PHONETIC AND GRAPHEMIC SYSTEMS FOR MULTI-GENRE BROADCAST TRANSCRIPTION

Y. Wang, X. Chen, M. J. F. Gales, A. Ragni and J. H. M. Wong

Cambridge University Engineering Dept, Trumpington St., Cambridge CB2 1PZ, U.K.
Email: {yw396, xc257, mjfg, ar527, jhmw2}@eng.cam.ac.uk

ABSTRACT

State-of-the-art English automatic speech recognition systems typically use phonetic rather than graphemic lexicons. Graphemic systems are known to perform less well for English as the mapping from the written form to the spoken form is complicated. However, in recent years the representational power of deep-learning based acoustic models has improved, raising interest in graphemic acoustic models for English, due to the simplicity of generating the lexicon. In this paper, phonetic and graphemic models are compared for an English Multi-Genre Broadcast transcription task. A range of acoustic models based on lattice-free MMI training are constructed using phonetic and graphemic lexicons. For this task, it is found that having a long-span temporal history reduces the difference in performance between the two forms of models. In addition, system combination is examined, using parameter smoothing and hypothesis combination. As the combination approaches become more complicated the difference between the phonetic and graphemic systems further decreases. Finally, for all configurations examined the combination of phonetic and graphemic systems yields consistent gains.

***Index Terms*—** Speech recognition, graphemic lexicon, lattice-free MMI, model combination

1. INTRODUCTION

Hidden Markov model (HMM) based automatic speech recognition (ASR) systems are typically built using sub-words units, such as phones or graphemes. System performance depends on an appropriate definition of sub-word units and the accuracy, and consistency, of decomposing words into these sub-word units. Phonetic lexicons provide a mapping between the orthographic representation of a word, a sequence of letters (graphemes), into a sequence of phones. However, generation of these lexicons requires linguistic knowledge of the target language, which is time-consuming and expensive. On the other hand, graphemic lexicons are attractive as the graphemes are directly used. Moreover, graphemic lexicons can be easily expanded to include out-of-vocabulary (OOV) words, unlike phonetic lexicons. For languages with a close grapheme-to-phone mapping, graphemic HMM-based systems have been shown to perform similarly to phonetic systems [1, 2, 3]. However, for languages with irregular grapheme-to-phone mappings, such as English, graphemic HMM-based systems normally perform significantly worse than their phonetic counterparts [2]. This is not surprising as the system relies on the acoustic model to implicitly capture the irregularities of the graphemic to acoustic realisation. When more powerful deep learning based acoustic models are used, such as connectionist temporal classification (CTC) [4] which model long-span temporal information, the gap between graphemic systems and phonetic systems is small on a read English task [5].

This paper aims to find out whether recent deep-learning based acoustic models, which also model long-span temporal information, allow HMM-based graphemic systems to perform at the same level of accuracy as phonetic systems for English. A range of models are available including long short-term memory (LSTM) networks [6], convolutional neural networks [7], time-delay neural networks (TDNN) [8] and bidirectional LSTM networks [9]. Additionally various layer-wise combination schemes allow the advantages of several models to be leveraged [10, 11]. These models also offer flexibility in terms of the span of the temporal information that they can capture. For instance, the interleaved TDNN-LSTM model [11] extends the temporal span of the LSTM model with a wide window into the future. These models can also be efficiently trained directly from random initialisation by using approaches such as lattice-free maximum mutual information (LF-MMI) estimation. This often results in improved performance over state-level minimum Bayes' risk (sMBR) trained models [12]. These complex models are likely to have, possibly significant, variations in ASR performance depending on the choice of training hyper-parameters. This variation in system performance can be taken advantage of using system combination [13]. This paper will examine two forms of system combination with different complexities and costs. The first is a random ensemble method [14], which utilises multiple training runs with different random seeds to produce slightly different yet complimentary systems. The second is model smoothing [15], which interpolates a number of intermediate model parameters using weights estimated on a subset of the training data. Finally, graphemic systems, if competitive with the phonetic system, should be complimentary to phonetic systems.

