# A SYSTEM FOR AUTOMATIC ALIGNMENT OF BROADCAST MEDIA CAPTIONS USING WEIGHTED FINITE-STATE TRANSDUCERS

*Peter Bell and Steve Renals*

Centre for Speech Technology Research, University of Edinburgh, Edinburgh EH8 9AB, UK

{peter.bell, s.renals}@ed.ac.uk

## ABSTRACT

We describe our system for alignment of broadcast media captions in the 2015 MGB Challenge. A precise time alignment of previously-generated subtitles to media data is important in the process of caption generation by broadcasters. However, this task is challenging due to the highly diverse, often noisy content of the audio, and because the subtitles are frequently not a verbatim representation of the actual words spoken. Our system employs a two-pass approach with appropriately constrained weighted finite state transducers (WFSTs) to enable good alignment even when the audio quality would be challenging for conventional ASR. The system achieves an f-score of 0.8965 on the MGB Challenge development set.

***Index Terms—*** MGB Challenge, broadcast media, alignment

## 1. INTRODUCTION

Automatic alignment of text to audio is an important and well-studied problem. Forced alignment of words to speech data, usually at the utterance level, is of course a standard step in acoustic model training for Automatic Speech Recognition (ASR) systems, but this approach has been extended to cases where the available text may not be well-matched to the accompanying audio, or where timings are available only at a very coarse level – the so-called *lightly-supervised* approach to acoustic model training [1]. The use of such of techniques has subsequently been widely reported for both ASR [2, 3, 4] and TTS [5, 6] applications, in cases where large quantities of in-domain text and audio resources are available, but where it is not feasible to generate careful manual transcription of the resources. However, the ability to produce good alignments of audio and text resources, in the absence of utterance-level timing and when the text is not a verbatim transcription, also has immediate benefits in end-user applications such as lecture transcription [7, 8], broadcast media captioning [9], parliamentary proceedings reporting [10], online video indexing [11] and e-book readers [12], to give some examples. In contrast with use for lightly-supervised training, in these applications, alignment systems typically must return align words to *all* the speech, rather than simply being able to select only regions of confident alignment.

Despite the range of work on lightly supervised alignment reported in the literature, as far as we are aware, to date there has been no means of evaluating competing alignment systems on common data. This motivated us to include a lightly-supervised alignment task in the *MGB Challenge* [13], an official challenge at ASRU 2015. The Challenge is an evaluation focused on multi-genre TV broadcasts, where there is a particular need for automatic caption alignment so that subtitles can be displayed to TV viewers on screen at the correct time. For the evaluation, the captions supplied as input for the alignment task are human-generated captions as displayed on TV when the material was originally broadcast, with the timings removed. Alignment must be performed at the show-level; shows are up to around one hour in duration.

From an acoustic modelling perspective, the data is extremely challenging, being highly diverse across genres with respect to background noise levels, accents and speaking styles (which include fast, natural conversations and dramatic, exaggerated speech). Although performance on genres such as news and documentaries is relatively good, genres such as drama and sport are much more difficult for ASR. As we report in [13], a baseline ML-trained HMM-GMM ASR system trained on 260 hours of in-domain audio data achieves a Word Error Rate (WER) of over 50%. These difficulties apply to lightly-supervised alignment also, necessitating careful system design. Additionally, the captions supplied frequently differ from verbatim transcriptions for a range of reasons, including the need to paraphrase or re-order to aid viewer understanding, edits to reduce the amount of text displayed when speech would be too fast for viewers to read, and the result of errors in the original caption generation.

Because not all words in the caption text are actually present in the audio, participating systems are not required to supply timings for every word. Alignment scoring is therefore composed of scores for precision (the proportion of supplied word timings that are correct) and recall (the proportion of words actually spoken that have correct timings supplied).

This research was supported by EPSRC Programme Grant grant EP/I031022/1, *Natural Speech Technology*. Supplementary data relating to the MGB Challenge may be accessed from the NST research data collection at http://datashare.is.ed.ac.uk/handle/10283/786.

Timing is judged correct with reference to a 100ms window, which is somewhat stricter than equivalent measures reported in the literature. Precision and recall are combined to produce the F-score, the final reported measure.

The system we propose for this task is designed to be efficient to run, so that it can be used in practical applications, whilst ensuring that the alignment is robust to noisy or otherwise difficult audio, and also to instances where the text differs substantially from the speech. With these objectives in mind, we use a two-pass system based using purely acyclic WFST based decoding constraints.

