INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

1-D Row-Convolution LSTM: Fast Streaming ASR at Accuracy Parity with LC-BLSTM

Kshitiz Kumar, Chaojun Liu, Yifan Gong, Jian Wu

Microsoft Corporation, Redmond, WA
{kshitiz.kumar, chaojun.liu, ygong, jianwu}@microsoft.com

Abstract

In this work we develop a simple, efficient, and compact automatic speech recognition (ASR) model based on purely 1-dimensional row convolution (RC) operation. We refer to our proposed model as 1-dim row-convolution LSTM (RC-LSTM), where we embed limited future information to standard UniLSTMs in 1-dim RC operation. We target fast streaming ASR solutions and establish ASR accuracy parity with latency-control bidirectional-LSTM (LC-BLSTM). We develop an application of future information at ASR features and hidden layer stages. We study connections with related techniques, analyze trade-offs and recommend uniform future lookahead to all hidden layers. We argue that our architecture implicitly factorizes training into orthogonal time and "frequency" dimensions for an effective learning on large scale tasks. We conduct a series of experiments on medium scale with 6k hrs of English corpus, as well as, large scale with 60k hrs training. We demonstrate our findings across unified ASR tasks. Compared to UniLSTM model, RC-LSTM achieved 16% relative reduction in word error rate (WER). RC-LSTM also achieved accuracy parity with LC-BLSTM on large scale tasks at significantly lower latency and computational cost.

Index Terms: Row Convolution, Speech Recognition, LSTM, LC-BLSTM, BLSTM, Robust ASR

1. Introduction

Deep learning techniques have enabled new speech devices for ever-growing applications. We strive for the highest accuracy under reasonable constraints in computational cost and user-perceived latency metrics. Over the years, speech research has delivered significant advances for ASR. The long-short term memory (LSTM) models in [1, 2, 3], and CLDNN in [4] have improved DNN-based techniques in [5, 6, 7, 8].

Our objective in this work is to develop new acoustic model (AM) architectures to better meet ASR metrics in accuracy, latency and computational cost. We have already noted a progression of model structures in GRU [9], LSTM [10], BLSTM [11, 12], LC-BLSTM [13, 14], cLTLSTM [15], and F&S AM [16] with respective trade-offs in aforementioned ASR metrics. BLSTMs provide substantial gains over LSTMs. LC-BLSTMs constrain latency, retain a close accuracy parity with BLSTM models, thus provide an approach to high accuracy streaming ASR solution.

Our work borrows a key learning from BLSTM that future information is crucial for ASR. We develop a 1-dim row convolution structure to embed future information to UniLSTM models for accuracy parity with LC-BLSTM. We have several other motivations, (a) UniLSTMs are significantly simpler with computational cost being 50% of that for LC-BLSTMs, (b) LC-BLSTMs process the data in large batches that introduces discontinuities in the intermediate ASR hypotheses and impacts user-perceived latency.

The significance of future information has also been noted in recent studies. [17] applied future information for LSTM in the context of developing mixed frame rate models with TDNN, [15] applied future context to LT-LSTM models, [18] developed row convolution to the top hidden layer in CTC training framework, and [19] applied it for GRU models. Among recent studies, our work is more closely related to [18]. The key contributions of our work include extending prior work to a simple and effective 1-dim RC operation in UniLSTM framework. We also develop a self-attention initialization for the row convolution parameters and report that to be critical for convergence in models with long lookaheads. We investigate the role of future information at ASR features and hidden layers, and report related trade-offs. We also describe that our architecture implicitly factorizes training into orthogonal dimensions in time and "frequency". This efficiently uses the network parameters and leads to effective learning on large tasks. Finally we achieve accuracy parity with LC-BLSTMs on a large training with over 60k hrs of data.

The rest of this work is organized in following. We briefly discuss our motivation for seeking new model structure in Sec. 2. We describe future information sources, and develop RC-LSTM with respect to 1-dim row convolution operation in Sec. 3. We present our experiments and results in Sec. 4. We conclude this work in Sec. 5.

2. ASR metrics and motivation

ASR consumers have following 3 key expectations from an ASR system: high accuracy, minimal perceived latency, and lower cost. ASR accuracy is a paramount metric; we build systems with the best possible accuracy that's also robust to acoustic and environmental factors [20]. Latency too is a significant ASR metric. Consumers expect a fast streaming ASR response for intermediate as well as final recognition results. Lower ASR computational cost too is critical metric for scalable ASR systems. We next benchmark few standard AMs along above ASR metrics. We identify scenarios with significant gaps, and subsequently motivate RC-LSTM model to address those gaps.

