Interspeech 2018
2-6 September 2018, Hyderabad

A Multistage Training Framework For Acoustic-to-Word Model

Chengzhu Yu[*], *Chunlei Zhang*[*], *Chao Weng, Jia Cui, Dong Yu*

Tencent AI Lab, Bellevue, USA

(czyu, cweng, jiaacui, dyu)@tencent.com, chunlei.zhang@utdallas.edu

Abstract

Acoustic-to-word (A2W) prediction model based on Connectionist Temporal Classification (CTC) criterion has gained increasing interest in recent studies. Although previous studies have shown that A2W system could achieve competitive Word Error Rate (WER), there is still performance gap compared with the conventional speech recognition system when the amount of training data is not exceptionally large. In this study, we empirically investigate advanced model initializations and training strategies to achieve competitive speech recognition performance on 300 hour subset of the Switchboard task (SWB-300Hr). We first investigate the use of hierarchical CTC pretraining for improved model initialization. We also explore curriculum training strategy to gradually increase the target vocabulary size from 10k to 20k. Finally, joint CTC and Cross Entropy (CE) training techniques are studied to further improve the performance of A2W system. The combination of hierarchical-CTC model initialization, curriculum training and joint CTC-CE training translates to a relative of 12.1% reduction in WER. Our final A2W system evaluated on Hub5-2000 test sets achieves a WER of 11.4/20.8 for Switchboard and CallHome parts without using language model and complex decoder.

Index Terms: speech recognition, end-to-end, acoustic-to-word, all-neural

1. Introduction

The goal of automatic speech recognition system is to recognize spoken words. However, speech recognition systems [1–3] have been relying on modeling sub-word units except some isolated word speech recognition tasks. This is mostly due to the difficulty of directly using word as acoustic modeling unit. The challenge of using word as acoustic modeling unit come from (1) the sparsity of training data, and (2) capturing long term dependencies between acoustic frames. With the recent success in applying recurrent neural network (RNN) and its variants in speech recognition, modeling long term dependencies of acoustic frames for word prediction becomes feasible. In recent study [4,5], the authors have successfully proposed a direct acoustic-to-word (A2W) system that achieves state-of-the-art speech recognition performance by leveraging 125,000 hours of training data collected from Youtube videos with captions. As the speech recognition system in [5] composed of single neural network trained in end-to-end fashion without any language model and complex decoder.

The concept of end-to-end all-neural speech recognition has gained much interest in recent study [5–7]. One attractive characteristics of end-to-end all-neural speech recognition system is that it could jointly optimize every components of speech recognition with neural network under a unified framework. Therefore, an important task of end-to-end speech recognition systems is to learn the output dependencies without using separately trained language model. For example, in RNN-Transducer [7–9] and attention based seq2seq model [10–13], the output dependencies are modeled with a separate neural network[1] but trained jointly with a single objective function. A2W system based on CTC [8] is also an example of an end-to-end all-neural speech recognition system where a single neural network models both acoustic and output dependencies.

One of the major difficulties of training A2W system is data sparsity problem. While the study in [5] has alleviated data sparsity problem by using exceptionally large training data up to 125,000 hours, collecting such amount of training data is itself a challenging task in practice. The data sparsity problem of A2W system arises as certain words in the vocabulary does not occur very frequently in the training data. However, as many words share the same structural representation, the data sparsity problem can be alternatively alleviated by exploiting these shared representations. The study in [14,15] is an example where A2W system achieves competitive speech recognition performance with a moderately sized training data by initializing the A2W system with CTC-phone model. It was observed in [15] that the model initialization and regularizations are very important for A2W system when the training data is not exceptionally large. Another approach that could address the data sparsity problem is to have a separate sub-word unit based model (or output layer) to assist predicting these rarely occurred words using detected $\langle unk \rangle$ boundary [16] or decompose these rare words into sub word units in A2W system [15, 17].

In this study, we aim to further enhance the performance of A2W system on moderately sized training data with improved model initialization, training strategy and network architecture. Specifically, we found three techniques that consistently improve existing A2W system on 300 hours Switchboard English speech recognition task. These include:

- Hierarchical-CTC [7, 18] pretraining with phonemes and grapheme as target at different network depth.
- Curriculum training [19] to gradually increase the vocabulary size from 10k to 20k.
- A joint CTC-CE training network.

