MULTITASK LEARNING AND SYSTEM COMBINATION FOR AUTOMATIC SPEECH RECOGNITION

Olivier Siohan and David Rybach

Google Inc., New York

ABSTRACT

In this paper we investigate the performance of an ensemble of convolutional, long short-term memory deep neural networks (CLDNN) on a large vocabulary speech recognition task. To reduce the computational complexity of running multiple recognizers in parallel, we propose instead an early system combination approach which requires the construction of a static decoding network encoding the multiple context-dependent state inventories from the distinct acoustic models. To further reduce the computational load, the hidden units of those models can be shared while keeping the output layers distinct, leading to a multitask training formulation. However in contrast to the traditional multitask training, our formulation uses all predicted outputs leading to a multitask system combination strategy. Results are presented on a Voice Search task designed for children and outperform our current production system.

***Index Terms*—** system combination, multitask learning, children's speech, ROVER

1. INTRODUCTION

State-of-the-art acoustic models for automatic speech recognition (ASR) are typically based on deep neural networks (DNN) trained to predict the posterior probabilities of a set of context-dependent (CD) HMM states [1]. In the past years, various architectures have been developed ranging from deep feed-forward neural networks, convolutional neural networks (CNN) [2], long short-term memory (LSTM) networks [3], or the recently proposed convolutional LSTM deep neural network (CLDNN) [4].

While those models are often trained to predict a single CD state corresponding to a short contextual input window of log-spectral feature vectors, the multitask learning (MTL) paradigm proposed in [5] was recently applied to acoustic modeling [6, 7, 8]. In MTL learning, multiple related tasks are jointly learned, such as for example predicting CD states and context-dependent graphemes [7] simultaneously. It was observed that when both the input features and the hidden units are shared, MTL learning may improve generalization [5] as the learning of one task may help learning the other tasks better.

In [6], a DNN is trained to predict CD states as primary task and either phone label, state context, or phone context as secondary task. At recognition time the secondary softmax outputs are discarded and only the primary softmax outputs are used to carry out the search. A related approach is proposed in [8] where a DNN is trained to jointly predict CD states as well as monophone targets. During recognition, the monophone outputs are ignored and the CD state posteriors only are used to drive the search procedure. A similar MTL-DNN architecture is used in [7] to jointly train a DNN model to predict triphone and trigrapheme classes. The 2 sets of class posteriors are then fed separately to a triphone-based and a trigrapheme-based decoder and their recognition hypotheses are combined using ROVER [9].

In this work, we address multitask learning within the context of system combination and propose to jointly train a CLDNN-based system to predict CD state targets from multiple and distinct state inventories (e.g. CD and CI states). In contrast to [6, 8], all predicted outputs are used and combined into an integrated score at decode time, achieving some form of early system integration with a single search procedure, as opposed to the late system integration of [7] which requires running multiple decodings in parallel. All experiments are conducted on a large vocabulary speech recognition task specifically designed for recognizing children's speech [10] using CLDNN [4] acoustic models trained on several thousand hours of training data.

2. TASK AND DATABASES

The focus of this paper is to construct acoustic models specifically designed for recognizing children's speech to support tasks such as the YouTube Kids mobile application [11]. While this application is targeted for children, it should deliver reasonable performance on adult speech as well. For that reason, our training set consists of a mix of 1.9M voice search (VS) utterances that were manually transcribed and labeled as child speech combined with 1.3M manually transcribed VS utterances from the general VS traffic, corresponding to a grand total of 2,100 hours of speech. Two test sets are used for evaluation purposes: a 25k VS utterances set from adult speakers and a 16k VS utterances set from children. All our data sets are anonymized.