## 2. BACKGROUND AND PREVIOUS WORK

A diverse range of techniques have been used for long audio alignment. Typically, methods require previously-trained ASR acoustic models in combination with a decoder. [11] proposed an iterative approach whereby successive ASR passes over the data aim to identify an increasing number of reliable "islands of confidence" where the ASR output matches the text. Successive iterations aim to refine the alignment between anchors. The publicly-available SailAlign toolkit [14] uses a similar method. Many approaches, for example [7, 5, 14] start by training a biased language model (LM) on the alignment text, possibly interpolated with a background LM, to be used in initial, and possibly subsequent, ASR passes. The resulting transcript is aligned to the reference text with dynamic programming. Methods for alignment in low-resource scenarios may train acoustic models from the alignment audio [4, 6]; in multi-pass systems it is possible to adapt acoustic models to the already aligned audio.

An alternative approach is to apply weaker constraints on the acoustic model decoding: [8, 10] use a phonetic decoding, which is matched to the original text by efficient dynamic programming of phoneme sequences (or grapheme sequences in the latter case). [12] avoids the need for ASR models completely by performing dynamic time warping to align with TTS output generated from the text. In the other direction, it is possible to apply much stronger constraints on the decoding: [2] used a *factor automaton*, which matches all possible sub-strings of the original text, to constrain the decoder to produce, for each utterance, only contiguous strings of words from the training text. This was found to significantly improve decoding efficiency and accuracy compared to an equivalent n-gram LM. We found this approach to be promising in previous experiments [6] and so used it is a starting point the system described here.

## 3. ALIGNMENT SYSTEM

### 3.1. Alignment with WFSTs

In common with many other ASR systems, we use a WFST-based decoder [15]. Transducers for the grammar, $G$, lexicon, $L$, context-dependency, $C$, and HMM-topology, $H$, are composed to form a single transducer $H \circ C \circ L \circ G$ with appropriate determinisation and minimisation steps. The decoding task is equivalent to finding the shortest path through the transducer. In ASR applications, $G$ is usually derived from an n-gram LM; however, it may trivially be replaced by a transducer specific to the task.

Since we use hybrid DNN acoustic models (described below), where the forward pass is computationally expensive and difficult to cache, we aim to minimise the number of passes over the data in order to limit the computation required. The final system operates with just two passes. In the first pass, we adopt the method of [2] discussed briefly above. Here, the complete text to align is converted into a linear factor transducer – one per audio file – where each word is both a potential entry point and final exit state. This transducer matches all substrings of the original text. An example is shown in Figure 1. We use a decoder provided by the Kaldi toolkit [16]. For robustness, we allow inter-word insertions – words spoken but not included in the text. The implementation is somewhat different to [2], where insertions are included as self-loops in the grammar, in that we implicitly allow inter-word insertions through the use of a "short-pause" (or "tee") model implemented as part of the lexicon The inclusion of the short-pause model in the lexicon is a standard approach in Kaldi; "#0" is used as a word-disambiguation symbol that is matched purely to this model. In our system, it models both silence and speech-shaped noise, and is therefore able to absorb arbitrary spoken words not included in the text with small cost.

The factor transducer approach has several advantages for the MGB alignment task: the decoding has strong text-based constraints, but only weak timing constraints – in the sense that outputs for successive utterances are not constrained to be in order, making it suitable for aligning long portions of audio. Because of the linearity of the transducer, decoding is significantly more efficient than using a 3-gram model trained on the same text. The constraint that each complete utterance must match substrings of the original text makes the decoding highly robust to portions of the audio where the acoustic models are poorly matched, due to high noise levels for example, where decoding with a 3-gram could yield very high error rate. Thus we can maximise alignment accuracy from just a single pass through the data without need to align the decoder output to the original text.

