INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Residual LSTM: Design of a Deep Recurrent Architecture for Distant Speech Recognition

Jaeyoung Kim[1], *Mostafa El-Khamy*[1], *Jungwon Lee*[1]

[1]Samsung Semiconductor, Inc.
4921 Directors Place, San Diego, CA, USA

jaey1.kim@samsung.com, mostafa.e@samsung.com, jungwon2.lee@samsung.com

Abstract

In this paper, a novel architecture for a deep recurrent neural network, residual LSTM is introduced. A plain LSTM has an internal memory cell that can learn long term dependencies of sequential data. It also provides a temporal shortcut path to avoid vanishing or exploding gradients in the temporal domain. The residual LSTM provides an additional spatial shortcut path from lower layers for efficient training of deep networks with multiple LSTM layers. Compared with the previous work, highway LSTM, residual LSTM separates a spatial shortcut path with temporal one by using output layers, which can help to avoid a conflict between spatial and temporal-domain gradient flows. Furthermore, residual LSTM reuses the output projection matrix and the output gate of LSTM to control the spatial information flow instead of additional gate networks, which effectively reduces more than 10% of network parameters. An experiment for distant speech recognition on the AMI SDM corpus shows that 10-layer plain and highway LSTM networks presented 13.7% and 6.2% increase in WER over 3-layer baselines, respectively. On the contrary, 10-layer residual LSTM networks provided the lowest WER 41.0%, which corresponds to 3.3% and 2.8% WER reduction over plain and highway LSTM networks, respectively.

Index Terms: ASR, LSTM, GMM, RNN, CNN

1. Introduction

Over the past years, the emergence of deep neural networks has fundamentally changed the design of automatic speech recognition (ASR). Neural network-based acoustic models presented significant performance improvement over the prior state-of-the-art Gaussian mixture model (GMM) [1, 2, 3, 4, 5]. Advanced neural network-based architectures further improved ASR performance. For example, convolutional neural networks (CNN) which has been huge success in image classification and detection were effective to reduce environmental and speaker variability in acoustic features [6, 7, 8, 9, 10]. Recurrent neural networks (RNN) were successfully applied to learn long term dependencies of sequential data [11, 12, 13, 14].

The recent success of a neural network based architecture mainly comes from its deep architecture [15, 16]. However, training a deep neural network is a difficult problem due to vanishing or exploding gradients. Furthermore, increasing depth in recurrent architectures such as gated recurrent unit (GRU) and long short-term memory (LSTM) is significantly more difficult because they already have a deep architecture in the temporal domain.

There have been two successful architectures for a deep feed-forward neural network: residual network and highway network. Residual network [17] was successfully applied to train more than 100 convolutional layers for image classification and detection. The key insight in the residual network is to provide a shortcut path between layers that can be used for an additional gradient path. Highway network [18] is an another way of implementing a shortcut path in a feed-forward neural network. [18] presented successful MNIST training results with 100 layers.

Shortcut paths have also been investigated for RNN and LSTM networks. The maximum entropy RNN (ME-RNN) model [19] has direct connections between the input and output layers of an RNN layer. Although limited to RNN networks with a single hidden layer, the perplexity improved by training the direct connections as part of the whole network. Highway LSTM [20, 21] presented a multi-layer extension of an advanced RNN architecture, LSTM [22]. LSTM has internal memory cells that provide shortcut gradient paths in the temporal direction. Highway LSTM reused them for a highway shortcut in the spatial domain. It also introduced new gate networks to control highway paths from the prior layer memory cells. [20] presented highway LSTM for far-field speech recognition and showed improvement over plain LSTM. However, [20] also showed that highway LSTM degraded with increasing depth.

In this paper, a novel highway architecture, residual LSTM is introduced. The key insights of residual LSTM are summarized as below.

