is Czech, source languages are Polish, English, Slovak, Holland, Spanish and French. The "third" language is English.

Every 1–to–1 alignment generates a set of time stamps for the target utterances (some of them may be untimed if the method does not find the fitting utterance). This means that *M* sources generate up to *M* time stamps for every target utterance. As we mentioned in subsection 2.1., these time stamps may be burdened by the additive offset error, which we can try to compensate.

If we want to compensate the offset we define the reference timing as the median of concrete time stamps per utterance. Separately for every source language we compute the offset via the method proposed in subsection 2.2.2. and adjust the corresponding time stamps.

The last step is the time stamp extraction. If there is at least one time stamp available for the utterance the median of the available time stamps is the output time stamp (so we have a chance to ignore incorrect time stamps). If there is no time stamp, we mark the utterance as untimed and it can be localized by the timing of its surroundings.

3. Experimental Evaluation

3.1. Experimental data and evaluation metrics

For the experimental evaluation we gathered subtitles of two series. First of them is the House of Cards (*HoC*). We process 8 episodes (45min each) with 5 source subtitles for each. As the series is from the political environment there are "standard dilogues" with full sentences. The typical utterance duration ranges from 1.5s to 2.5s.

The second series is the South Park[4], which we denote *SoPa*. We process 10 episodes (20min each). We have 5 source subtitles again. The utterances are much shorter than in HoC and "speakers" change very quickly. The typical utterance duration ranges from 1.0s to 1.8s.

The reference data were obtained by manual verification of the Czech subtitles (the additive offset was proposed automatically, but verified). As can be seen in Fig. 1, manually made subtitles should have timing error less then 1.0s. In case our system was used for pre-timing we added some experiments up to 2.0s tolerance. The accuracy (eq. 2) is computed as the number of time stamps within the given timing tolerance ($times_{OK}$) divided by the number of utterances in the reference subtitles (N_{ref}). We also count the number of utterances without timing.

$$accuracy = \frac{times_{OK}}{N_{ref}} \tag{2}$$

3.2. Experimental results

In the following tables we evaluate the performance of the proposed scheme. Tables 1 and 2 summarize the results reached for the HoC and SoPa series. As their structure (and results) are quite different, we show the results separately. Every cell contains results of two sub-experiments: with / without applying the block of additive offset correction (marked as *(optional)* in the Fig. 4).

Tables 3 and 4 show the baseline performance. It shows the accuracy of subtitle beginnings and the number of untimed utterances (in brackets). As the results obtained by concrete source data greatly differs, we show the worst, the best and average performance of the 1-to-1 alignment.

Table 1: Accuracy of subtitle timing - HoC series

	tol. [s]	ACC_{beg}[%]	ACC_{end}[%]	untimed
trgt↔src2trgt	0.5	73.4 / 75.0	69.1 / 69.3	6 / 15
	1.0	88.2 / 88.6	86.4 / 86.7	6 / 15
	1.5	92.9 / 93.1	92.5 / 92.9	6 / 15
	2.0	94.8 / 94.7	95.2 / 95.1	6 / 15

	tol. [s]	ACC_{beg}[%]	ACC_{end}[%]	untimed
trgt2src↔src	0.5	76.4 / 78.0	69.4 / 70.6	5 / 14
	1.0	91.2 / 91.6	87.7 / 88.1	5 / 14
	1.5	94.6 / 94.5	93.5 / 93.6	5 / 14
	2.0	95.9 / 95.8	95.6 / 95.5	5 / 14

	tol. [s]	ACC_{beg}[%]	ACC_{end}[%]	untimed
trgt2en↔src2en	0.5	78.6 / 78.5	69.3 / 69.2	23 / 26
	1.0	91.3 / 91.3	87.2 / 87.2	23 / 26
	1.5	93.9 / 93.9	92.4 / 92.3	23 / 26
	2.0	95.2 / 95.2	94.6 / 94.5	23 / 26

4. Conclusion & Future work

In subsection 2.1. we summarized the most limiting properties of the manually created subtitles. Based on these restrictions we proposed two approaches for dealing with the subtitle additive offset (subsection 2.2.). The main focus of the paper is paid to the task of merging the information from multiple different-language source subtitles in order to time target language subtitles. We presume that this multi-source approach should outperform our previous approach using only one input subtitles (Boháč et al., 2015). Thus this previous approach is considered to be the base-line. All experiments are conducted with three language setups, so we can estimate which "direction of translation" provides the best results.