However, there are are number of limitations to the factor transducer approach. Firstly, as noted by [2], the method performs poorly on very short audio segments, where the constraints imposed by the linear $G$ transducer are weak. Secondly, although the factor transducer is robust to word insertions, deletions are much more problematic. Small deletions – words included in the text but not actually spoken – can be handled by the decoder simply by aligning them to very short portions of audio between words. However, this means

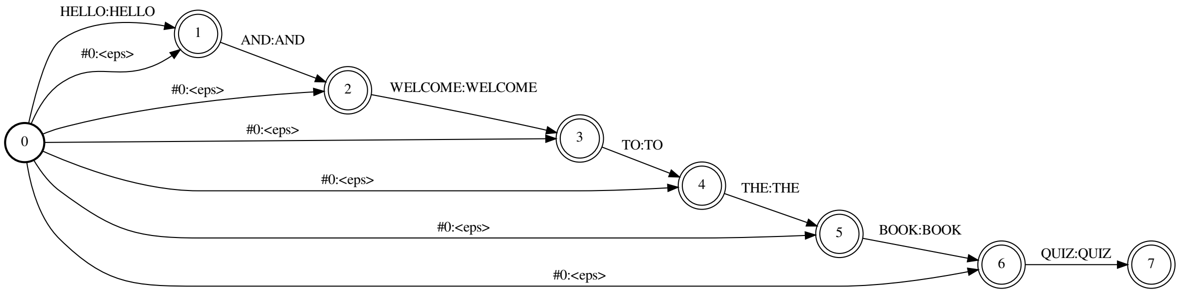


**Fig. 1**. Factor transducer for a short example text

that they cannot be detected as missing. Furthermore, it is common in this task that there are longer deleted portions of text caused, for example, by re-ordering within a dialogue to increase viewer comprehension. In this situation, the factor transducer method may be unable to align the correct words on either side of the deleted region, causing alignment to fail for the whole utterance.

When alignment is used for training purposes, these issues may not be particularly important, since these segments may simply be discarded; however, this is not possible when alignment is the primary goal. We address the problems with a second-pass alignment. Following alignment of the first-pass output to the original text we resegment the data, extending and joining segments where there were missing words in the first pass. Given an utterance, we use the text alignment to identify the portion of text that could potentially be aligned to it using a broad time window. We now create a factor transducer on the fly for the utterance in question. The new transducer includes optional word skips providing robustness to deletions, which are removed from the decoder output. The resulting transducer is illustrated in Figure 2. The word skips are assigned a fixed prior probability, $p$, tuned on the development data, which acts to penalise excessive word removal. A similar design was used by [7], though with a standard linear transducer. The transducer creation and decoding were implemented by means of extending a number of Kaldi tools.