2.1. Review of standard acoustic model structures

We review few relevant state-of-the-art AM structures, and benchmark the models along the ASR metrics in Table 1.

2.1.1. UniLSTM

Deep long-short term memory (LSTM) recurrent cells (LSTM-RNN) [1] are among widely used AM structures. We have made significant advances over the years and built large scale ASR systems with LSTM-RNN models. The LSTM-RNN models constitute a few layers of LSTM cells along with a Softmax

http://dx.doi.org/10.21437/Interspeech.2020-2894

Table 1: *Models vs. ASR metrics. ✓meets expectations. '+' and '-' are below expectations, though '+' is relatively better.*

Models	Accuracy	Latency	Compute Cost
LSTM	-	✓	✓
BLSTM	✓	-	-
LC-BLSTM	✓	+	-
RC-LSTM	✓	✓	✓

layer at top. The context-dependent tied triphones constitute the acoustic states, and the model predicts a distribution of the acoustic states for input speech features. As in Table 1, UniLSTMs have the simplest structure. UniLSTMs do not have lookahead requirements and lead to the smallest latency operations. UniLSTMs satisfy most of the ASR metrics except that UniLSTMs have weaker accuracy than BLSTMs. In this work we also use the acronym LSTM for UniLSTM.

2.1.2. BLSTM and LC-BLSTM

The Bidirectional LSTMs (BLSTM) incorporate a sequence of forward as well as a backward LSTM cells. That aggregates past and future information, and shows accuracy gains over LSTMs. Furthermore, in comparison to LSTMs, BLSTMs are 2x in the model size as well as computation requirements. We benchmark these metrics in Table 1.

LC-BLSTM improves latency aspects in BLSTM while retaining ASR accuracy. LC-BLSTMs achieve that by restricting the forward and backward computation to batches of data. However processing the data in larger batches breaks natural continuity of intermediate ASR hypotheses. Above significantly impacts the user-perceived latency for intermediate as well as final ASR hypotheses. The LC-BLSTM computation with overlapping data batches also increases the computational cost, as summarized in Table 1.

3. Row convolution LSTM (RC-LSTM)

The physical movement of articulatory organs in human speech introduce continuity and dependency of the current speech signal on past and future speech. This makes speech a very contextual signal. BLSTM models benefit from the contextual nature by using past and future context to predict the current acoustic states. We infer that encoding future information is a key attribute of BLSTM models, and focus on leveraging future information in the context of much simpler LSTM models.

3.1. RC-LSTM future information sources

We discuss potential information sources for future information in RC-LSTM, along with associated trade-offs.

(A) ASR Feature Level - ASR features provide a starting framework to ingest future information. We can stack the current frame with few future frames and train ASR models on the stacked features. This doesn't require changes to the core LSTM evaluation, as all changes are isolated to the input stage. The approach is also advantageous in data communication over networks, where typically data is sent in packets of say 100 ms, so ASR system can readily access a few future frames without incurring hard latency.

(B) Hidden LSTM Outputs - Hidden layers provide a good alternative [21], we can embed future information to all or particular hidden layers. One approach can be to splice current and few future hidden activations, and feed this to the next hidden layer, as in [17]. However splicing significantly increases the model size due to new parameters. In this work we pursue a purely 1-dim convolution operation to minimize new parameters and seek compact models that can also be used for adaptation [22] purposes.

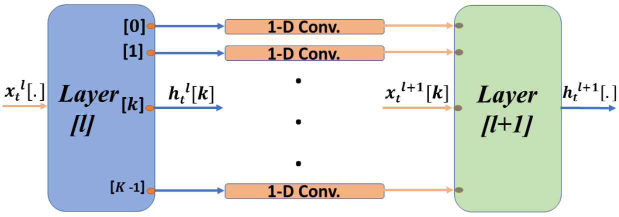

Figure 1: *1-dim RC-LSTM model. k^{th} input for the next layer, $l+1$, depends only on k^{th} output of the previous layer, l. This information bottleneck implicitly factorizes the network along time and "frequency" dimensions and reduces new parameters from typically $O(N^2)$ to $O(N)$.*

(C) Top layer activation and other combinations - Using top layer activation in the context of CTC has been presented in [18]. We can also extend the approach by jointly leveraging future information to different combinations across feature, hidden layer and top layer with layer-specific lookaheads.