The rest of the papers is organized as below. In Sec.2, we present the baseline A2W system. In Sec. 3 we describe the model initialization and training strategies explored in this study. In Sec.4 we describe the experimental setup and the results. Finally, we conclude the paper in Sec. 5

2. Baseline Acoustic-to-Word Model

2.1. CTC loss

The CTC loss [20] is used as the objective function for baseline A2W system as in [5, 15]. The CTC loss is defined as negative

[*] Authors contributed equally to this work.

[1] It is called prediction network in RNN-Transducer, and decoder network in attention based seq2seq model.

10.21437/Interspeech.2018-1452

log probability of correct label sequence given input observation for all the data in the training set.

$$\mathcal{L}_{CTC} = -\sum_{(x,l)} lnP(l|x) \quad (1)$$

For calculating CTC loss, the conditional probability of correct label is computed as the accumulated sum of probabilities of all alignment paths belong to given target label sequence.

$$P(l|x) = \sum_{\pi \in l} P(\pi|x). \quad (2)$$

In order to compute the conditional probability more efficiently, the forward-backward algorithm is employed. Another main characteristic of CTC is the use of blank label to allow network to avoid making non-confidential decisions in some frames.

2.2. Network

We use CLDNN [21] as our baseline neural network model architecture. Our input feature is 40-dimensional log filterbank features. A 9x9 frequency-time filter is used for the first convolutional layer, and a 4x4 convolutional filter is used for the second and third convolutional layers. We uses 16 feature maps for each convolutional layers. We reduce the size of final CNN output to 256 with a linear projection layer. We use stride of 3 along the time dimension in the first convolution layer, and 1 for the rest. The output from CNN is then passed to a 5-layer bidirectonal LSTM (BLSTM) where each LSTM has 512 cells. The output of BLSTM is finally passed to a linear projection layer with 320 units followed by the final output layer. As the final output layer of A2W system can be very large depending on the size of vocabulary, a linear projection layer could effectively reduce the training time of A2W system. At the same time, it also improves the performance of A2W system [14, 15].

2.3. Target Vocabulary

In A2W system, the whole words are used as acoustic modeling units. When training data is large enough to ensure sufficient samples for every words in the vocabulary, the entire vocabulary from the training data can be used as target sets for training A2W system [5]. However, when the training data is not large enough, the words that occur less frequently in training data can be mapped to a special label of $\langle unk \rangle$ and these words will not be recognized in A2W system [4, 14, 15]. Specifically, we construct the vocabulary by selecting words appeared more than 5 times into the vocabulary while treating other words as $\langle unk \rangle$. This results in a vocabulary with size of 10,000. This setup is very similar to [14, 15]. Note that all the partial words in the training data is mapped to a special label of [vocalized noise] and ignored during scoring.

2.4. Model Initialization

We initialized our baseline A2W system from pretrained CTC-phone model as in [14,15]. We also found that it is important to have good initialization for the success of A2W system.

2.5. Training Process

We use Adam optimizer [22] for training our model. It is found that use Adam with fixed start learning rate results in poor convergence. The start learning rate of Adam is adjusted every epoch based on the error observed from the development data. Specifically, we start with a learning rate of 5×10^{-3}, and decrease it by a factor of 4 if the average edit distance in the development data doesn't decrease. The training ends when the learning rate is less than 1×10^{-6}. We sorted our training data in an ascending order according to sequence lengths. A batch size of 64 is used and L2 regularization of 0.001 is used. We use a momentum of 0.9.