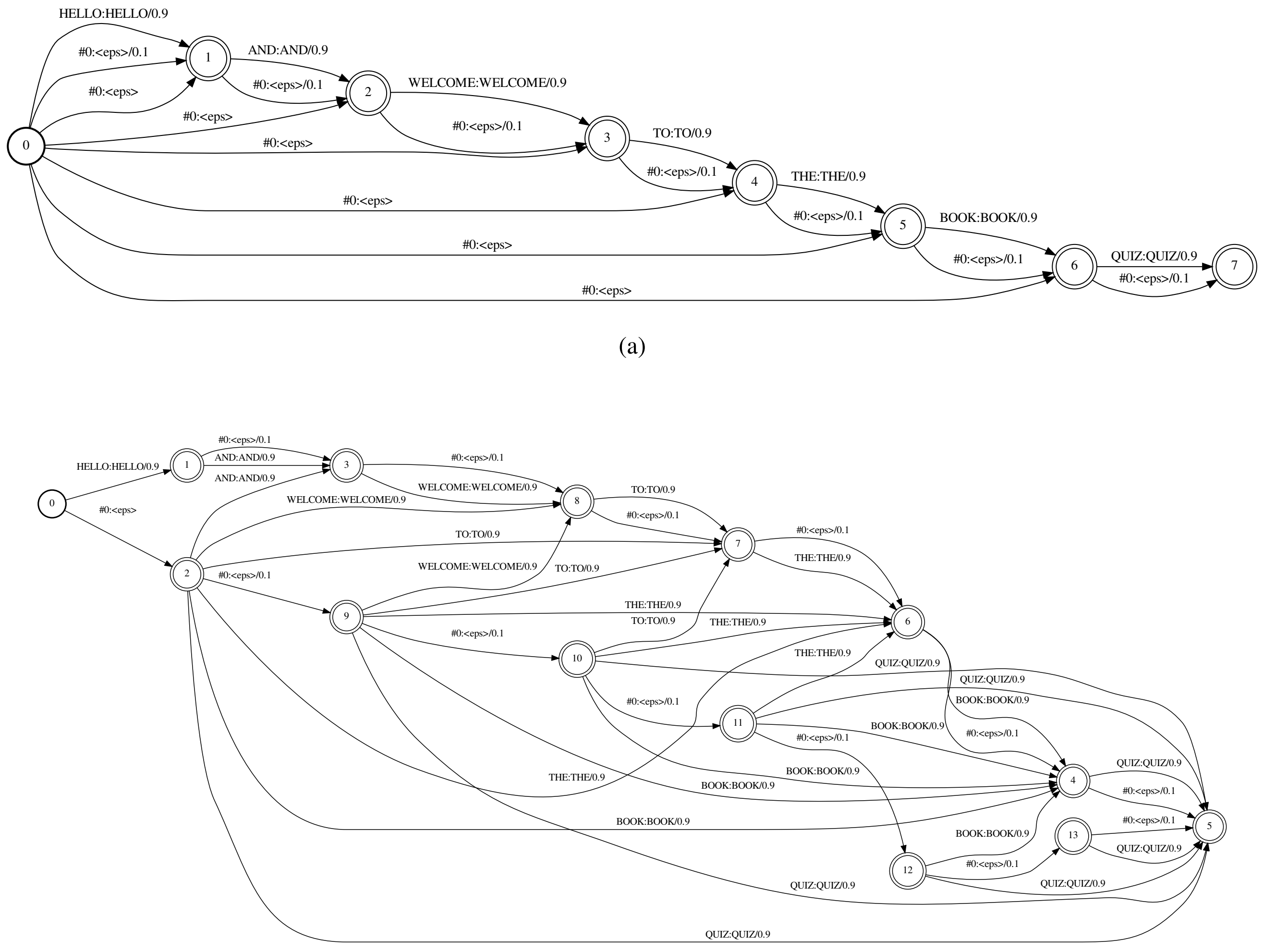
As illustrated in Figure 2, the determinised version of this new transducer is significantly larger than the original; proportionally it is even larger after composition with the $L$ and $C$ transducers. Although the size is of course limited by the use of a short portion of text, we find that it is necessary to use the skip penalty in conjunction with suitably-chosen pruning factors for efficient decoding with this transducer. At this stage, decoding is considerably less efficient that using an equivalent n-gram transducer – however, the strong advantage is that the second-pass output is guaranteed to consist of complete strings from the original text, subject only to possible deletions. We use this as our final output, without further text-to-text alignment.

### 3.2. Acoustic models

Our final acoustic models were trained on 640 hours of in-domain speech data from the MGB Challenge training set of broadcast multi-genre TV [13]. This data is itself obtained using a lightly-supervised alignment, which we did not update during system-building. We selected data with a ceiling of word-level Matched Error Rate (MER) of 40%. GMM acoustic models were trained on MFCC features using a standard Kaldi recipe that we made available to all challenge participants. We trained DNNs with 6 hidden layers and 2048 units per layer to generate posterior probabilities over 28k tied states derived from the GMM. The input features were projected with MLLT, and a window of 9 frames was used. We did not use speaker adaptation. One feature of training on the data is that, because no ground-truth speech/non-speech segmentation is available, and because the training is lightly-supervised, it is inevitable that short-pause models will be trained on significant quantities of spoken material. It therefore has ability to align to arbitrary speech, which as discussed in Section 3.1 above, happens to be useful for purpose of alignment.

We found that, particularly in view of the large number of tied states and the difficult nature of the acoustic data, it was important for good ASR performance to generate a new alignment of the training data with the fully-trained DNN and train a new DNN completely from scratch on these new alignments. The final DNNs additionally had two iterations of sequence training applied [17].

We investigated these models for use in standard ASR, on the transcription task of the MGB Challenge using a 4-gram model, again trained according to the standard recipe. They scored 30.5% WER on the development set and 32.0% on the evaluation set, both with the supplied baseline segmentation.



(a)

(b)

**Fig. 2**. Factor transducers with optional word deletions: (a) original version, (b) determinised version

It should be noted that this system is not particularly competitive on the ASR task compared to other entries to the Challenge. We also trained preliminary acoustic models following a quicker DNN recipe, on just 260 hours of speech (selected according to an MER ceiling of 10%), which were used in early system development experiments.