The rest of this paper is organized as follows. Sections 2 and 3 describe graphemic acoustic models and model combination approaches respectively. Section 4 details the experiments conducted on an English multi-genre broadcast transcription task with the phonetic and graphemic models as well as using different combination approaches. Finally, conclusions are given in Section 5.

2. GRAPHEMIC ENGLISH SYSTEMS

2.1. Graphemic lexicon

At the core of any graphemic system is the graphemic lexicon. For English, it is straightforward to form this from the 26 alphabet letters /a-z/. In addition to these base graphemes, it may also be useful to mark additional attributes such as apostrophes (DA) and abbreviations (DB). Excerpts from phonetic and graphemic lexicons are:

This research was partly funded under the ALTA Institute, University of Cambridge. Thanks to Cambridge English, University of Cambridge, for supporting this research.

Phonetic Lexicon

B.B.C.'s	/b/ /iy/ /b/ /iy/ /s/ /iy/ /z/
information	/ih /n/ /f/ /ax/ /m/ /ey/ /sh/ /en/
moon	/m/ /uw/ /n/
the	/dh/ /ax/

Graphemic Lexicon

B.B.C.'s	b;DB b;DB c;DADB s
information	i n f o r m a t i o n
moon	m o o n
the	t h e

From the first entry, the use of abbreviation and apostrophe attributes potentially allows the graphemic system to handle the discrepancy between the pronounced and written forms. The other three examples illustrate situations where graphemic systems may struggle to model letter omission ('r' in information), vowel ('oo' in moon), consonant ('th' in the) and vowel-consonant ('tio' in information) recombination. Though some of these issues can be handled using context dependent models, e.g. bi-graphemes and tri-graphemes, for others the length of the context necessary for disambiguation will be prohibitively large. For example, the phonetic lexicon used in Section 4 associates three phones, /dh/, /th/ and /t/, with the sound corresponding to the grapheme sequence 'th'. This problem is further compounded by the fact that the following grapheme 'e' depending on its neighbor is represented by 9 different vowel/consonant phones. The examples given in this section suggest that for graphemic systems, context modelling may be even more important than it is for phonetic systems [16].

2.2. Acoustic Model Structure

Rather than solely relying on acoustic modelling units to handle the intricate grapheme-to-phone rules, it is also possible to examine acoustic models capable of modelling long-span temporal information. In a deep neural network (DNN) acoustic model [17], only a small number of preceding and succeeding frames are typically used to predict the current state, s_t, as is shown in (1)

$$\text{(T)DNN: } P\left(s_t|\mathcal{O}_{1:T}\right) \approx P\left(s_t|\mathbf{o}_{t-\tau^{(l)}}, \ldots, \mathbf{o}_{t+\tau^{(r)}}\right) \quad (1)$$

$$\text{LSTM: } P\left(s_t|\mathcal{O}_{1:T}\right) \approx P\left(s_t|\mathbf{o}_1, \ldots, \mathbf{o}_{t+\tau^{(r)}}\right) \quad (2)$$

where the left $\tau^{(l)}$ and right $\tau^{(r)}$ context window sizes are typically less than 10. A TDNN [8] has a more complex structure that enables it to cover a significantly larger number of preceding and succeeding frames without significantly increasing the number of model parameters. For example, the model considered in Section 4 uses $\tau^{(l)} = 15$ past and $\tau^{(r)} = 10$ future frames. The use of recurrent units in a LSTM network, described in equation (2), allows even longer-span temporal information to be modelled. Note that in practice the past information is typically truncated after some fixed, yet large, number of frames (40 in this work). Furthermore, the TDNN-LSTM model [11] obtained by interleaving TDNN layers [8] with LSTM layers [6] increases the context window to 50 frames into the past and 20 frames into the future. In addition to being more powerful classifiers, these advanced deep-learning based acoustic models thus can utilise a significantly longer span of temporal information than that used in previous work with Gaussian mixture models.