The proposed solution is evaluated on two data subsets (HoC and SoPa). In the case of HoC series (tables 1 and 3) the new approach completely outperforms the

[4]http://southpark.cc.com/

Table 2: Accuracy of subtitle timing - SoPa series

	tol. [s]	ACC_{beg}[%]	ACC_{end}[%]	untimed
trgt↔src2trgt	0.5	37.6 / 47.8	37.5 / 47.8	22 / 22
	1.0	55.2 / 59.7	56.9 / 61.4	22 / 22
	1.5	64.2 / 67.0	65.4 / 68.5	22 / 22
	2.0	69.3 / 71.7	70.6 / 72.9	22 / 22

	tol. [s]	ACC_{beg}[%]	ACC_{end}[%]	untimed
trgt2src↔src	0.5	38.8 / 50.1	39.0 / 48.9	27 / 27
	1.0	57.9 / 63.4	57.3 / 62.6	27 / 27
	1.5	66.3 / 70.1	65.8 / 69.8	27 / 27
	2.0	71.9 / 74.7	71.6 / 74.5	27 / 27

	tol. [s]	ACC_{beg}[%]	ACC_{end}[%]	untimed
trgt2en↔src2en	0.5	37.6 / 44.6	36.6 / 42.9	61 / 68
	1.0	52.6 / 53.8	52.2 / 53.2	61 / 68
	1.5	57.9 / 58.1	57.4 / 57.2	61 / 68
	2.0	60.5 / 60.5	60.6 / 60.1	61 / 68

Table 3: Accuracy of 1-to-1 alignment - HoC series

	tol. [s]	worst	average	best
trgt↔src2trgt	0.5	46.1% (48)	59.2% (49)	68.7% (42)
	1.0	69.1% (60)	77.0% (49)	83.0% (42)
	1.5	78.1% (66)	83.4% (49)	87.8% (39)
	2.0	81.7% (66)	86.2% (49)	89.9% (42)

	tol. [s]	worst	average	best
trgt2src↔src	0.5	47.3% (62)	60.8% (50)	70.9% (38)
	1.0	70.1% (64)	78.9% (50)	84.5% (44)
	1.5	80.4% (69)	85.2% (50)	89.4% (41)
	2.0	83.7% (72)	87.8% (50)	91.1% (40)

	tol. [s]	worst	average	best
trgt2en↔src2en	0.5	48.7% (51)	60.8% (42)	69.3% (47)
	1.0	71.7% (61)	78.8% (42)	83.2% (44)
	1.5	82.2% (70)	85.1% (42)	87.7% (38)
	2.0	85.0% (68)	87.5% (42)	90.0% (38)

Table 4: Accuracy of 1-to-1 alignment - SoPa series

trgt↔src2trgt

tol. [s]	worst	average	best
0.5	5.3% (27)	35.5% (31)	58.6% (34)
1.0	7.5% (26)	45.1% (31)	70.7% (31)
1.5	8.5% (32)	50.3% (31)	75.8% (31)
2.0	9.3% (32)	54.0% (31)	81.0% (25)

trgt2src↔src

tol. [s]	worst	average	best
0.5	1.3% (31)	36.5% (34)	61.9% (36)
1.0	3.9% (31)	46.5% (34)	74.8% (34)
1.5	6.3% (32)	51.8% (34)	79.5% (33)
2.0	8.4% (32)	55.1% (34)	83.4% (16)

trgt2en↔src2en

tol. [s]	worst	average	best
0.5	7.3% (37)	40.0% (33)	64.1% (36)
1.0	8.6% (37)	47.8% (33)	76.5% (36)
1.5	9.2% (38)	51.0% (33)	80.7% (34)
2.0	9.6% (37)	53.1% (33)	83.8% (34)

baseline in both criteria - the overall accuracy and the number of untimed utterances. In the case of SoPa series (tables 2 and 4), the number of untimed utterances is reduced as well. The overall accuracy of the new scheme outperforms the average accuracy of the base-line but can be outperformed by the best one source language. This is caused by the very short and quickly changing utterances of the SoPa subtitles (as mentioned in subsection 3.1.).