3.2. Row convolution LSTM (RC-LSTM)

We motivated UniLSTMs in Sec. 2.1.1 and develop RC-LSTM on UniLSTM foundations to optimize ASR metrics in Table 1. Our baseline UniLSTM cells include special connections, *e.g.* peephole and output projection, so for completeness and later reference we describe our UniLSTM cell following [15] in:

$$\mathbf{i}^l_t = \sigma(\mathbf{W}^l_{ix}\mathbf{x}^l_t + \mathbf{W}^l_{ih}\mathbf{h}^l_{t-1} + \mathbf{p}^l_i \otimes \mathbf{c}^l_{t-1} + \mathbf{b}^l_i) \quad (1)$$

$$\mathbf{f}^l_t = \sigma(\mathbf{W}^l_{fx}\mathbf{x}^l_t + \mathbf{W}^l_{fh}\mathbf{h}^l_{t-1} + \mathbf{p}^l_f \otimes \mathbf{c}^l_{t-1} + \mathbf{b}^l_f) \quad (2)$$

$$\mathbf{c}^l_t = \mathbf{f}^l_t \otimes \mathbf{c}^l_{t-1} + \mathbf{i}^l_t \otimes \phi(\mathbf{W}^l_{cx}\mathbf{x}^l_t + \mathbf{W}^l_{ch}\mathbf{h}^l_{t-1} + \mathbf{b}^l_c) \quad (3)$$

$$\mathbf{o}^l_t = \sigma(\mathbf{W}^l_{ox}\mathbf{x}^l_t + \mathbf{W}^l_{oh}\mathbf{h}^l_{t-1} + \mathbf{p}^l_o \otimes \mathbf{c}^l_t + \mathbf{b}^l_o) \quad (4)$$

$$\mathbf{g}^l_t = \mathbf{o}^l_t \otimes \phi(\mathbf{c}^l_t) \quad (5)$$

$$\mathbf{h}^l_t = \mathbf{W}^l_{hg}\mathbf{g}^l_t \quad (6)$$

The core input to the network is a sequence of ASR features $\mathbf{s}_t$ at time t. The network has L LSTM layers, $\mathbf{x}^l_t$ is input vector for the l-th layer, where $\mathbf{x}^1_t = \mathbf{s}_t$. The output of the l-th layer is $\mathbf{h}^l_t$. The network is parametrized by a set of weight matrices in $\mathbf{W}^l_{.x}$, $\mathbf{W}^l_{.h}$, and $\mathbf{W}^l_{hg}$. The activation vectors are $\mathbf{i}^l_t$, $\mathbf{o}^l_t$, $\mathbf{f}^l_t$, $\mathbf{c}^l_t$, the bias vectors are $\mathbf{b}^l_.$, and, $\mathbf{p}^l_i$, $\mathbf{p}^l_o$, $\mathbf{p}^l_f$ constitute peephole connections. We apply a projection matrix $\mathbf{W}^l_{hg}$ for lower-dimensional outputs $\mathbf{h}^l_t$. We refer to [15] for additional details.

Next we illustrate the core innovation in RC-LSTM with respect to Fig. 1. We build on an application of row convolution to the top hidden layer in [18], and propose to enrich the model capacity by uniformly distributing the total available lookahead to all hidden layers in the network. We also extend [18] along new dimensions. We argue that our proposed 1-dim row convolution LSTM in Fig. 1 provides a simple and yet effective framework to ingest future information in UniLSTMs. We document our

Table 2: *en-US Training data.*

Tasks	Training [hrs]
Medium scale	6000
Large scale	60000

1-dim RC-LSTM with respect to Fig. 1 in below:

$$\mathbf{x}_t^{l+1}[k] = \sum_{\tau=0}^{\mathcal{T}} \alpha_\tau^l[k]\mathbf{h}_{t+\tau}^l[k], \quad l > 0, k \in [0, \cdots, K) \qquad (7)$$

where, k indicates a particular unit in K-dim output $\mathbf{h}_t^l$. $\alpha_\tau^l[k]$ constitutes the 1-dim convolution parameters. We reiterate that the k^{th} input to the next layer $\mathbf{x}_t^{l+1}[k]$ depends on only the k^{th} output in $\mathbf{h}_t^l[k]$ and its corresponding $\mathcal{T}$ future states. A larger $\mathcal{T}$ embeds greater future information in the network and implies larger user-perceived latency. We also document a few attributes and significant implications of our work.