### 3.3. Other system features

To obtain pronunciations for words in the text, we use the Combilex lexicon of British English[1] [18, 19], which was made available to all MGB Challenge participants. For words not in the lexicon we generated pronunciations automatically with the Sequitur tool [20]. For the text-to-text alignment required after the first pass decode, we use NIST's SCLITE tool[2].

## 4. RESULTS

We first present development results on an internal test set of media data, for which verbatim manual transcriptions were not available. This was used during early stages of algorithm development, to optimise pruning factors. As a proxy for measuring system performance, we made the assumption that

[1] http://www.cstr.ed.ac.uk/research/projects/combilex/

[2] ftp://jaguar.ncsl.nist.gov/pub/

| System | MER | Del | RTF |
|---|---|---|---|
| Pass 1 FT (GMM) | 27.6 | 16.5 | 0.186 |
| Pass 2 n-gram (GMM) | 57.8 | 15.8 | 1.844 |
| Pass 1 FT (DNN) | 20.1 | 13.6 | 0.720 |
| Pass 2 FT+del (DNN) | 6.9 | 5.4 | 1.488 |

**Table 1**. Development results on an internal test set where verbatim transcriptions were not available. MER and the deletion rate ("Del") are calculated with reference to the supplied captions text.

| System | Precision | Recall | F-score |
|---|---|---|---|
| Preliminary DNN AMs | | | |
| Pass 1 FT | 0.8816 | 0.7629 | 0.8180 |
| + force align | 0.8290 | 0.7855 | 0.8066 |
| Pass 2 FT+del | 0.8679 | 0.8563 | 0.8620 |
| Final DNN AMs | | | |
| Pass 1 | 0.9009 | 0.8128 | 0.8546 |
| Pass 2 FT+del | *0.8856* | *0.9013* | *0.8934* |
| Pass 2(b) FT | 0.8896 | 0.8946 | 0.8921 |
| Pass 2(b) FT+del | 0.8928 | 0.9002 | **0.8965** |

**Table 2**. Results on the MGB Challenge development set.

the supplied text for alignment is approximately correct. Here we show error rates with reference to this text. Results are summarised in Table 1. To illustrate the benefits of the factor transducer (FT) approach, we first show results using our initial GMM system trained using the baseline Kaldi recipe. We compare this with a decode using a biased 3-gram trained on the same transcription text. It can be observed that the MER and real-time factor (RTF) for the decoding are substantially lower for the factor transducer method. Moving to the preliminary DNN system, the speed advantage is lessened, since we are forced to output posteriors for all tied states in the DNN forward-pass, which is a significant portion of the computation: this reduces the benefits of decoder optimisation somewhat. The MER is however, substantially reduced. The second-pass decode with deletions allowed (FT+del) has a significantly higher real-time factor, but reduces MER much further. The system can, of course, easily be parallelised within each pass.

| Skip prob, $p$ | Precision | Recall | F-score |
|---|---|---|---|
| 0.0001 | 0.8665 | 0.8573 | 0.8619 |
| 0.001 | 0.8679 | 0.8563 | **0.8620** |
| 0.01 | 0.8700 | 0.8521 | 0.8610 |
| 0.05 | 0.8729 | 0.8463 | 0.8594 |
| 0.1 | 0.8732 | 0.8399 | 0.8562 |
| 0.2 | 0.8742 | 0.8292 | 0.8511 |

**Table 3**. Pass 2 results on the development set with varying skip probability, using preliminary DNN AMs

In Table 2 we present results on the full MGB Challenge development set using the official scoring package. The first section of table shows results with the smaller, preliminary acoustic models. It is interesting that applying a forced-alignment (with the same models) to the output of the first pass actually makes the F-score slightly worse. This appears to be due to the reduction in precision caused by forcing times to be output for every word in the text. However, the F-score improves substantially after the second pass decoding with the factor transducer, with deletions allowed. The second pass decoding uses a skip probability $p = 0.001$, obtained through tuning on the development set. Table 3 gives results with other values of $p$, showing that the F-score is relatively insensitive to the choice.

The contrast between the first and second sections of Table 2 illustrates the benefit of using the better acoustic models. There is substantial gain here, implying that we could improve results of the proposed alignment algorithm still further by using more competitive models. The italicised row of the table shows the results of our primary entry to the Challenge. The system achieved an official F-score of **0.8773** on the evaluation set, placing it second out of six entries to this task.