We were quite surprised that the optional offset correction module did not improve the final performance. This also means that median filtering of the available time stamps is a robust solution and can operate even with slightly mistimed data.

The performance of 3 compared language setups is almost the same. The only relevant conclusion is that trgt2en↔src2en setup performs slightly worse. It makes one more translation than the other setups so there is higher probability of translation errors.

The performance of the method make us believe it is suitable for fully automatic timing of subtitles and other documents. This can spare a lot of human time when preparing closed captions or at least it can make a pre-timing for manual verification.

We plan to use this approach for automatic preparation of parallel multilingual corpora. It would also be interesting if we were able to define a common frame for multiple single-language subtitles management able to assist the process of adding new language into this frame.

Our system is not publicly available yet for two reasons. We are going to incorporate it as a plug-in into some already existing transcription editor e.g. (Šeps, 2013). As we use a third-party translation tools we must find a solution which is in accordance with the license agreements.

5. Acknowledgment

This work was supported by the Student Grant Scheme (SGS 2015) at the Technical University of Liberec.

6. References

Boháč, Marek, Michaela Kuchařová, Zoraida Cajellas, Jan Nouza, and Petr Červa, 2014. A cross-lingual adaptation approach for rapid development of speech recognizers for learning disabled users. *EURASIP Journal on Audio, Speech, and Music Processing*.

Boháč, Marek, Michal Rott, and Karel Blavka, 2015. On automatic cross-lingual subtitle timing. In *Electronics, Control, Measurement, Signals and their Application to Mechatronics*. IEEE.

Cook, Gary, James Christie, Daniel PW Ellis, Eric Fosler-Lussier, Yoshi Gotoh, Brian Kingsbury, Nelson Morgan, Steve Renals, Tony Robinson, and Gethin Williams, 1999. An overview of the sprach system for the transcription of broadcast news. In *Proceedings of the DARPA Broadcast News Workshop, February 28-March 3, 1999, Hilton at Washington Dulles Airport, Herndon, Virginia*. Information Technology Laboratory, National Institute of Standards and Technology.

EuroArchive, 2015. http://europa.eu/publications/libraries-archives/index_en.htm. [Online; accessed 9-9-2015].

EuroNews, 2015. http://m.euronews.com/en/. [Online; accessed 9-9-2015].

Heeren, Willemijn, Roeland Ordelman, and Franciska De Jong, 2008. Affordable access to multimedia by exploiting collateral data. In *International Workshop on Content-Based Multimedia Indexing, 2008*. IEEE.

Lyu, Dau-Cheng, Ren-Yuan Lyu, Yuang-Chin Chiang, and Chun-Nan Hsu, 2008. Cross-lingual audio-to-text alignment for multimedia content management. *Decision Support Systems*, 45(3):554–566.

Nouza, Jan and Marek Boháč, 2011. Using tts for fast prototyping of cross-lingual asr applications. In *Analysis of Verbal and Nonverbal Communication and Enactment*. Springer, pages 154–162.

Nouza, Jan, Petr Cerva, Jindrich Zdansky, and Michaela Kucharova, 2012. A study on adapting czech automatic speech recognition system to croatian language. In *ELMAR, 2012 Proceedings*. IEEE.

Šeps, L., 2013. Nanotrans; editor for orthographic and phonetic transcriptions. In *Telecommunications and Signal Processing (TSP), 2013*.

Wagner, R.A. and M.J. Fischer, 1974. The string-to-string correction problem. *Journal of the ACM*, 21:168–173.