- Highway connection between output layers instead of internal memory cells: LSTM internal memory cells are used to deal with gradient issues in the temporal domain. Reusing it again for the spatial domain could make it more difficult to train a network in both temporal and spatial domains. The proposed residual LSTM network uses an output layer for the spatial shortcut connection instead of an internal memory cell, which can less interfere with a temporal gradient flow.
- Each output layer at the residual LSTM network learns residual mapping not learnable from highway path. Therefore, each new layer does not need to waste time or resource to generate similar outputs from prior layers.
- Residual LSTM reuses an LSTM projection matrix as a gate network. For an usual LSTM network size, more than 10% learnable parameters can be saved from residual LSTM over highway LSTM.

The experimental result on the AMI SDM corpus [23] showed 10-layer plain and highway LSTMs had severe degradation from increased depth: 13.7% and 6.2% increase in WER over 3-layer baselines, respectively. On the contrary, a 10-layer residual LSTM presented the lowest WER 41.0%, which outperformed the best models of plain and highway LSTMs.

http://dx.doi.org/10.21437/Interspeech.2017-477

2. Revisiting Highway Networks

In this section, we give a brief review of LSTM and three existing highway architectures.

2.1. Residual Network

A residual network [17] provides an identity mapping by shortcut paths. Since the identity mapping is always on, function output only needs to learn residual mapping. Formulation of this relation can be expressed as:

$$y = F(x; W) + x \tag{1}$$

y is an output layer, x is an input layer and $F(x; W)$ is a function with an internal parameter W. Without a shortcut path, $F(x; W)$ should represent y from input x, but with an identity mapping x, $F(x; W)$ only needs to learn residual mapping, $y - x$. As layers are stacked up, if no new residual mapping is needed, a network can bypass identity mappings without training, which could greatly simplify training of a deep network.

2.2. Highway Network

A highway network [18] provides another way of implementing a shortcut path for a deep neural-network. Layer output $H(x; W_h)$ is multiplied by a transform gate $T(x; W_T)$ and before going into the next layer, a highway path $x \cdot (1 - T(x; W_T))$ is added. Formulation of a highway network can be summarized as:

$$y = H(x; W_h) \cdot T(x; W_T) + x \cdot (1 - T(x; W_T)) \tag{2}$$

A transform gate is defined as:

$$T(x; W_T) = \sigma(W_T x + b_T) \tag{3}$$

Unlike a residual network, a highway path of a highway network is not always turned on. For example, a highway network can ignore a highway path if $T(x; W_T) = 1$, or bypass a output layer when $T(x; W_T) = 0$.

2.3. Long Short-Term Memory (LSTM)

Long short-term memory (LSTM) [22] was proposed to resolve vanishing or exploding gradients for a recurrent neural network. LSTM has an internal memory cell that is controlled by forget and input gate networks. A forget gate in an LSTM layer determines how much of prior memory value should be passed into the next time step. Similarly, an input gate scales new input to memory cells. Depending on the states of both gates, LSTM can represent long-term or short-term dependency of sequential data. The LSTM formulation is as follows:

$$i_t^l = \sigma(W_{xi}^l x_t^l + W_{hi}^l h_{t-1}^l + w_{ci}^l c_{t-1}^l + b_i^l) \tag{4}$$

$$f_t^l = \sigma(W_{xf}^l x_t^l + W_{hf}^l h_{t-1}^l + w_{cf}^l c_{t-1}^l + b_f^l) \tag{5}$$

$$c_t^l = f_t^l \cdot c_{t-1}^l + i_t^l \cdot \tanh(W_{xc}^l x_t^l + W_{hc}^l h_{t-1}^l + b_c^l) \tag{6}$$

$$o_t^l = \sigma(W_{xo}^l x_t^l + W_{ho}^l h_{t-1}^l + W_{co}^l c_t^l + b_o^l) \tag{7}$$

$$r_t^l = o_t^l \cdot \tanh(c_t^l) \tag{8}$$

$$h_t^l = W_p^l \cdot r_t^l \tag{9}$$

l represents layer index and i_t^l, f_t^l and o_t^l are input, forget and output gates respectively. They are component-wise multiplied by input, memory cell and hidden output to gradually open or close their connections. x_t^l is an input from $(l-1)^{th}$ layer (or an input to a network when l is 1), h_{t-1}^l is a l^{th} output layer at time $t-1$ and c_{t-1}^l is an internal cell state at $t-1$. W_p^l is a projection matrix to reduce dimension of r_t^l.