978-1-4799-7291-3/15/$31.00 ©2015 IEEE

In this paper all our acoustic models are trained from scratch using the online training procedure described in [12, 13]. Our baseline training procedure operates in 4 stages. First, we flat-start a context-independent DNN model, run decision tree state tying using Chou's partitioning algorithm [14], and train a DNN model with 8 hidden layers of 2560 units on the resulting CD state inventory using a cross-entropy objective function. Then, we refine our DNN model using sequence training until convergence. Next, we use the alignments obtained from the sequence-trained DNN to bootstrap a CLDNN model using a cross-entropy criterion. Last, we refine the CLDNN model using sequence training. The topology of the CLDNN model consists of one CNN layer, 2 LSTM layers and 2 DNN layers, described in detail in [10]. The acoustic features are 40-dim log mel-spaced filterbank coefficient without any temporal context. The recurrent network is unrolled for 20 time steps for training and the output of the state label is delayed by 5 frames since we found that information about future frames helps predicting the label of the current frame [10].

All evaluations are conducted using a child-friendly language model (LM) consisting of about 100M n-grams for a vocabulary of 4.6M words. The LM was constructed according to the procedure described in [10] which greatly reduces the chance of outputting an offensive transcript compared to the language model used in the general VS application.

3. SYSTEM COMBINATION

System combination is often used in speech recognition as a post-recognition strategy to combine the outputs of multiple systems for example using a majority voting approach such as ROVER [9] or confusion network combination [15]. Ideally, one would like to design those systems to have similar performance but make independent errors so that they would benefit from a voting-based combination strategy. Unfortunately, while work such as [16] enforces some complementarity among the different systems being trained, most acoustic modeling approaches to train an ensemble rely on heuristics such as using different input feature representations or different system architectures. In this work, we also rely on an ad-hoc approach to build an ensemble by constructing multiple CD state inventories via randomization of the decision tree state tying procedure [17]. While there is no theoretical guarantee that such an ensemble training strategy would lead to improved recognition accuracy, it was shown to be effective on various tasks [17, 18]. In addition, we will see that such a procedure fits well with the notion of MTL learning since the multiple systems operate on the same input features, define related classification tasks and are candidate for sharing their hidden units.

In section 3.1, we first illustrate that an ensemble of multiple CLDNN-based acoustic models trained on various CD state inventories either of different sizes or constructed using randomized decision trees can lead to significant error reductions when their recognition hypotheses are combined with majority-voting ROVER. Unfortunately, such a late combination strategy is computationally expensive as it requires running multiple recognizers in parallel. To lower the computational requirements, we then describe in Section 3.2 an early combination strategy which combines the acoustic scores of different models taking into account the distinct CD state inventories and enabling the use of a single decoding procedure. Next, in Section 4, rather than training those acoustic models independently, we instead cast the problem as a multitask learning procedure by sharing the input and all hidden layers of those CLDNN models while keeping distinct softmax outputs corresponding to the different CD state inventories.

3.1. Late system combination

In [17], multiple GMM-HMM acoustic models were constructed by randomizing the phonetic decision tree state tying procedure. Unlike the regular CART-based decision tree procedure which grows each tree by selecting an optimal binary split at each step, the randomized procedure randomly selects a near-optimal split from the top N best split candidates with N typically set to 5. By repeating the state tying procedure multiple times using random seeds, one can then construct a distinct system for each state inventory.

In this paper, we construct our decision trees using Chou's partitioning algorithm which avoids having to specify a set of phonetic questions as required when using CART. The randomization of the clustering is carried out slightly differently compared to CART. Chou's algorithm includes a K-means step which iteratively assigns states to clusters and recomputes the clusters' centroids, repeating this procedure until convergence. Rather than iterating until convergence, we randomly stop the K-means partitioning in the last 5 steps before reaching convergence. This provides sufficient randomization to construct multiple state inventories in a systematic fashion without impairing the quality of the resulting model, as illustrated next.