3. MODEL COMBINATION

Training the deep neural network models discussed in Section 2 is a complicated process involving highly non-convex optimisation. There may thus be large variations between the behaviours of intermediate models from iteration to iteration, or between final models when originating from different starting points. The latter is likely to be larger when models are trained from different random initialisations using the LF-MMI criterion, as there is no cross-entropy initialisation stage with common targets for all systems. Such variation typically results in the models making different predictions. Depending on the level of useful variation, such diverse predictions may help to resolve confusions. This serves as the basis for various system combination approaches [18, 19, 20, 21].

3.1. Ensemble Methods

A combination of an ensemble of diverse and yet individually accurate systems can often result in significant gains [22]. Common methods to introduce diversity include random parameter initialisation for ASR [13, 14], bagging [23] and random decision trees [24]. Using different random initialisations has been shown to be a simple and efficient approach of introducing diversity [13, 14]. In [14], this method was able to provide significant diversity while keeping a similar performance across the systems. Thus, combining the systems in the ensemble could yield strong gains. A less common method of ensemble generation is to take a number of intermediate models during training and interpolate their parameters

$$\mathbf{\Phi} = \sum_{m=1}^{M} \alpha_m \mathbf{\Phi}_m \quad (3)$$

where M is the number of models, $\mathbf{\Phi}_m$ represents the parameters of the mth model and α_m represents its combination weight. This is the idea behind *model smoothing* [15], designed to reduce unwanted variations during the training. The models are normally selected from the later stages of training using a fixed iteration interval between the selected models (6 in this work). The combination weights are associated with the individual layers and optimised on a subset of training examples. The combination weights are constrained to sum to 1. Though generally it is hard to ensure that the combined model would improve over the final trained model, this paper shows that large performance improvements are possible.

To measure the diversity of the generated systems, it is possible to use cross word error rate (cWER) [13]

$$\text{cWER} = \frac{1}{M\left(M-1\right)} \sum_{m=1}^{M} \sum_{n \neq m} \frac{1}{\sum_{r=1}^{R} |\mathcal{W}_r^n|} \sum_{r=1}^{R} L\left(\mathcal{W}_r^m, \mathcal{W}_r^n\right), \quad (4)$$

where $\mathcal{W}_r^m$ represents the 1-best hypothesis of the rth utterance, using the mth model, and R is the total number of utterances. The cWER measures how different the 1-best hypotheses are between models and was found to be more correlated with the combination gains than the standard deviation of WERs [13].

3.2. Minimum Bayes Risk Combination

It is only possible to use model smoothing in equation (3) for combining iterations of the same model training run. A more general system combination approach is hypothesis-level combination. Examples of this form of approach are: ROVER [18]; confusion network combination (CNC) [19]; and minimum Bayes risk (MBR) combination [20]. In this work MBR combination is used, which finds the word sequence that attempts to minimise the expected WER across the systems being combined [20]:

$$\widehat{\mathcal{W}} = \underset{\mathcal{W}}{\operatorname{argmin}} \left\{ \sum_{m=1}^{M} \lambda_m \sum_{\mathcal{W}' \in \mathcal{H}} P\left(\mathcal{W}'|\mathcal{O}_{1:T};\mathbf{\Phi}_m\right) L\left(\mathcal{W},\mathcal{W}'\right) \right\}, \tag{5}$$

where λ_m are the combination weights, $P\left(\mathcal{W}|\mathcal{O}_{1:T};\mathbf{\Phi}_m\right)$ is the posterior probability of the word sequence, $\mathcal{W}$, given the observation sequence, $\mathcal{O}_{1:T}$, and the acoustic model, $\mathbf{\Phi}_m$, $\mathcal{H}$ is a set of hypotheses and $L\left(\mathcal{W},\mathcal{W}'\right)$ represents the Levenshtein distance between two word sequences $\mathcal{W}$ and $\mathcal{W}'$. Though more computationally expensive, MBR combination has been shown to perform better than the ROVER combination and CNC.