1. Smaller network due to fewer parameters: For a typical row convolution operation in [18] or TDNN in [17], the k^{th} input to the next layer depends on all outputs from previous layer. That constitutes $O(N^2)$ parameters. Whereas, our proposed 1-dim RC model requires only $O(N)$ parameters and leads to compact models.
2. Simplicity in network: We know that simplicity is ultimate sophistication, and build simple 1-dim row convolution operation on top of UniLSTMs. Furthermore in contrast to window selection parameters in TDNN, we uniformly apply the work to all hidden layers with identical $\mathcal{T}$ parameter, and achieve parity with LC-BLSTM. Our design requires no investments in parameter selection or related tuning. Furthermore our design doesn't require additional work from layer trajectory in [15] and greatly simplifies the model.
3. Time and "frequency" factorization in network: We argue that 1-dim RC operation creates an information bottleneck by implicitly factorizing time and "frequency" information. RC operation distills the information along "time" dimension without access to "frequency" dimension, *i.e.* other hidden layer outputs. The core LSTM layers learn from "frequency" information, without access to time dimension. We believe above orthogonal attributes are desirable and lead to effective training for a large scale deep learning models.

3.2.1. *Self-attention initialization for RC-LSTM α_0^l parameters*

It's customary to initialize all the LSTM parameters with a certain distribution in a range, say $[-0.05, 0.05]$. In our work, we embed a strong self-attention attribute in the RC-LSTM model by initializing the α_0 parameters in (7) as $\alpha_0^l[k] = 1 \ \forall \ k, l$. This initialization is naturally motivated as RC-LSTM reduces to LSTM with $\mathcal{T} = 0$ and $\alpha_0^l[k] = 1 \ \forall \ k, l$. We report above initialization to be critical for deep lookahead models.

4. Experiments and results

We conducted experiments on a large vocabulary en-US task over medium and large scale tasks noted in Table 2. The data is anonymized with personal identifiable information removed. The data reflects a great variety of our usage scenarios including close-talk, far-talk, natural conversation, command-and-control, dictation, voice search, and call center tasks etc. Our baseline is a standard 6-layers unidirectional LSTM model with cross-entropy (CE) [5] training criterion. The LSTM cells have 1024 memory units with 9k output acoustic states. Our model evaluation applies frame-skipping. Our core ASR features are 80-dim log-Mel features, with feature processing time window as 25-ms with 10-ms window shift. We use a 5-gram language model with vocabulary of over 1M words.

Table 3: *WER [%] performance for RC models. System (A) uses 80-dim features, rest all use 160-dim. We also report WERR [%] over System (B)*

Models	WER [%]	WERR [%]	Latency [ms]
80-dim Baseline (A)	14.4	-1.4	0
160-dim Baseline (B)	14.2	-	0
(B) + 6 future features (C)	13.4	5.6	120
RC1	12.9	9.2	120
RC2	12.3	13.4	240
RC3	12.1	14.8	360
RC4	11.9	16.2	480

In Table 3 we report results on a medium scale training task with over 6000 hrs of data. The evaluation consists of near-field task with over 30 hrs of speech across devices. Our goal is to obtain an initial assessment on RC-LSTM model, study the impact of future lookahead parameter, and summarize the role of information sources we discussed in sec. 3.1. We plan to arrive at a final configuration for eventual comparison with LC-BLSTM on large scale training tasks. In Table 3 we report 14.4% WER for our baseline system (A) that uses 80-dim features. We also report a stronger baseline (B) trained on 160-dim features [20] by stacking 80-dim $\mathbf{s}_t$ to the preceding feature $\mathbf{s}_{t-1}$ that was skipped in our frame-skipping model evaluation. The 160-dim features improved Baseline WER to 14.2%. We base the rest of our work on 160-dim features.

