Automated Production of True-cased Punctuated Subtitles for Weather and News Broadcasts

Joris Driesen[1], *Alexandra Birch*[1], *Simon Grimsey*[2],
Saeid Safarfashandi[2], *Juliet Gauthier*[2], *Matt Simpson*[2], *Steve Renals*[1]

[1]Centre for Speech Technology Research, University of Edinburgh, UK
[2]Red Bee Media, UK

{jdriesen,abmayne,srenals}@inf.ed.ac.uk,
{Saeid.Safarfashandi,Simon.Grimsey,Juliet.Gauthier,Matt.Simpson}@redbeemedia.com

Abstract

Providing subtitling for multimedia content is a highly costly process. Any system aimed at automating at least part of this process may therefore yield significant economic benefits for content providers. In this paper, we present an integrated automatic system capable of automatically subtitling weather forecasts and news broadcasts. In this system, a number of different modules are stringed together, each performing a single processing step in the pipeline. An ASR (Automatic Speech Recognition) module first converts raw audio into an uninterrupted stream of written words. A decision tree classifier then marks sentence boundaries in the resulting word sequence. Finally, a SMT (Statistical Machine Translation) module 'translates' the resulting sentences into punctuated true-cased text. The system has been developed in close cooperation with Red Bee Media and will be deployed in their commercial production pipeline.

Index Terms: subtitles, multimedia, automatic speech recognition, machine translation

1. Introduction

Multimedia content providers endeavour to subtitle their programmes, with the goal of making them accessible for viewers who suffer from a hearing impairment. Subtitles are usually made by human subtitlers by means of *re-speaking*. This involves listening to a televised broadcast and repeating the spoken content in a noise-free environment to an ASR system that is highly tuned to the speech characteristics of the subtitler. The few transcription errors made in this process are fixed manually. An efficient and accurate method like this is necessary, since multimedia companies are often legally obliged to provide a very high coverage under very stringent quality demands. For example Red Bee Media, a UK-based content provider formerly part of BBC[1], adheres to the *Client Service Level Agreement*, which demands readable punctuated subtitles for the majority of their produced content, with a Word Error Rate (WER) no higher than 2% for live subtitles (i.e. made on-the-fly) and 0% for prepared subtitles. Even though the process of subtitling has been optimized and streamlined, it is still a highly labour-intensive task which carries with it significant economic costs.

This work has been funded by the European Union as part of the Seventh Framework Programme, under grant agreement no. 287658 (EU-BRIDGE)

[1]British Broadcasting Corporation, the UK's national radio and television corporation

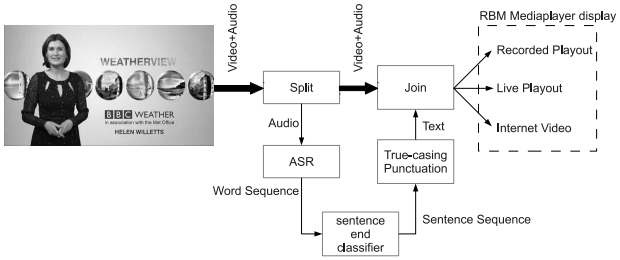

Figure 1: *A schematic overview of the proposed subtitling process.*

Furthermore, failure to meet the legal demands of coverage and accuracy is penalized monetarily, which may drive costs even higher. Because of these high costs and the sheer volume of produced media content, there are substantial commercial benefits to even modest improvements to the current subtitling process.

In this paper, we present a modular end-to-end system which takes a video as input and produces readable subtitles in a fully automated way. A schematic overview of this system is shown in figure 1. The video content that serves as input to this system comes from two different domains: 'Weatherview' and 'Sky News'. Weatherview is the daily weather forecast for the UK on BBC. From an ASR perspective, this data is the easier of the two to handle. It consists of 3-minute segments, uttered by a single speaker in isolation, employing a restricted vocabulary. For more details, see [1]. The other input type consists of broadcast news, taken from the news channel 'Sky News'. This data poses a bigger challenge, as it contains a multitude of different speakers, recording and noise conditions, a much larger vocabulary which tends to change over time as words and expressions rise and fade in popularity, etc. All inputs in this paper consists of data that was aired recently. As such, the proposed system is designed to cope with inputs of real-world scale and complexity.