4. EXPERIMENTS

Experiments were conducted using the data from the 2017 English Multi-Genre Broadcast (MGB-3) challenge. The data was supplied by British Broadcasting Corporation (BBC) and consists of audio from BBC television programmes. The data contains a wide range of genres such as comedy, drama and sports shows. A total of 375 hours of audio data with associated subtitles is available for acoustic model training. Lightly supervised decoding and selection was used to extract 275 hours for training [25, 26]. A 6 hours development set, dev17b, was also supplied. The acoustic model features were 40-dimensional Mel-filter bank features normalised using utterance level mean normalisation and show-segment level variance normalisation [26]. Decision trees for both phonetic and graphemic systems were constructed in the same way as the Switchboard recipe in Kaldi. Around 3600 left bi-phone dependent states[1] were used as targets. The results are based on automatic audio segmentation using a DNN based segmenter [27] trained on the MGB-3 data.

To examine the impact of the acoustic model complexity on phonetic and graphemic system performance, a range of acoustic models of different topology and spans of temporal information were built. These include feed-forward DNN, sub-sampled TDNN, unidirectional LSTM and interleaved TDNN-LSTM models. The DNN models had 7 hidden layers of 600-dimensional sigmoid units and an input context window spanning from 10 frames into the past to 10 frames into the future. The TDNN models had 7 layers of 600-dimensional rectified linear units (ReLU) and wider input context window spanning from 15 frames into the past to 10 frames into the future.[2] The LSTM model had 3 LSTMP layers, each with 512-dimensional cells and 128-dimensional recurrent and non-recurrent projections. The effective temporal information window for the LSTM spans from 40 frames into the past to 7 frames into the future. The interleaved TDNN-LSTM models had 9 layers of 600 dimensional ReLU units.[3] The TDNN-LSTM model has the widest temporal information window, starting from 50 frames into the past and ending at 20 frames into the future. All models were trained using the LF-MMI criterion on a single GPU [28] using Kaldi toolkit [15]. For this work, only speaker-independent systems were used.

For the first pass decoding language model, a 3-gram language model with a 64K words lexicon was used. This was trained on the audio subtitles and 650M words of supplied BBC subtitles. In addition, a recurrent neural network language model (RNNLM) [29] was also used to refine the result of the first pass decoding. The CUED-RNNLM Toolkit v1.0 [30] was used to train the RNNLM using 1 layer of 1024-dimensional GRU units. Given the vocabulary size (64K) and quantity of training data (e.g. 650M words), noise contrastive estimation (NCE) was adopted to speed up training and evaluation [31]. At test time, a 4-gram approximation [32] of the RNNLM was used to rescore 4-gram lattices. As the RNNLM was trained with the NCE, the unnormalized output layer probabilities were used in rescoring, which provided a large speed up. MBR decoding/combination was used to produce the final output. Unless stated otherwise, performance with the 3-gram model is quoted.

4.1. Phonetic and Graphemic Models

Model		Single		Ph/Gr Comb	
		%WER	%Rel	%WER	%Rel
DNN	Ph	27.8	—	26.3	-5.4
	Gr	30.7	+10.4		
TDNN	Ph	24.4	—	23.0	-5.7
	Gr	26.9	+10.3		
LSTM	Ph	25.0	—	23.2	-7.2
	Gr	26.7	+6.8		
TDNN-LSTM	Ph	23.4	—	21.7	-7.3
	Gr	25.0	+6.8		

Table 1. %WER of phonetic and graphemic systems and their MBR combination on dev17b.

The impact of the acoustic model on the performance difference between phonetic (Ph) and graphemic (Gr) systems is illustrated in Table 1. The second column shows the relative degradation in performance of the graphemic system. As the complexity of the model and the span of available temporal information increases, the difference between phonetic and graphemic system WERs drops from 10.4 to 6.8% relative. The largest drop happens when the LSTM units are used to model longer history information. This implies that graphemic systems are more sensitive to shorter histories than are phonetic systems. The third column in Table 1 shows that as the graphemic system gets more competitive, the gain from combining it with the phonetic system increases from 5.4 to 7.3% relative.