In sec. 3.1 we noted embedding future information at ASR features stage. To evaluate that we built on Baseline (B) by additionally stacking 6 future feature frames. Note that each future frame incurs 20 ms due to frame skipping, so stacking 6 frames results in a latency of 120 ms. This system improved WER to 13.4%. Clearly incorporating future information at ASR feature stage is an effective approach. In sec. 3.1 we also motivated exploring future information at hidden layer stages. Consequently we trained RC-LSTM models with different uniform lookaheads at each of the hidden layers. We reiterate that all the models in this work is based on 6-layers UniLSTM with 1024 cells projected to 512 output units. In Table 3 "RC1" refers to RC-LSTM with $\mathcal{T} = 1$ for each of the 6 hidden layers. A direct comparison for RC1 can be done with system (C), as both systems have a lookahead of 6 future frames and incur 120-ms latency. There RC1 model uniformly distributes the lookahead to all hidden layers with $\mathcal{T} = 1$ for every layer, whereas, system (C) uses all 6 lookaheads at the ASR feature stage. RC1 improves WER to 12.9% and verifies our hypothesis that incorporating lookahead at hidden layers leads to stronger learning. This can also be explained in the context that a lookahead at hidden layers incorporates progressively richer information that can be helpful for final training.

We expand our study by increasing lookahead $\mathcal{T} = 1$ in RC1 to larger lookaheads. Following the RC1 notation, RC2,

Table 4: *WER [%] performance for RC models with 120-ms latency.*

Models	RC-TopLayer	RC-Upper3Layers	RC1
WER [%]	13.3	13.3	12.9

Table 5: *Large scale evaluation of RC-LSTM (RC4) and LC-BLSTM.*

Models	LC-BLSTM	RC-LSTM
WER [%]	10.8	10.9

RC3, and RC4 in Table 3 respectively refer to RC-LSTM model with $\mathcal{T} = 2$, $\mathcal{T} = 3$, and $\mathcal{T} = 4$, for all 6 hidden layers. We report continued gains with larger lookaheads and RC4 demonstrates a very strong 16.2% WER relative reduction (WERR) over our baseline (B). We also expanded system (C) by incorporating additional future frames but found that WER quickly saturated.

We continued our developments by training a few lookahead combinations in Table 4. There all models have an identical overall lookahead of 6 frames and consequently a latency of 120 ms. "RC-TopLayer" follows [18] and develops an RC structure with 6 future frames at the top LSTM layer, *i.e.* $\mathcal{T} = 6$ for $l = 6$. In "RC-Upper3Layers" we embed identical future information to each of the upper 3 hidden layers, *i.e.* $\mathcal{T} = 2$ for $l \in [4, 5, 6]$. Table 4 demonstrates an additional evidence for identical lookahead in RC1. We also tried a few other lookahead combinations for 120-ms latency and concluded RC1 as the best combination. We have also tried few lookahead configurations in the context of other RC models in Table 3, and found that our uniform lookahead recommendation generalized to those models as well. Our results lead to significant modeling simplification and don't require investments in lookahead parameter selection.

4.1. Evaluation on large scale task

Finally in Table 5, we compare LC-BLSTM and RC-LSTM models on our large scale training with over 60k hrs Unified English data, and evaluate on over 200 hrs task. The LC-BLSTM model size is 290-MB, it consists of 6 layers with 960 cells in the forward and backward LSTM cells, that's projected to 480 nodes. We designed a comparably sized RC-LSTM model with 6 layers and 1600 cells in each layer that's projected to 800 nodes. The results conclude that RC-LSTMs have a close accuracy parity with LC-LBLSTMs (10.8% vs. 10.9% for RC-LSTM). We have noted that LC-BLSTM computation occurs in longer data batches with overlapping windows that entails duplicate computation. The above RC-LSTM incurs an algorithmic latency of 480-ms, whereas, LC-BLSTM has an average latency of 600 ms, that demonstrates 20% relative reduction in latency with RC-LSTM. In practice the latency reduction is even larger as RC-LSTMs computational cost is only 50% of that for LC-BLSTM, so RC-LSTMs are much quicker to evaluate. We also document that the 1st ASR response from LC-BLSTM incurs a latency of 800-ms, whereas, that's still 480-ms for RC-LSTM, thus an absolute saving of over 320-ms in user-perceived latency for the 1st ASR response.