Table 1 demonstrates the effectiveness of this approach on 3 different test sets. An ensemble of 5 independently trained CLDNN models is constructed using sequence training. The first 3 systems have distinct CD state inventories of 12k states each constructed via randomization and the last 2 systems have 6k states each, also constructed via randomization. One can notice that those 5 systems have very similar performance and that the size of the CD state inventories (6k vs. 12k states) has little impact on the word error rate. The second part of Table 1 reports the WER when applying ROVER on the output of the first 3, 4 and 5 systems. On all test sets, a significant 5% to 7% WER reduction is obtained at the expense of running 5 systems in parallel.

Individual Systems	Child	Adult
#1 (12k states)	9.4%	13.2%
#2 (12k states)	9.7%	13.4%
#3 (12k states)	9.4%	13.3%
#4 (6k states)	9.6%	13.3%
#5 (6k states)	9.4%	13.4%
ROVER	Child	Adult
#1, #2, #3	8.9%	12.6%
#1, #2, #3, #4	8.8%	12.4%
#1, #2, #3, #4, #5	8.7%	12.3%

Table 1. Top: Individual WER on the Child and Adult test sets (%) for 3 systems with a 12k CD state inventory and 2 systems with a 6k CD state inventory. Bottom: ROVER between systems #1 to #3, #1 to #4, and #1 to #5.

3.2. Early system combination

To avoid running multiple recognizers in parallel, we propose to use instead a single decoder and combine the acoustic scores of the different models during the search. Our decoder relies extensively on Finite State Transducer (FST) technology [19] and operates on a static decoding graph constructed by composing a context-dependency transducer C with a lexicon transducer L and a grammar transducer G. The C transducer is defined by the decision tree state tying procedure and converts sequences of HMM symbols onto sequences of phone symbols. Note that the size of C is related to the degree of tying between states and that a larger CD state inventory will lead to a larger C transducer. The L transducer translates sequences of phone symbols into sequences of words, and the G transducer attaches language model probabilities to the sequences of words. The resulting $C \circ L \circ G$ graph, denoted CLG for short, translates sequences of HMM symbols onto weighted sequences of words and is used to drive the search procedure. During search, HMM arcs are dynamically expanded onto their corresponding sequence of CD state symbols. Such a graph structure combined with the acoustic model providing $p(X|S)$ where X is the input feature vector and S a CD state from the CD state inventory enables running Viterbi search. Note that when using DNN-based acoustic models, $p(X|S)$ is approximated as $p(S|X)/p(S)$, the posterior probability of state S provided by the softmax output scaled by the prior distribution of state S.

To enable the use of a single decoding procedure combining the acoustic scores of multiple models with different state inventories, we have to construct a single CLG graph where the HMM arcs are defined on tuples of HMM symbols from the multiple systems rather than a single HMM label. In many regards this is similar to the use of decision tree arrays described in [20] for dynamic decoders but applied in our case to a static decoder. The procedure to construct this modified CLG graph only involves modifying the C transducer and is otherwise unchanged. Our approach also bears some similarity with the work in [21] which integrates two decision trees with leaves holding a 2-tuples of state indices, as well as with the work in [22] which uses a single decoder integrating multiple models.

We will describe our procedure using an example with 2 systems but it can be generalized to an arbitrary number of systems. Suppose that a given CD state inventory is constructed for a first system where a given triphone, say h-@+V, represents center phone @ with phone h and V as left and right context. Lets assume that this triphone is mapped to a given HMM symbol, say HMM^1_5, defined as a sequence of CD state symbols, for example $\text{HMM}^1_5 \equiv (\text{CD}^1_{1\_21}, \text{CD}^1_{2\_35}, \text{CD}^1_{3\_42})$ corresponding to a 3-state HMM topology. The superscript used on the HMM and CD state symbol names refers to the system index (here system 1) while the subscript on the CD state names refers to the HMM state position (1, 2 or 3, assuming a 3-state HMM topology) and tied-state indices and the subscript on the HMM names refers to the HMM indices. For the sake of this example, the CD states and HMMs indices are arbitrary. Suppose also that a second state inventory is constructed, mapping the same triphone h-@+V to, say, HMM symbol HMM^2_9 defined for example as $\text{HMM}^2_9 \equiv (\text{CD}^2_{1\_53}, \text{CD}^2_{2\_5}, \text{CD}^2_{3\_17})$.