Model	Context	%WER	RTF
Ph	Bi-phone	23.4	0.9
	Mono-phone	23.9	0.7
Gr	Bi-grapheme	25.0	0.8
	Mono-grapheme	26.2	0.6

Table 2. %WER of context dependent and independent phonetic and graphemic TDNN-LSTM models on dev17b.

Graphemic systems are also expected to be sensitive to the choice of acoustic modelling context. Wider contexts should be more suitable for graphemic systems as they can better account for the mismatch between the orthographic and spoken form. However, shorter contexts are appealing due to their simplicity and speed of training as well as decoding. Table 2 shows that phonetic systems are significantly more robust when bi-phone units are replaced with mono-phone units. Though mono-grapheme units

[1] For the LF-MMI systems, experiments showed that the performance of the tri-phone models was slightly worse than that of the bi-phone models, which was consistent with the discussions in https://groups.google.com/forum/#!topic/kaldi-developers/NNx2xVoYctU/discussion.

[2] The splicing indexes per layer can be described as {-1,0,1} {-1,0,1} {-1,0,1,2} {-3,0,3} {-3,0,3} {-6,-3,0} {0} using the notation of [8, 11].

[3] The architecture can be described as {-2,-1,0,1,2} {-1,0,1} $\mathcal{L}$ {-3,0,3} {-3,0,3} $\mathcal{L}$ {-3,0,3} {-3,0,3} $\mathcal{L}$, where $\mathcal{L}$ represents an LSTMP layer with 512 cells and 128-dimensional recurrent and non-recurrent projections, using notation of [8, 11].

yield twice as large a degradation as mono-phone units, the simplicity of graphemic lexicons offers an interesting compromise. Both context-independent systems are approximately 25% faster than their context-dependent counterparts as shown by the real time factor (RTF) in Table 2.

4.2. Model combination

Training Criterion	%WER	Comb %WER	Comb %Rel
sMBR	23.7	21.3	-9.0
LF-MMI	23.4		

Table 3. %WER of sMBR and LF-MMI trained phonetic TDNN-LSTM models and their MBR combination on dev17b.

Rather than combining phonetic and graphemic systems, it is also possible to combine systems from any diverse ensemble as discussed in Section 3. One simple way to produce an additional system is to utilise an alternative training criterion such as sMBR training. Table 3 shows that sMBR training yields a competitive model and combination gains between systems with these criteria is larger than that between phonetic and graphemic systems in Table 1. This can partly be attributed to the larger performance differences between the phonetic and graphemic systems being combined.

Additional systems can also be generated using simpler approaches. For example, the use of model smoothing does not require another model to be trained. In this work, 20 models with an iteration interval of 6 were taken from the final epoch of LF-MMI training and their combination weights were estimated on a subset of training data as discussed in Section 3. Table 4 shows that model smoothing is an effective way to improve system performance for both graphemic and phonetic systems. Additionally by performing model smoothing the difference between the phonetic and graphemic systems is reduced (+6.0%). Though the gains from combining phonetic and graphemic systems decrease after model smoothing, dropping from 7.3 to 5.6% relative, there is still a large gain in performance in the combined systems after model smoothing, yielding a better performance than modifying the training criterion.

Model	%WER Ph	%WER Gr	Ph/Gr Comb %WER	Ph/Gr Comb %Rel
Single	23.4	25.0	21.7	-7.3
Smooth	21.5	22.8	20.3	-5.6

Table 4. %WER of phonetic and graphemic TDNN-LSTM models with and without model smoothing on dev17b.