2.4. Highway LSTM

Highway LSTM [20, 22] reused LSTM internal memory cells for spatial domain highway connections between stacked LSTM layers. Equations (4), (5), (7), (8), and (9) do not change for highway LSTM. Equation (6) is updated to add a highway connection:

$$\begin{aligned} c_t^l &= d_t^l \cdot c_t^{l-1} + f_t^l \cdot c_{t-1}^l + \\ &\quad i_t^l \cdot \tanh(W_{xc}^l x_t^l + W_{hc}^l h_{t-1}^l + b_c^l) \qquad (10) \\ d_t^l &= \sigma(W_{xd}^l x_t^l + W_{cd}^l c_{t-1}^l + w_{cd}^l c_t^{l-1} + b_d^l) \qquad (11) \end{aligned}$$

Where d_t^l is a depth gate that connects c_t^{l-1} in the $(l-1)^{th}$ layer to c_t^l in the l^{th} layer. [20] showed that an acoustic model based on the highway LSTM network improved far-field speech recognition compared with a plain LSTM network. However, [20] also showed that word error rate (WER) degraded when the number of layers in the highway LSTM network increases from 3 to 8.

3. Residual LSTM

In this section, a novel architecture for a deep recurrent neural network, residual LSTM is introduced. Residual LSTM starts with an intuition that the separation of a spatial-domain shortcut path with a temporal-domain cell update may give better flexibility to deal with vanishing or exploding gradients. Unlike highway LSTM, residual LSTM does not accumulate a highway path on an internal memory cell c_t^l. Instead, a shortcut path is added to an LSTM output layer h_t^l. For example, c_t^l can keep a temporal gradient flow without attenuation by maintaining forget gate f_t^l to be close to one. However, this gradient flow can directly leak into the next layer c_t^{l+1} for highway LSTM in spite of their irrelevance. On the contrary, residual LSTM has less impact from c_t^l update due to separation of gradient paths.

Figure 1 describes a cell diagram of a residual LSTM layer. h_t^{l-1} is a shortcut path from $(l-1)^{th}$ output layer that is added to a projection output m_t^l. Although a shortcut path can be any lower output layer, in this paper, we used a previous output layer. Equations (4), (5), (6) and (7) do not change for residual LSTM. The updated equations are as follows:

$$r_t^l = \tanh(c_t^l) \tag{12}$$

$$m_t^l = W_p^l \cdot r_t^l \tag{13}$$

$$h_t^l = o_t^l \cdot (m_t^l + W_h^l x_t^l) \tag{14}$$

Where W_h^l can be replaced by an identity matrix if the dimension of x_t^l matches that of h_t^l. For a matched dimension, Equation (14) can be changed into:

$$h_t^l = o_t^l \cdot (m_t^l + x_t^l) \tag{15}$$

Since a highway path is always turned on for residual LSTM, there should be a scaling parameter on the main path output. For example, linear filters in the last CNN layer of a residual network are reused to scale the main path output. For residual LSTM, a projection matrix W_p^l is reused in order to scale the LSTM output. Consequently, the number of parameters for residual LSTM does not increase compared with plain LSTM. Simple complexity comparison between residual LSTM and highway LSTM is as follows. If the size of the internal memory cells is N and the output layer dimension after projection