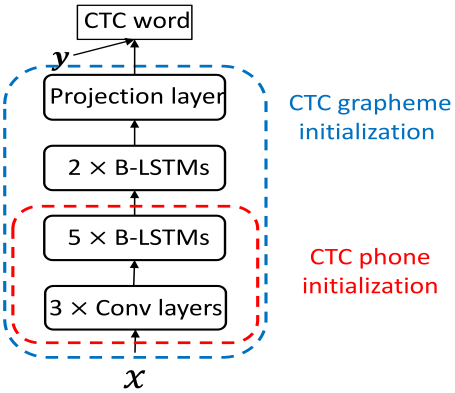

Figure 1: *Illustration of Hierarchical CTC initialization for A2W model. The 3 CNN and 5 BLSTM layers are pretrained with phone based CTC. Additional two layers of BLSTM and a linear projection layer are stacked on top and pretrained with grapheme based CTC.*

The baseline A2W system is trained with Switchboard 300 hour training data and evaluated on Switchboard and CallHome part of Hub5-2000. We compare our baseline A2W system performance with the number reported in [15] in Table 1. Both systems use the same dataset for training and testing, but the models used are not exactly the same. And it can be observed that the WERs in two systems are similar with our A2W baseline performs slightly worse in CallHome part ot test set.

Table 1: *A2W baseline WER on Switchboard and CallHome with 300-hour training data.*

	SWB	CH
A2W [15]	14.6	23.6
A2W (ours)	14.8	25.8

3. Improvements

In this section, we introduce the techniques that improve A2W system in our study.

3.1. Hierarchical CTC pretraining

Model initialization with CTC-phone has shown to be very important in A2W system when the training data is moderately sized [15]. By pretraining the A2W model with CTC-phone, the underlying shared representation of words can be learned in advance. With the same motivation, we investigate Hierarchical CTC pretraining for improved model initialization.

The use of Hierarchical CTC [7, 18] based model initialization has been explored in [14] for A2W system. However, it was reported that the Hierarchical CTC based model initialization could not outperform simple CTC-phone initialization [14]. In our study, we use different implementation of Hierarchical CTC pretraining and found that it brings consistent improvement over the baseline CTC-phone pretraining.

In our implementation of Hierarchical CTC, we first initialize bottom 3 CNN and 5 BLSTM layers with phone based CTC. Then 2 additional BLSTM layers are stacked on top of 5 BLSTM layers. Instead of randomly initialize top two BLSTM directly for A2W system as in [14], we further pretrain the top two BLSTM layers with CTC criterion with grapheme as targets. The motivation of using grapheme as target is to exploit additional structural representations coming from grapheme learning. Figure 1 illustrates hierarchical CTC based A2W model initialization.

3.2. Curriculum Training

When training A2W system, rare words in the training data is much difficult to train than the ones with more occurrences. If we attempt to model all the words in training data simultaneously, it will result in suboptimal performance when the training data is not large enough. The curriculum training [19] has been successfully applied for many difficult optimization problems where a challenging task is addressed by starting from learning easier subtasks.

In this study, we also explore the use of curriculum training to gradually increasing target vocabulary size for A2W system. Specifically, the training is performed in an order of increasing vocabulary size from 10k to 20k. We start from training A2W model for predicting only the most frequently occurred 10k words. During the first curriculum training stage with 10k vocabulary, all utterances with words not belonging to the the selected 10k vocabulary are excluded from training. Therefore, in this first stage of curriculum training, $\langle unk \rangle$ label does not exist. After the training of A2W model with 10k vocabulary converges, the model is then used as starting point to continuously learning to predict vocabulary of larger size (20k) with the rest of words mapped to $\langle unk \rangle$ label. The curriculum training used here first ensures a good convergence point for predicting more frequently occurred words, and the learned representation from the earlier stage could also help predicting the words with less examples.

3.3. Joint CTC-CE Training

Cross Entropy (CE) and CTC is two alternative loss functions for training speech recognition systems. The CE loss is used in conventional speech recognition systems where a fixed alignment between acoustic frames and labels is needed. On the other hand, CTC loss is used in end-to-end speech recognition systems where the loss is computed from all alignment paths belong to given target label sequence.

As both CTC and CE based model are capable of word prediction, we propose to combine CTC and CE for A2W system. The combination of CTC and CE training has been investigated in previous studies. However, in these studies, CE model is used as a way to stabilize the training of CTC system by means of pretraining [4, 23].