Alternatively, random ensembles [14] can be built by changing the random seed used to initialise models for LF-MMI training. This is more expensive than model smoothing, but allows additional diversity to be introduced. LF-MMI training may benefit more from random ensemble generation, as it avoids the cross-entropy initialisation stage of approaches such as sMBR training, where the same targets are normally used for all system, possibly reducing the diversity of the final systems after sequence training. In this work, an ensemble of 3 TDNN-LSTM models was created by building 2 additional models using different seeds for random parameter initialisation. Table 5 shows that although the WER standard deviation across systems is small, the cWER is large suggesting that these systems may be complementary. To put the cWER number in context, an ensemble of sMBR trained models on the AMI IHM task with a mean WER of 25% had a cWER of 12%. The last block in Table 5 shows that ensemble combination of multiple single models yields the large gains of the approaches examined in this work. Given the large gains from model smoothing, it is interesting to examine ensembles of smoothed models. These are also shown in Table 5. As expected the cWERs for the ensembles are reduced, as model smoothing reduces the diversity from the precise stopping points. However, there are still large gains of over 7% relative from the ensemble combination. Additionally, the difference between phonetic and graphemic, smoothed or unsmoothed, systems when combining random ensembles has been reduced to just 5% relative.

Model		%WER μ	%WER σ	%cWER	Ensemble Comb %WER	Ensemble Comb %Rel
Ph	Single	23.5	0.06	17.9	20.9	-11.1
	Smooth	21.6	0.10	13.5	20.0	-7.4
Gr	Single	25.0	0.10	20.4	22.1	-11.6
	Smooth	22.8	0.06	14.9	21.0	-7.9

Table 5. %WER of phonetic and graphemic ensembles of TDNN-LSTM models with and without model smoothing on dev17b.

Given the small difference between the phonetic and graphemic ensembles, additional gains from combining the systems might be expected. However, the extensive use of combination techniques means that diversity between these ensembles has already been significantly reduced. Table 6 shows that combining phonetic and graphemic ensembles yields only 0.5% absolute or 2.5% relative reduction in WER [4]. At this point, it is interesting to see if improved language modelling approaches can yield further benefits. The last column in Table 6 shows that 4-gram LM rescoring reduces the WER from 19.5 to 18.8%. The RNNLM gave an additional improvement yielding a final error rate of 17.9% on this task.

Model		Comb tg	Ph/Gr Comb tg	Ph/Gr Comb fg	Ph/Gr Comb +rnn
Ph	Ensemble	20.0	19.5	18.8	17.9
Gr		21.0			

Table 6. %WER of the final MGB-3 system on dev17b.

5. CONCLUSION

This paper has investigated whether the recent advances in deep learning based approaches have enabled graphemic English ASR systems to reach the performance level of traditionally used phonetic systems. It was found that a combination of long-span temporal history and future information with context-dependent graphemic units is important to obtain competitive performance for graphemic English ASR systems. The relative difference betweqen phonetic and graphemic systems can be further reduced by employing system combination approaches, model smoothing and random ensemble methods were both found to be effective. The combination of these two methods yielded a graphemic English ASR system for multi-genre broadcast transcription that is only 5% relatively worse than an equivalent phonetic English ASR system, and is complementary.

[4] It is worth noting that the performance of combining an ensemble of two phonetic systems was 20.2%. Thus simply enlarging the size of the phonetic ensemble is not expected to match this graphemic/phonetic ensemble performance.

6. REFERENCES

[1] S. Kanthak and H. Ney. Context-dependent acoustic modeling using graphemes for large vocabulary speech recognition. In *Proc. ICASSP*, 2002.

[2] M. Killer, S. Stüker, and T. Schultz. Grapheme based speech recognition. In *Proc. INTERSPEECH*, 2003.

[3] M. J. F. Gales, K. M. Knill, and A. Ragni. Unicode-based graphemic systems for limited resource languages. In *Proc. ICASSP*, pages 5186–5190, 2015.

[4] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. ICML*, pages 369–376, 2006.

[5] F. Eyben, M. Wöllmer, B. Schuller, and A. Graves. From speech to letters — using a novel neural network architecture for grapheme based ASR. In *Proc. ASRU Workshop*, pages 376–380, 2009.

[6] H. Sak, A. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Proc. INTERSPEECH*, 2014.

[7] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, 22(10):1533–1545, 2014.