Continuing with this example, we can then define a "meta" HMM symbol to represent triphone h-@+V as a tuple of HMM symbols from the individual systems, for example here $\text{HMM}'_7 \equiv \langle \text{HMM}^1_5, \text{HMM}^2_9 \rangle$. One can then enumerate all possible triphones and construct the entire meta-HMM inventory from the corresponding tuples of system-specific HMM symbols. This results in the definition of a meta C transducer translating sequences of HMM tuples onto phone sequences. If we further assume that HMMs have the same topology in both systems, each meta HMM symbol can be represented as a sequence of meta CD state symbols, for example, $\text{HMM}'_7 \equiv (\text{CD}'_{1\_6}, \text{CD}'_{2\_1}, \text{CD}'_{3\_23})$, with each meta CD state symbol defined as a tuple of CD state symbols from the individual systems, here $\text{CD}'_{1\_6} \equiv \langle \text{CD}^1_{1\_21}, \text{CD}^2_{1\_53} \rangle$, $\text{CD}'_{2\_1} \equiv \langle \text{CD}^1_{2\_35}, \text{CD}^2_{2\_5} \rangle$, and $\text{CD}'_{3\_23} \equiv \langle \text{CD}^1_{3\_42}, \text{CD}^2_{3\_17} \rangle$. The meta-$C$ transducer can be directly used to construct the decoding graph leading to a graph translating sequences of HMM tuples onto word sequences. In Table 2, we report the number of CD states and HMM symbols corresponding to individual system configurations and the resulting meta-C transducer obtained by combining those individual C transducers. One can observe that with a CD state inventory of 12k states, the number of HMM symbols is around 43k. A meta-C transducer combining those two 12k state systems leads to a total number of ∼56k meta-HMM symbols and combining all 4 systems leads to a total number of ∼60k meta-HMM symbols. Despite this increase in the number of HMM symbols, it has a negligeable impact on the size of the resulting static CLG graph.

Note that the graph construction described above should be accompanied with specifying how to compute $p(X|S')$

System	# CD states	# HMMs
#1	12,000	43,538
#2	12,000	42,616
#4	6,000	29,456
#5	6,000	33,814
Meta(#1, #2)	46,541	55,981
Meta(#4, #5)	25,567	50,356
Meta(#1, #2, #4, #5)	84,961	60,751

Table 2. Number of CD-state and HMM symbols for 4 individual systems and corresponding number of meta CD states and meta HMM symbols obtained by constructing a meta-C transducer combining systems (#1, #2), systems (#4, #5) and all 4 systems.

where S' is an arbitrary meta CD state defined as a tuple of CD states from the 2 systems, $S' \equiv \langle S^1, S^2 \rangle$. Specifically, we define $p(X|\langle S^1, S^2 \rangle) = f(p(X|S^1), p(X|S^2))$ where $f()$ is an arbitrary predefined function defining how to combine the acoustic scores of the individual systems. It can for example be defined to compute the average of the acoustics scores of the different systems or to select the highest likelihood score from the ensemble. Note that because our decoder operates on negative log-probabilities or costs, selecting the highest likelihood score is equivalent to selecting the minimum cost and we will use the term cost hereafter. In Table 3 we report the WER obtained using a single decoding graph for different system configurations. The top part reports the WER using a single graph constructed from 2 systems with 12k CD states each using a score combination function returning either the cost of the first system, of the second system, or selecting the minimum cost between the 2 systems and shows that the early integration of the acoustic score provides an absolute ∼0.4% WER reduction over the individual system on the Child test set. The second part of the table report similar results but using 2 systems of 6k states each. Last, we also report results using a single decoding graph integrating those 4 acoustic models, leading to 9.0% WER. This is slighly worse than the late system combination results reported in Table 1 but offers a significant reduction of the computational cost since it involves a single decoding.