In this study, we investigate joint training of CTC and CE for A2W prediction. We investigate two different architectures for joint CTC and CE training. The first approach that we investigate is to combine CTC and CE is through regular multitask learning where the CTC loss in A2W system is replaced with the sum of CTC and CE loss as in Figure2-b. We also investigate a different joint CTC-CE network where the final projection layer in baseline A2W model is extended with two linear transformation with the CE loss updating only one of the two linear layers as in Figure2-c. A potential benefits of having two linear transform layers is to preserve one linear layer dedicated to CTC objective. Specifically, we design our system as in Figure 2-c. The bottom CNN and BLSTM layers are the same as in baseline A2W system. To include the CE loss into CTC based A2W system, we project the output of top linear projection layer with two separate linear projection layers. The output from the second linear layer highlighted with green color in Figure 2-c is directly connected to the final output layer of CE model to receive error signals from CE loss. At the same time, the hidden activations of both projection layers are concatenated to obtain the final output distribution for computing the CTC loss.

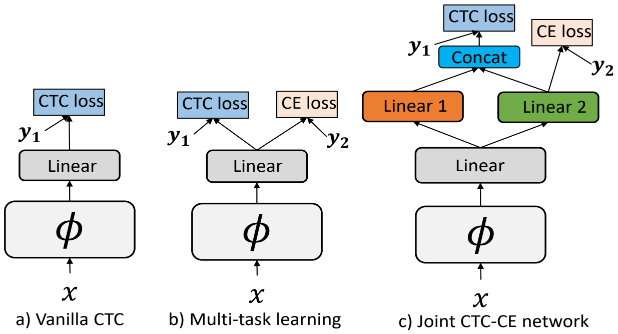

Figure 2: *Representation of (a) Vanila CTC, (b) multi-task learning , and (c) joint CTC-CE training network.The symbol ∅ is function represents CNN and BLSTM layers before the linear projection layer.*

The loss function of multitask learning and joint CTC-CE network training can be both represented as sum of CTC loss and CE loss as in Eq. 3. λ is hyperparameter controls the strength of CE loss.

$$\mathcal{L}_{total} = \mathcal{L}_{CTC} + \lambda \mathcal{L}_{CE} \tag{3}$$

4. Experiments & Results

Our experiments are evaluated on Hub5-2000 test sets using 300 hour Switchboard English speech corpus as training data. Our final model is trained with augmented training data with different speaking rates and volumes as in [3].

4.1. Hierarchical CTC pretraining

We compare the performance of hierarchical CTC pretraining with CTC phone initialization in Table 2. We observe consistent improvements with Hierarchical CTC based model initialization. We could also see that simply increasing depth of layers from 5 to 7 result in only marginal gains in performance.

Table 2: *Comparison of CTC-phone initialization and hierarchical CTC initialization on Hub5-2000 test set.*

Initialization Method	SWB	CH
CTC-Phone (5L-BLSTM)	14.8	25.8
CTC-Phone (7L-BLSTM)	14.7	25.9
Hierarchical CTC (7L-BLSTM)	**14.2**	**24.9**

4.2. Curriculum Training

We compare the performance of training A2W system with 20k vocabulary with regular training strategy and curriculum training in Table 3. It is observed that simply increasing target vocabulary size from 10k to 20k achieves little improvement. But, with curriculum training strategy, using 20k vocabulary as output target indicates a consistent improvement in WER. We found that using A2W model with 20k vocabulary trained with curriculum training could result in 1% absolute reduction in deletion errors. At the same time we notice a slight increase in substitution and insertion errors when using 20k vocabulary.

Table 3: *Comparison of regular training and curriculum training on Switchboard part of Hub5-2000 test set.*

	Regular	Curriculum
A2W (10K vocab)	14.8	-
A2W (20K vocab)	14.7	**14.1**

4.3. Joint CTC-CE Training

We compare the performance of proposed joint CTC-CE network with previous multi-task learning in Table 4. We observe that the simple multi-task learning with CTC and CE loss could not improve the performance of A2W system. On the other hand, the joint CTC-CE network proposed in this study indicates a consistent improvement over baseline A2W system on both Swtichboard and CallHome evaluations. As the joint CTC-CE has two different linear transformation resulting in more parameters than baseline A2W system, we compare it with A2W system with the same network architecture but without CE loss ($\lambda = 0$). And the result indicates that the improvement injoint CTC-CE training is not due to increased parameter size.