[8] V. Peddinti, D. Povey, and S. Khudanpur. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Proc. INTERSPEECH*, 2015.

[9] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *Proc. ICML*, pages 1764–1772, 2014.

[10] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak. Convolutional, long short-term memory, fully connected deep neural networks. In *Proc. ICASSP*, pages 4580–4584, 2015.

[11] V. Peddinti, Y. Wang, D. Povey, and S. Khudanpur. Low latency acoustic modeling using temporal convolution and LSTMs. *IEEE Signal Processing Letters*, 25(3):373–377, 2017.

[12] D. Povey and B. Kingsbury. Evaluation of proposed modifications to MPE for large scale discriminative training. In *Proc. ICASSP*, 2007.

[13] J. H. M. Wong and M. J. F. Gales. Multi-task ensembles with teacher-student training. In *Proc. ASRU Workshop*, 2011.

[14] J. H. M. Wong and M. J. F. Gales. Sequence student-teacher training of deep neural networks. In *Proc. INTERSPEECH*, 2016.

[15] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al. The Kaldi speech recognition toolkit. In *Proc. ASRU Workshop*, 2011.

[16] J. Odell. The use of context in large vocabulary speech recognition, 1995. PhD thesis.

[17] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[18] J. G. Fiscus. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). In *Proc. ASRU Workshop*, pages 347–354, 1997.

[19] G. Evermann and P.C. Woodland. Large vocabulary decoding and confidence estimation using word posterior probabilities. In *Proc. ICASSP*, 2000.

[20] H. Xu, D. Povey, L. Mangu, and J. Zhu. Minimum Bayes risk decoding and system combination based on a recursion for edit distance. *Computer Speech & Language*, 25(4):802–828, 2011.

[21] H. Wang, A. Ragni, M. J. F. Gales, K. M. Knill, P. C. Woodland, and C. Zhang. Joint decoding of tandem and hybrid systems for improved keyword spotting on low resource languages. In *Proc. INTERSPEECH*, 2015.

[22] T. G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems (Proceedings of MCS 2000)*, pages 1–15. Springer-Verlag, LNCS, 2000.

[23] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[24] O. Siohan, B. Ramabhadran, and B. Kingsbury. Constructing ensembles of ASR systems using randomized decision trees. In *Proc. ICASSP*, 2005.

[25] P. Bell, M. J. F. Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester, et al. The MGB challenge: Evaluating multi-genre broadcast media recognition. In *Proc. ASRU Workshop*, pages 687–693, 2015.

[26] P. C. Woodland, X. Liu, Y. Qian, C. Zhang, M. J. F. Gales, P. Karanasou, P. Lanchantin, and L. Wang. Cambridge University transcription systems for the multi-genre broadcast challenge. In *Proc. ASRU Workshop*, pages 639–646, 2015.

[27] L. Wang, C. Zhang, P. C. Woodland, M. J. F. Gales, P. Karanasou, P. Lanchantin, X. Liu, and Y. Qian. Improved DNN-based segmentation for multi-genre broadcast audio. In *Proc. ICASSP*, pages 5700–5704, 2016.

[28] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur. Purely sequence-trained neural networks for asr based on lattice-free MMI. In *Proc. INTERSPEECH*, pages 2751–2755, 2016.

[29] T. Mikolov, M. Karafiát, L. Burget, et al. Recurrent neural network based language model. In *Proc. INTERSPEECH*, 2010.

[30] X. Chen, X. Liu, M. Gales, and P. Woodland. CUED-RNNLM an open-source toolkit for efficient training and evaluation of recurrent neural network language models. In *Proc. ICASSP*, 2015.

[31] X. Chen, X. Liu, Y. Wang, M. Gales, and P. Woodland. Efficient training and evaluation of recurrent neural network language models for automatic speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(4):2146–2157, 2016.

[32] X. Liu, X. Chen, Y. Wang, M. Gales, and P. Woodland. Two efficient lattice rescoring methods using recurrent neural network language models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(8):1438–1449, 2016.