4. MULTITASK TRAINING

Given that the different systems constructed in Section 3 operate on the same input features and correspond to related tasks (predicting tied-states from different CD states inventories), the training procedure can be reformulated as a multitask learning. Instead of training the multiple systems independently, as represented on the left part of Fig. 1, we share the internal structure of our CLDNN models keeping only distinct softmax outputs corresponding to the different CD state inventories. In essence, the network architecture is very similar to the one used in the multitask learning procedure proposed in [8] except that in our case all softmax layers attempt to predict CD state targets (from different inventories). However, a significant difference w.r.t. the work in [8] is that at recognition time, all softmax outputs are used and integrated during the search based on the procedure described in Section 3.2.

System	Child	Adult
#1 (12k states)	9.5%	13.2%
#2 (12k states)	9.6%	13.3%
MinCost(#1, #2)	9.1%	12.8%
#4 (6k states)	9.6%	13.3%
#5 (6k states)	9.4%	13.4%
MinCost(#4, #5)	9.2%	12.8%
MinCost(#1, #2, #4, #5)	9.0%	12.8%

Table 3. Top part: WERs (%) using a single decoding graph constructed from model #1 and #2 and using acoustic costs from model #1 only, model #2 only, or the minimum cost between model #1 and #2. Middle part: Similar to top part but using model #4 and #5. Bottom part: WER using a single decoding graph constructed from all 4 models and using minimum cost between all models.

In a first series of experiments we trained 2 DNNs based on 2 sets of 6k CD-states (obtained via randomized decision trees) first using cross-entropy training and followed by sequence training. Each training utterance was then separately aligned using the 2 models, so that each training frame would be labeled with a pair of CD states, one from each system. Those multi CD state targets were then used to train a CLDNN model using cross-entropy in a multitask fashion, as represented in Fig. 2. Note that unlike the previous sections where all CLDNN models were sequence-trained, all CLDNN models in this section are trained using cross-entropy since multitask training is not directly amenable to sequence training. As in Section 3.2, a single decoding graph was constructed from the multiple CD state inventories and recognition was carried out using either the softmax output from the first or second CD state inventory or a combination of their acoustic costs using a minimum cost function. Results are given in Table 4. Considering that single-task training of a 12k states CLDNN using cross entropy gives a 11.0% WER on the Child test set, the multitask training does not here lead to a significant performance difference over single task training.

To further evaluate the performance of multitask training and for faster experimental turnaround, we adopted a slightly different training procedure where an existing 12k states sequence-trained DNN model was used to align all our training data and the state alignments were relabeled based on various CD state inventories, as represented in Fig. 3. We first constructed a set of targets using 2 softmax outputs, one corresponding to a 12k CD state inventory and the other a 1k CD state inventory and ran multitask learning of a

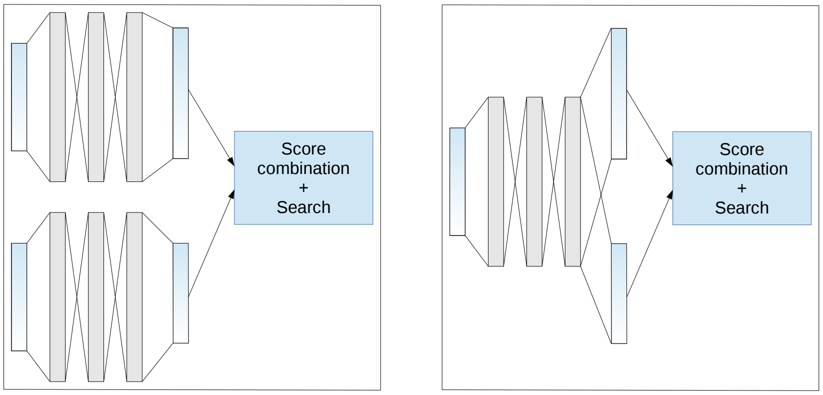

Fig. 1. Single-task training (left): the models are trained independently and their scores are combined during search. Joint multitask training (right): the model is trained to jointly predict multiple CD state targets and scores are combined during search.