4.4. Combination

In previous subsections, we investigated hierarchical CTC pretraining, curriculum training, and joint CTC-CE training separately. In this subsection, we combine three techniques described in previous subsections. We start from baseline A2W system which has a WER of 14.8/25.8% on Switchboard/CallHome part of Hub5-2000 test set. Then we add Hierarchical CTC pretraining for model initialization to reduce the WER to 14.1/24.5%. By increasing the vocabulary size from 10k to 20k with curriculum training, the WER is reduced to 13.5/24.2%. When we combine the curriculum training with join CTC-CE training the WER is further reduced to 13.0/23.4.

Table 4: *Evaluation of joint CTC-CE training on Switchboard and CallHome with 300-hour training data.*

	SWB	CH
A2W	14.8	25.8
A2W (MTL)	15.1	26.3
A2W (Joint CTC-CE, $\lambda = 0$)	14.9	25.4
A2W (Joint CTC-CE, $\lambda = 0.3$)	**14.3**	**25.0**

4.5. Final Model with Data Augmentation

In our final model, we use similar training recipe described in previous sections, but with 3-fold data augmentation using *speed-perturbation* technique described in [3]. The augmented training data is used in each training stage of current A2W system. We also changed our convolutional strides from 3 to 2 along time axis as we found that using stride 2 gives consistently better performance in our later studies. We also added Gaussian weight noise with a standard deviation of 0.0625 [24] and a drop connection rate of 0.1 [25] when experimented using augmented training data. We found that adding Gaussian weight noise and drop connection is important in order to achieve good performance when using speed-perturbed data. Using this model we obtain our final WER of 11.4/20.8 without language model and complex decoder.

Table 5: *Combination of hierarchical CTC model initialization, curriculum training and joint CTC-CE training evaluated on Switchboard and CallHome part of Hub5-2000 test set.*

	SWB	CH
A2W-10K (baseline)	14.8	25.8
+Hierarchical CTC Pretraining	14.1	24.5
+Curriculum Training (10k→20k)	13.4	24.2
+Joint CTC-CE	13.0	23.4
Final model (*speed perturbation*)	11.4	20.8

Finally, we compare the performance of our final model with the WER reported by others using conventional speech recognition system as well end-to-end speech recognition systems in Table 6. Our final model shows competitive performance compared to previous A2W system on the same task as well as other end-to-end speech recognition systems. The gap between A2W system and conventional speech recognition system with 300 hour training data has also been measurably reduced in this work.

Table 6: *Comparing our final model to other systems built on Switchboard 300hrs.*

Model	Output Unit	LM/ Decoder	SWB	CH
DNN+sMBR [26]	CD state	Y	12.6	24.1
BLSTM [27]	CD state	Y	10.8	19.5
BLSTM+LFMMI [3]	CD state	Y	9.6	19.3
Attention Seq2seq [28]	char	Y	25.8	36.0
CTC+CharLM [29]	char	Y	21.4	40.2
Iterated CTC [6]	char	Y	15.1	26.3
CTC [30]	char	Y	14.5	-
A2W [15]	word	N	14.6	23.6
A2W (current)	word	N	**11.4**	**20.8**

5. Conclusions

In this study, we advanced A2W system by using Hierarchical CTC based model initialization, curriculum training, as well as joint CTC-CE training. The A2W system proposed in this work could achieve 11.4/20.8% WER without using language model and complex decoder on Switchboard and Callhome part of Hub5-2000 evaluation set with SWB-300Hr training data. Future study includes the evaluation of current A2W system with larger training set by including 2000 hours of Fisher dataset.

6. References

[1] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *CoRR*, vol. abs/1610.05256, 2016. [Online]. Available: http://arxiv.org/abs/1610.05256

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," *CoRR*, vol. abs/1703.02136, 2017. [Online]. Available: http://arxiv.org/abs/1703.02136

[3] 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 *Interspeech*, 2016, pp. 2751–2755.

[4] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *arXiv preprint arXiv:1507.06947*, 2015.