2. Automatic Speech Recognition

In order to convert the input audio to text, we trained up two ASR systems, one for each of the input types of interest. For both ASR systems, 13 MFCC's along with their first and second order differences were calculated within 25 ms frames which were shifted in steps of 10 ms. The resulting 39-dimensional spectral representations were then modified with Cepstral Mean

Normalization (CMN). For Sky News data, where the speaker and recording conditions at any given time are unknown, we divided the input into segments of 40 seconds and assumed each of these segments contained a single unique speaker. Since most speakers in the data talk for longer than 40 seconds, this is a reasonable assumption to make for CMN. In a next step, the data is linearly transformed using MLLT and LDA [2]. A DNN-HMM hybrid ASR system for both systems was then trained by means of lightly supervised training, making use of approximate transcriptions, in the same way as in [1]. The DNN in both systems contained 6 layers with 2048 nodes each, and a softmax output layer with targets of approximately 3000 tied states. The GMM-HMM component of the hybryd setup contained 48000 Gaussians. For the Weatherview system, this GMM-HMM was trained using Speaker Adaptive Training (SAT) [3]. For the Sky News system, no SAT was applied, since no sufficiently reliable speaker information was available.

The language model (LM) in both systems consisted of a general background model biased towards each task's respective domain. For the training of this background model, approximately 700 million words of text were available. A selection of 30% from this data was made according to maximum cross-entropy with the target domain, and only this selected data was used for training. The resulting 4-gram background LM was then biased by linearly interpolating with a domain-specific 4-gram LM. The interpolation factor was optimized on the same set on which the domain-specific LM was trained, and therefore turns out heavily in favour of the latter.

3. Sentence End Classification

As discussed below in section 4, a SMT system can be effectively applied to punctuate and true-case a word stream produced by ASR. Due to computational limitations, however, one cannot do this for word sequences of arbitrary length. A solution would be to split the word stream at fixed intervals, e.g. every 100 words, and operate on the resulting segments. However, since SMT systems perform optimally on grammatically correct sentences, it would be preferrable to introduce breaks at correct locations in the word stream.

To this end, we trained a classifier to determine for each word in the ASR output whether it is the last word of a sentence or not. As inputs for this classifier we use the POS-tags in a context window of 7 words (i.e., the POS-tags associated with the 3 preceding words, the word itself, and the 3 following ones). To perform the classification, we used the C5.0 decision tree algorithm [4], which was trained on the 700 million words text corpus from section 2.

The accuracy of this classification could likely be improved by using more informative input, such as words or word classes. This, however, would significantly increase the costs in terms of speed and computational complexity. Evaluation of the proposed system on a held out set of Sky News transcriptions showed that 57.2% of predicted sentence ends corresponded to a full-stop. Manual verification showed that the vast majority of the remaining 42.8% corresponded to the location of other punctuation marks, e.g. commas, semicolons, exclamation marks, etc. In the end, the classifier splits the ASR output into segments of on average 58 words. Although these segments may contain more than a single sentence, each of them does constitute an adequate input to the SMT system.

4. Statistical Machine Translation for True-casing and Punctuation

SMT systems are designed to convert text from a source language into a target language. They achieve this by analyzing large parallel text corpora spanning both languages, and constructing a statistical mapping between the two. By training a SMT system on a text corpus, paired with a lower-cased depunctuated version of itself, the system can effectively learn to "translate" lower-cased word sequences into true-cased text with punctuation. The SMT system used in this paper is Moses [5]. This system regards text input as a sequence of "phrases", i.e. word blocks of varying length that often occur together as a unit. Since the order of these phrases is not necessarily fixed between languages, the search space for a translation increases drastically with the length of the input, as all reorderings must be considered. This is the main reason for splitting up the ASR output, as discussed in section 3. The reason for splitting at sentence boundaries is that partial sentences tend to introduce ambiguities. For example, in a sentence like ``i took a sip the coffee was hot'', the erroneous phrase ``sip the coffee'' will be more likely to be selected if the first and last few words of the sentence are omitted. It is clear that when this happens, the system will fail to insert the full-stop which separates the two sentences.

5. Integration in Red Bee Media Workflow

As was stated above, multimedia content providers are under constant pressure to reduce the cost of subtitling, and increase both its efficiency and accuracy. The proposed system was designed from the ground up to be integrated smoothly into the production pipeline at RBM. The subtitles it produces are easily transcoded into the required formats for content delivery, such as internet video, live playout, etc. Although the proposed system is very unlikely ever to replace human transcribers, it may simplify and speed up their work. Moreover, it may provide a valuable back-up system, ready to take over when the need arises. Lastly, by training up the SMT module with different language pairs, the system allows subtitling in different languages at almost negligible additional costs.

6. References

[1] J. Driesen and S. Renals, "Lightly supervised automatic subtitling of weather forecasts," in *Proc. Automatic Speech Recognition and Understanding Workshop*, Olomouc, Czech Republic, December 2013.

[2] M. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75–98, 1998.

[3] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, 1998.

[4] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[5] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation," in *Proc. ACL*, Prague, Czech Republic, 2007, pp. 177–180.