INTERSPEECH 2016: Show & Tell Contribution
September 8–12, 2016, San Francisco, USA

STON: Efficient Subtitling in Dutch Using State-of-the-Art Tools

Lyan Verwimp[1]*, Brecht Desplanques*[2]*, Kris Demuynck*[2]*,*
Joris Pelemans[1]*, Marieke Lycke*[3]*, Patrick Wambacq*[1]

[1]ESAT, KU Leuven, Belgium,
[2]ELIS, Ghent University - iMinds, Belgium,
[3]VRT, Belgium,
firstname.lastname@{esat.kuleuven.be,elis.ugent.be,vrt.be}

Abstract

We present a modular video subtitling platform that integrates speech/non-speech segmentation, speaker diarisation, language identification, Dutch speech recognition with state-of-the-art acoustic models and language models optimised for efficient subtitling, appropriate pre- and postprocessing of the data and alignment of the final result with the video fragment. Moreover, the system is able to learn from subtitles that are newly created. The platform is developed for the Flemish national broadcaster VRT in the context of the project STON, and enables the easy upload of a new fragment and inspection of both the timings and results of each step in the subtitling process.

Index Terms: automated subtitling, speaker diarisation, acoustic modelling, language modelling

1. Introduction

In Flanders, the broadcast networks are obliged to provide subtitles to help the deaf and hard of hearing, and this for the majority of their programmes. The production of subtitles is very labour intensive, but with support from speech and language technology (SLT) the efficiency can be improved substantially. In the project *STON* (*Spraak- en Taaltechnologisch Ondertitelen in het Nederlands* – Dutch subtitling using speech and language technology) initiated by the Flemish national broadcaster (VRT), we have developed an integrated and modular platform that combines several SLT tools in a flexible and user-friendly subtitling application. An important design requirement is that the system is able to automatically learn from already produced subtitles such that its performance will increase throughout its lifetime. Given the VRT's very high quality standards, the system was not designed to replace the human subtitler entirely, but to provide a substantial efficiency gain, requiring only limited human interventions on the automatically produced result. The main use cases for the project are high volume productions with high to medium quality speech such as documentaries, news and web content. The system is also designed to use screenplay information (a script that is more or less followed by the speakers) if available. In that case, script and background language model are combined, resulting in a recogniser that primarily follows the script and only falls back to automatic speech recognition (ASR) if the audio deviates too much from the script.

The system contains several modules: (1) a user interface that implements the subtitling workflow, (2) a module to import screenplay information (dialogues, voice-over and relevant metadata), (3) robust automatic audio segmentation (AAS), (4) ASR, and (5) a synchronisation module that splits the ASR output into well-formed subtitles, optionally replacing ASR sentences with the corresponding sentences from the script (with timestamps derived from the ASR output).

In the remainder of this paper, we will focus on the ASR related aspects of the system. We first describe how text data such as language model (LM) training data, scripts and the ASR output are processed (section 2). Next, the techniques used for AAS (section 3), acoustic modelling (AM) (section 5) and language modelling (section 4) are described. We conclude with some results and an outlook to future improvements (section 6).

This research is funded by IWT–INNOVATIEF AANBESTEDEN and VRT in the STON project. The authors wish to thank the project partners for their contributions to the STON subtitling architecture.

2. Text pre- and postprocessing

Text preprocessing is required for the LM training material and for screenplays. The text normalisation is based on the one described in [1], but several changes were made: capital correction was improved, short 'garbage' words (that easily lead to confusion in recognition) are removed and fillers such as *uh* are treated in a class-based way such that semantically and/or acoustically similar fillers are collapsed in the LM.

The ASR output is a word stream augmented with markers for plausible sentence breaks, fillers and silences. A postprocessing step converts this to a format that is more suitable for subtitles by making the sentence-final punctuation explicit, capitalising sentence-initial words, writing long numbers in digits, converting time indication and percentages into a standardised form, and compounding words [1] whenever appropriate.

3. Audio segmentation

The AAS incorporates three main components: speech/non-speech segmentation, speaker diarisation and language identification. The speech/non-speech system detects long non-speech intervals ($>$1s) that can be discarded in the further processing of the audio stream. Non-speech intervals can contain music and strong background sounds such as applause and street noise, so we rely on a model-based approach [2] to detect these segments.

Speaker diarisation deals with the "who-spoke-when?" problem. The objective is to assign a speaker label to every speech segment. A segmentation stage splits the audio stream into homogeneous segments, and a subsequent clustering stage groups the generated segments into clusters representing single speakers. We use an iVector-based method for both stages [3]. The speaker diarisation allows us to add informative colour codes to the generated subtitles and to profit from speaker adapted models during speech recognition. In addition, the speaker change points coincide with sentence boundaries and thus deliver useful information to the LM.

Many TV programmes in Flanders comprise multiple languages. Dubbing is a rare practice and most foreign speech segments are subtitled. To cope with this scenario we use a

Copyright © 2016 ISCA

language recognition module based on language factor extraction [4] to detect Flemish, English, French and German segments. The current approach is adaptive to speaker accents (see [4]) and assumes that each speaker uses one language only. The Flemish segments are forwarded to the speech recogniser. Foreign speech segments are discarded as they will be translated and transcribed by an interpreter anyhow.