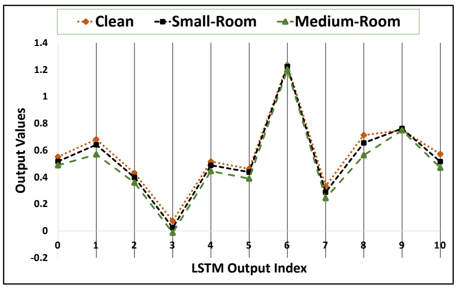

Figure 2: *Adaptation for RC-LSTM α_2^6 parameters in reverberant environments.*

4.2. RC-LSTM model adaptation in reverberant conditions

We briefly present a study on RC-LSTM adaptation in reverberant environments. The convolution parameters in RC structure present a very compact set of parameters for developing adaptation techniques. In Fig. 2 we plot a few α_2^6 parameters for our RC2 model, along with corresponding parameters from adapting over few digitally synthesized utterances for Small and Medium rooms. We find α_2^6 parameters very indicative of room reverberation. The α_2^6 values are gradually lower for rooms with higher reverberation. The average of $\alpha_2^6[.]$ decreased from 0.51 for close-talk (clean) condition to 0.49 for Small-Room, and 0.45 for Medium-Room. Above likely reflects the weaker confidence of the model on future states in reverberant environments. We have also noted similar trends for other α parameters. This study further speaks to the compactness attributes in RC-LSTM and enables a quick speaker or environment adaptation in limited data scenarios.

4.3. Discussion

We designed an experiment to test the implications of time and "frequency" factorization in RC-LSTM. We expanded the RC operation for $\mathbf{x}_t^{l+1}[k]$ in (7) to include future information from neighboring k-indexes but that didn't improve our proposed RC-LSTM models. Above further shows that time and "frequency" decoupling leads to compact and efficient learning. In another study we extended (7) to a bidirectional version by embedding past and future information. That treatment for RC4 in Table 3 resulted in 1% WERR.

The RC-LSTMs increase the corresponding LSTM model size by less than 1% relative, and yet demonstrate large gains. We believe the simplicity and compactness of RC operation will have significant applications for smaller models like GRU [9]. We are also following up this work for end-to-end ASR models as well as applications beyond ASR. We also acknowledge Jinyu Li for valuable insights in this work.

5. Conclusion

We presented RC-LSTM acoustic model structure, we demonstrated big gains over UniLSTM, and accuracy parity with LC-BLSTM on large scale tasks. We also developed a self-initialization attribute and conducted experiments to evaluate the future information across features and hidden layers, and recommended equal lookahead at all hidden layers. Overall we report close accuracy parity with state-of-the-art LC-BLSTMs, 20-40% relative reduction in user-perceived latency, and 50% reduction in computational cost.

6. References

[1] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[2] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proc. Interspeech*, 2014, pp. 338–342.

[3] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," in *Proc. Interspeech*, 2015.

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

[5] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[6] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7398–7402.

[7] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. L. Seltzer, G. Zweig, X. He, J. D. Williams, Y. Gong, and A. Acero, "Recent advances in deep learning for speech research at microsoft," *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 8604–8608, 2013.

[8] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing,*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[9] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[10] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Fifteenth annual conference of the international speech communication association*, 2014.

[11] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[12] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *International Conference on Artificial Neural Networks*. Springer, 2005, pp. 799–804.

[13] K. Chen and Q. Huo, "Training deep bidirectional lstm acoustic model for lvcsr by a context-sensitive-chunk bptt approach," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 7, pp. 1185–1193, 2016.

[14] S. Xue and Z. Yan, "Improving latency-controlled blstm acoustic models for online speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5340–5344.

[15] J. Li, L. Lu, C. Liu, and Y. Gong, "Improving layer trajectory lstm with future context frames," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6550–6554.

[16] K. Kumar, E. Stoimenov, H. Khalil, and J. Wu, "Fast and slow acoustic model," in *Interspeech*, 2020.

[17] V. Peddinti, Y. Wang, D. Povey, and S. Khudanpur, "Low latency acoustic modeling using temporal convolution and lstms," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 373–377, 2017.

[18] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International conference on machine learning*, 2016, pp. 173–182.

[19] J. Li, X. Wang, Y. Zhao, and Y. Li, "Gated recurrent unit based acoustic modeling with future context," *arXiv preprint arXiv:1805.07024*, 2018.

[20] K. Kumar, B. Ren, Y. Gong, and J. Wu, "Bandpass noise generation and augmentation for unified ASR," in *Interspeech*, 2020.

[21] K. Kumar, C. Liu, and Y. Gong, "Non-negative intermediate-layer DNN adaptation for a 10-KB speaker adaptation profile," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5285–5289.

[22] C. Liu, Y. Wang, K. Kumar, and Y. Gong, "Investigations on speaker adaptation of LSTM RNN models for speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5020–5024.