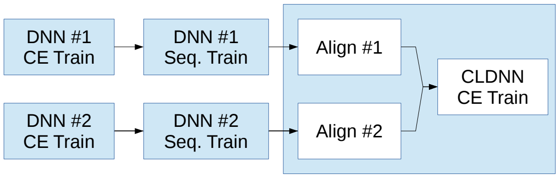

Fig. 2. Multitask training procedure using individual models for alignment.

System	Child	Adult
Softmax#1 (6k states)	11.0%	14.2%
Softmax#2 (6k states)	11.0%	14.3%
MinCost(#1, #2)	10.9%	14.2%

Table 4. Multitask training of 2 distinct 6k CD state inventories. Recognition is carried out using a single decoding graph and selecting either the first softmax outputs, second softmax outputs, or minimum cost combination during search.

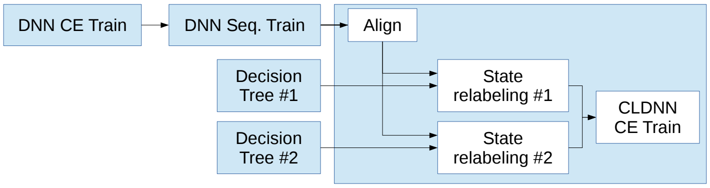

Fig. 3. Multitask training procedure using a common alignment model and relabeling the resulting state sequence based on different decision trees.

System	Child	Adult
Softmax#1 (12k states)	11.3%	14.6%
Softmax#2 (1k states)	12.4%	16.0%
MinCost(#1, #2)	11.3%	14.7%

Table 5. Multitask training for 12k/1k CD state inventories. Recognition is carried out using a single decoding graph and selecting the costs from either the first softmax, second softmax, or minimum cost combination during search.

cross-entropy CLDNN model on those targets. Experimental results are given in Table 5. As expected, decoding using the 1k state softmax outputs performs significantly worse than when using the 12k state softmax outputs. When combining both outputs during search using the minimum cost combination, we did not observe any performance improvement over using the 12k outputs only. Given the size of our training set (2,100 hours), this seems to confirm some of the results from [8] where multitask training led to diminishing returns for increasing amount of training data.

We then constructed 3 CD state inventories, 2 consisting of 6k CD states and one of 1k CD states and ran multitask training of a cross-entropy CLDNN model on those 3 output targets. Results are available in Table 6. Despite having the same total number of output targets, this system performed slightly worse than when using the 12k/1k set of targets above. Again, the multitask training does not lead to any performance improvement.

5. CONCLUSIONS

In this paper, we showed that multiple CLDNN-based acoustic models trained on distinct state inventories constructed using randomized decision trees can outperform a single system when using either late or early system combination. While a late combination approach such as ROVER may not be attractive since it requires running multiple recognizers in parallel, we have proposed to construct a specialized C transducer encoding the multiple decision trees on each arc. This enables

System	Child	Adult
Softmax#1 (6k states)	11.5%	15.0%
Softmax#2 (6k states)	11.5%	15.0%
Softmax#3 (1k states)	12.5%	16.4%
MinCost(#1, #2, #3)	11.5%	15.1%

Table 6. Multitask training for 6k/6k/1k state inventories. Recognition is carried out using a single decoding graph and selecting the costs either from the first softmax, second softmax, or minimum cost combination during search.

the construction of a single decoding graph which can be used to support running recognition using multiple acoustic models with different state inventories, either individually or by combining their scores during the search. Using a recently improved language model, an early combination of 4 acoustic models gives an 8.0% WER on the Child test set, significantly outperforming our current production models which operates at a 8.7% WER. To further reduce the computational load, those multiple acoustic models can share their input and hidden units, leading to a multitask training formulation. Unfortunately, we did not observe any learning transfer between the different tasks and the multitask training did not improve performance over single task training, which we explain by the large amount of training data used in our experiments.