4. Language modelling

Initially, we envisaged language model adaptation where smaller in-domain LMs, generated automatically based on previous subtitles, are interpolated with a larger background LM. However, tests have shown that this does not result in significant improvements. We attribute this to the fact that the large background LM is trained on 1.2B words from newspapers and magazines [1], which is a good match for our use cases (documentaries and news). Currently, we use a 4-gram LM with interpolated modified Kneser-Ney smoothing and a vocabulary of the 100k most frequent words for scripted programmes or 400k for non-scripted programmes. Given that speed is important and the available resources are limited, we refrained from using a larger 5-gram (no significant improvements) or neural network based LMs (no efficient implementation yet).

The subtitling practice will generate new material that can be used to enrich or further adapt the models. In order to quickly update the LM and lexicon, new words are extracted and assigned to a class (e.g. capitalised words, unknown words, the semantic head of the word ...) that is already present in the LM. Updating the LM with new n-grams can be done overnight or during the weekend, such that the subtitler does not have to wait too long every time one or more new words are detected. Pronunciations for new words are generated by a G2P module [1].

5. Acoustic modelling & decoding

The ASR system was built with the SPRAAK toolkit [5] and started from the system developed in [1]. The existing GMM-based baseline system employs 3873 tied states to model the 49 three-state cross-word triphones (46 phones, silence, garbage and speaker noise) and 1 single-state triphone (short schwa). A new DNN-based AM was created as well, using the exact same training material. Since SPRAAK currently lacks GPU support for training DNNs, Kaldi was used for training the AM, and the resulting DNN was converted to SPRAAK's internal format. The DNN takes 11 frames as input (396 features) and combines 6 sigmoid-based hidden layers with 1024 nodes with a softmax output layer to model 4101 tied states. For both systems, the speaker adaptation is limited to vocal tract length normalisation followed by spectral mean normalisation [1].

The recognition itself consists of the following steps: (1) add new words to the lexicon based on the script (using the G2P from [1]) and combine lexicon with tied-state information in a compact finite state transducer (this optional step takes approximately 2 min); (2) the main recognition pass which creates the word lattice; (3) a lattice rescoring pass, using the acoustic scores from the lattice and with the same LM – this pass takes very little time and helps in reducing errors due to pruning (see [1]); (4) post-processing (see section 2, approx. 1 min).

Progr.	Dur. (m:s)	WER (%)		Timing (m:s)	
		baseline	new	baseline	new
Docu V	49:50	9.10	7.76	5:45	5:49
Docu V+S		1.03	0.99	6:12	7:27
Docu I	51:32	32.28	24.45	49:51	32:38
Docu I+S		17.06	13.00	33:57	25:59
Soap	30:17	75.18	61.79	61:46	51:35

Table 1: ASR results for 1 documentary with voice-over only (Docu V) and 1 documentary with interviews (I) with (+ S) or without using the screenplay, and 1 episode of a daily soap.

6. Results

The system has been tested by VRT on several types of programmes. A few of these were transcribed manually to establish a ground truth: one (scripted) documentary with only voice-over (*Docu V*), one (scripted) documentary with interviews (*Docu I*), and one episode of a daily soap (*Soap*; although not a main use case for this project, it provides insight in the limitations of the current technology when dealing with very spontaneous, dialectal language). The results and timings for the baseline system [1] and the new system are summarised in Table 1. The timing is done on an Intel Core i5-2400 processor and includes all steps described in section 5. The system has a 2GB memory footprint, which is mainly determined by the LM.

The results show a relative improvement between 15% and 24% for the new system. Moreover, the fact that DNN scores are more discriminative helps in keeping the decoding time more in check when handling very difficult data such as *Soap* and *Docu I* which contain passages with dialect speech, loud background noise and/or music, concurrent speech and other compounding factors. For easy material with a script (*Docu V+S*) the overhead of the lexicon creation and post-processing (3 min) is no longer negligible. This will be solved in future updates of the system. The results on the soap taught us that the very challenging conditions encountered in such programs (dialect, loud background music/noise ...) have a detrimental impact on the accuracy of the current SLT tools. In *Docu V+S*, the remaining errors are due to errors in the G2P, text pre- and postprocessing (capitals, interpretation of quotes, indication of a long pause in the script, overzealous compounder), interpretation differences (script versus annotator), and colloquial language. The errors in *Docu I+S* are also due to unscripted parts, very noisy speech passages that were labelled as noise by the AAS, and spontaneous speech phenomena (broken-off words, repetitions, dialect speech ...).

A short demonstration of the system can be found here: `http://www.esat.kuleuven.be/psi/spraak/demo/STON`.

7. References

[1] K. Demuynck, A. Puurula, D. Van Compernolle, and P. Wambacq, "The ESAT 2008 system for N-Best Dutch speech recognition benchmark," in *ASRU*, 2009, pp. 339–343.

[2] B. Desplanques and J.-P. Martens, "Model-based speech/non-speech segmentation of a heterogeneous multilingual TV broadcast collection," in *ISPACS*, 2013, pp. 55–60.

[3] B. Desplanques, K. Demuynck, and J.-P. Martens, "Factor analysis for speaker segmentation and improved speaker diarization," in *Proc. Interspeech*, 2015, pp. 3081–3085.

[4] B. Desplanques, K. Demuynck, and J.-P. Martens, "Robust language recognition via adaptive language factor extraction," in *Proc. Interspeech*, 2014, pp. 2160–2164.

[5] K. Demuynck, J. Roelens, D. Van Compernolle, and P. Wambacq, "SPRAAK : an open source SPeech Recognition and Automatic Annotation Kit," in *Proc. Interspeech*, 2008, pp. 495–498.