[5] H. Soltau, H. Liao, and H. Sak, "Neural Speech Recognizer: Acoustic-to-Word LSTM Model for Large Vocabulary Speech Recognition," *ArXiv e-prints*, Oct. 2016.

[6] G. Zweig, C. Yu, J. Droppo, and A. Stolcke, "Advances in all-neural speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4805–4809.

[7] K. Rao, H. Sak, and R. Prabhavalkar, "Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Dec 2017, pp. 193–199.

[8] A. Graves, "Sequence Transduction with Recurrent Neural Networks," *ArXiv e-prints*, Nov. 2012.

[9] E. Battenberg, J. Chen, R. Child, A. Coates, Y. Gaur, Y. Li, H. Liu, S. Satheesh, D. Seetapun, A. Sriram, and Z. Zhu, "Exploring neural transducers for end-to-end speech recognition," *CoRR*, vol. abs/1707.07413, 2017. [Online]. Available: http://arxiv.org/abs/1707.07413

[10] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *CoRR*, vol. abs/1506.07503, 2015. [Online]. Available: http://arxiv.org/abs/1506.07503

[11] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4960–4964.

[12] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina, N. Jaitly, B. Li, J. Chorowski, and M. Bacchiani, "State-of-the-art Speech Recognition With Sequence-to-Sequence Models," *ArXiv e-prints*, Dec. 2017.

[13] T. N. Sainath, C. Chiu, R. Prabhavalkar, A. Kannan, Y. Wu, P. Nguyen, and Z. Chen, "Improving the performance of online neural transducer models," *CoRR*, vol. abs/1712.01807, 2017. [Online]. Available: http://arxiv.org/abs/1712.01807

[14] K. Audhkhasi, B. Ramabhadran, G. Saon, M. Picheny, and D. Nahamoo, "Direct acoustics-to-word models for english conversational speech recognition," in *Proc. Interspeech 2017*, 2017, pp. 959–963. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2017-546

[15] K. Audhkhasi, B. Kingsbury, B. Ramabhadran, G. Saon, and M. Picheny, "Building competitive direct acoustics-to-word models for English conversational speech recognition," *ArXiv e-prints*, Dec. 2017.

[16] J. Li, G. Ye, R. Zhao, J. Droppo, and Y. Gong, "Acoustic-To-Word Model Without OOV," *ArXiv e-prints*, Nov. 2017.

[17] J. Li, G. Ye, A. Das, R. Zhao, and Y. Gong, "Advancing acoustic-to-word CTC model," *arXiv preprint arXiv:1803.05566*, 2018.

[18] K. Rao and H. Sak, "Multi-accent speech recognition with hierarchical grapheme based models," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4815–4819.

[19] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 41–48.

[20] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06. New York, NY, USA: ACM, 2006, pp. 369–376. [Online]. Available: http://doi.acm.org/10.1145/1143844.1143891

[21] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4580–4584.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[23] H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, "Learning acoustic frame labeling for speech recognition with recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4280–4284.

[24] K.-C. Jim, C. L. Giles, and B. G. Horne, "An analysis of noise in recurrent neural networks: convergence and generalization," *IEEE Transactions on neural networks*, vol. 7, no. 6, pp. 1424–1438, 1996.

[25] L. Wan, M. Zeiler, S. Zhang, Y. L. Cun, and R. Fergus, "Regularization of neural networks using dropconnect," in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28, no. 3. Atlanta, Georgia, USA: PMLR, 17–19 Jun 2013, pp. 1058–1066.

[26] K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks."

[27] G. Saon, T. Sercu, S. J. Rennie, and H. J. Kuo, "The IBM 2016 english conversational telephone speech recognition system," *CoRR*, vol. abs/1604.08242, 2016. [Online]. Available: http://arxiv.org/abs/1604.08242

[28] L. Lu, X. Zhang, and S. Renais, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 5060–5064.

[29] A. Maas, Z. Xie, D. Jurafsky, and A. Ng, "Lexicon-free conversational speech recognition with neural networks," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 345–354.

[30] Y. Miao, M. Gowayyed, X. Na, T. Ko, F. Metze, and A. Waibel, "An empirical exploration of CTC acoustic models," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 2623–2627.