6. REFERENCES

[1] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara Sainath, and Brian Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[2] Tara N. Sainath, Abdel rahman Mohamed, Brian Kingsbury, and Bhuvana Ramabhadran, "Deep convolutional neural networks for LVCSR," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[3] Hasim Sak, Andrew Senior, and Francoise Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Conference of the International Speech Communication Association (InterSpeech)*, 2014.

[4] Tara N. Sainath, Oriol Vinyals, Andrew Senior, and Hasim Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[5] Rich Caruana, *Multitask learning*, Ph.D. thesis, Carnegie Mellon University, 1997.

[6] Michael Seltzer and Jasha Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[7] Dongpeng Chen, Brian Mak, Cheung-Chi Leung, and Sunil Sivadas, "Joint acoustic modeling of triphones and trigraphemes by multi-task learning deep neural networks for low-resource speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[8] Peter Bell and Steve Renals, "Regularization of context-dependent deep neural networks with context-independent multi-task training," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, 2015.

[9] Jonathan G. Fiscus, "A post-processing system to yield reduced word error rates:recognizer output voting error reduction (ROVER)," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 1997.

[10] Hank Liao, Golan Pundak, Olivier Siohan, Melissa Carroll, Noah Coccaro, Qi-Ming Jiang, Tara Sainath, Andrew Senior, Francoise Beaufays, and Michiel Bacchiani, "Large vocabulary automatic speech recognition

for children," in *Conference of the International Speech Communication Association (InterSpeech)*, 2015.

[11] "Introducing the newest member of our family, the YouTube Kids app—available on Google Play and the App Store," 2015, http://youtube-global.blogspot.com/2015/02/youtube-kids.html.

[12] Michiel Bacchiani and David Rybach, "Context dependent state tying for speech recognition using deep neural network acoustic models," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[13] Michiel Bacchiani, Andrew Senior, and Georg Heigold, "Asynchronous, online, GMM-free training of a context dependent acoustic model for speech recognition," in *Conference of the International Speech Communication Association (InterSpeech)*, 2014.

[14] Philip A. Chou, "Optimal partitioning for classification and regression trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 340–354, 1991.

[15] Lidia Mangu, Eric Brill, and Andreas Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, vol. 14, pp. 373–400, 2000.

[16] Yuuki Tachioka, Shinji Watanabe, Jonathan Le Roux, and John R. Hershey, "A generalized discriminative training framework for system combination," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2013.

[17] Olivier Siohan, Bhuvana Ramabhadran, and Brian Kingsbury, "Constructing ensembles of ASR systems using randomized decision trees," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2005.

[18] Catherine Breslin and Mark J. F. Gales, "Complementary system generation using directed decision trees," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007.

[19] Mehryar Mohri, Fernando Pereira, and Michael Riley, *Springer Handbook of Speech Processing*, chapter Speech Recognition with Weighted Finite-State Transducers, pp. 559–584, Springer, 2008.

[20] Hagen Soltau, George Saon, and Brian Kingsbury, "The IBM Attila speech recognition toolkit," in *IEEE Workshop on Spoken Language Technology (SLT)*, 2010.

[21] Andras Zolnay, *Acoustic Feature Combination for Speech Recognition*, Ph.D. thesis, RWTH Aachen-University, 2006.

[22] Peter Beyerlein, "Discriminative model combination," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1998, vol. 1, pp. 481–484.