In the final section of the table we show some results obtained after the end of the Challenge evaluation period, following improvements to the text alignment used to generate the utterance-specific text selection for the second pass. We denote this by Pass 2(b). The improvements were designed mostly to deal with particularly noisy audio where we observed that a very poor first-pass can lead to erroneous text alignments input to the second pass. These improved scores are not reflected in our official results.

## 5. CONCLUSIONS

We have demonstrated an efficient algorithm for caption alignment that is reasonably robust to challenging speech data and when the captions are not necessarily well-matched to the text. On the MGB Challenge task, it was found to perform well when compared to other systems that attained lower word error rates on the standard transcription task, suggesting that the transducer-based constraints we used are useful for this application.

To make this system more suitable for deployment at scale, we will in future investigate the use of context-independent DNNs. In the second-pass decoding, we also plan to use dynamic composition of the $G$ transducer with the $H \circ C \circ L$ portion, in order to minimise the computation and memory requirements of the transducer construction in this step.

## 6. REFERENCES

[1] L. Lamel, J.-L. Gauvain, and A. Adda, "Lightly supervised and unsupervised acoustic model training," *Computer Speech and Language*, vol. 16, pp. 115–129, 2002.

[2] P.J. Moreno and C. Alberti, "A factor automaton approach for the forced alignment of long speech recordings," in *Proc. ICASSP*, 2009.

[3] M. Paulik and P. Panchapagesa, "Leveraging large amounts of loosely transcribed corporate videos for acoustic model training," in *Proc. ASRU*, 2011.

[4] M. Alessandrini, G. Biagetti, A. Curzi, and C. Turchetti, "Semi-automatic acoustic model generation from large unsynchronized audio and text chunks," in *Proc. Interspeech*, 2011.

[5] N. Braunschweiler, M.J.F. Gales, and S. Buchholz, "Lightly supervised recognition for automatic alignment of large coherent speech recordings," in *Proc. Interspeech*, 2010.

[6] A. Stan, P. Bell, and S. King, "A grapheme-based method for automatic alignment of speech and text data," in *Proc. SLT*, 2012.

[7] T.J. Hazen, "Automatic alignment and error correction of human generated transcripts for long speech recordings," in *Proc. Interspeech*, 2006.

[8] A. Haubold and J.R. Kender, "Alignment of speech to highly imperfect text transcriptions," in *Proc. ICME*, 2007.

[9] A. Dubinsky, "Syncwords: A platform for semi-automated closed captioning and subtitles," in *Proc. Interspeech*, 2014, show & tell.

[10] G. Bordel, M. Peñagarikano, L. Rodríguez-Fuentes, and A. Varona, "A simple and efficient method to align very long speech signals to acoustically imperfect transcriptions," in *Proc. Interspeech*, 2012.

[11] P. Moreno, C. Joerg, J.-M. Van Thong, and O. Glickman, "A recursive algorithm for the forced alignment of very long audio segments," in *Proc. ICSLP*, 1998.

[12] A. Anguera, N Perez, A. Urruela, and N. Oliver, "Automatic synchronization of electronic and audio books via TTS alignment and silence filtering," in *Proc. ICME*, 2011.

[13] P. Bell, M.J.F. Gales, J. Kilgour, P. Lanchantic, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester, and P.C. Woodland, "The MGB challenge: Evaluating multi-genre broadcast media transcription," 2015.

[14] A. Katsamanis, M. Black, P. Georgiou, L. Goldstein, and S. Narayanan, "Sailalign: Robust long speech-text alignment," in *Proc. Workshop on New Tools and Methods for Very-Large Scale Phonetics Research*, jan 2011.

[15] M. Mohri, F. Pereira, and M. Riley, *Handbook on Speech Processing and Speech Communication,*, chapter Speech recognition with weighted finite-state transducers, Springer-Verlag, Heidelberg, Germany, 2008.

[16] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *Proc. ASRU*, Dec. 2011.

[17] K. Vesely, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. Interspeech*, 2013.

[18] K. Richmond, R. Clark, and S. Fitt, "Robust LTS rules with the Combilex speech technology lexicon," in *Proc. Interspeech*, 2009.

[19] K. Richmond, R. Clark, and S. Fitt, "On generating Combilex pronunciations via morphological analysis," in *Proc. Interspeech*, 2010.

[20] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, 2008.