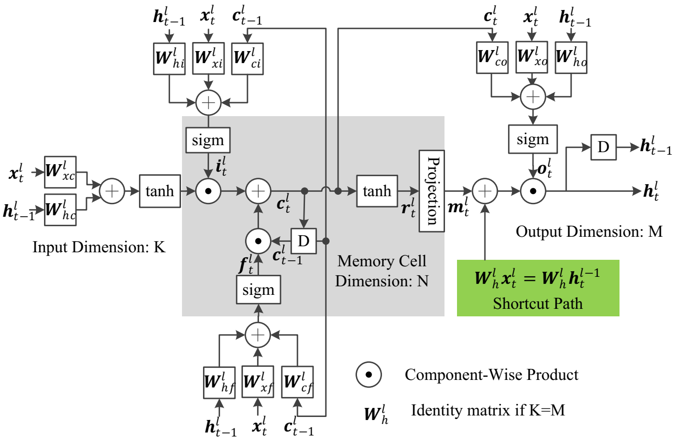

h_{t-1}^l x_t^l c_{t-1}^l
W_{hi}^l W_{xi}^l W_{ci}^l
sigm
i_t^l
x_t^l W_{xc}^l
h_{t-1}^l W_{hc}^l
tanh
Input Dimension: K
c_t^l
D
f_t^l c_{t-1}^l
sigm
Memory Cell Dimension: N
W_{hf}^l W_{xf}^l W_{cf}^l
h_{t-1}^l x_t^l c_{t-1}^l
tanh
r_t^l
Projection
m_t^l
c_t^l x_t^l h_{t-1}^l
W_{co}^l W_{xo}^l W_{ho}^l
sigm
o_t^l
D
h_{t-1}^l
h_t^l
Output Dimension: M
$W_h^l x_t^l = W_h^l h_t^{l-1}$
Shortcut Path
Component-Wise Product
W_h^l Identity matrix if K=M

Figure 1: *Residual LSTM: A shortcut from a prior output layer h_t^{l-1} is added to a projection output m_t^l. W_h^l is a dimension matching matrix between input and output. If K is equal to M, it is replaced with an identity matrix.*

is $N/2$, the total number of reduced parameters for a residual LSTM network becomes $N^2/2 + 4N$. For example, if N is 1024 and the number of layers is more than 5, the residual LSTM network has approximately 10% less network parameters compared with the highway LSTM network with same N and a projection matrix.

One thing to note is that a highway path should be scaled by an output gate as in Equation (14). The initial design of residual LSTM was to simply add an input path to an LSTM output without scaling, which is similar to a ResLSTM block in [24]. However, it showed significant performance loss because highway paths keep accumulated as the number of layers increase. For example, the first layer output without scaling would be $o_t^1 \cdot m_t^1 + x_t^1$, which consists of two components. For the second layer output, however, the number of components increases as three: $o_t^2 \cdot m_t^2 + o_t^1 \cdot m_t^1 + x_t^1$. Without proper scaling, the variance of an residual LSTM output will keep increasing.

The convolutional LSTM network proposed in [24] added batch normalization layers, which can normalize increased output variance from a highway path. For residual LSTM, output gate is re-used to act similarly without any additional layer or parameter. Output gate is a trainable network which can learn a proper range of an LSTM output. For example, if an output gate is set as $\frac{1}{\sqrt{2}}$, an l^{th} output layer becomes

$$h_t^l = \sum_{k=1}^{l} (\frac{1}{\sqrt{2}})^{(l-k+1)} m_t^k + (\frac{1}{\sqrt{2}})^l x_t \qquad (16)$$

Where, x_t is an input to LSTM at time t. If m_t^l and x_t are independent each other for all l and have fixed variance of 1, regardless of layer index l, the variance of layer l^{th} output becomes 1. Since variance of a output layer is variable in the real scenario, a trainable output gate will better deal with exploding variance than a fixed scaling factor.

4. Experiments

4.1. Experimental Setup

AMI meeting corpus [23] is used to train and evaluate residual LSTMs. AMI corpus consists of 100 hours of meeting recordings. For each meeting, three to four people have free conversation in English. Frequently, overlapped speaking from multiple speakers happens and for that case, the training transcript always follows a main speaker. Multiple microphones are used to synchronously record conversations in different environments. Individual headset microphone (IHM) recorded clean close-talking conversation and single distant microphone (SDM) recorded far-field noisy conversation. In this paper, SDM is used to train residual LSTMs at Section 4.2 and 4.3 and combined SDM and IHM corpora are used at Section 4.4.

Kaldi [25] is a toolkit for speech recognition that is used to train a context-dependent LDA-MLLT-GMM-HMM system. The trained GMM-HMM generates forced aligned labels which are later used to train a neural network-based acoustic model. Three neural network-based acoustic models are trained: plain LSTM network without any shortcut path, highway LSTM network and residual LSTM network. All three LSTM networks have 1024 memory cells and 512 output nodes for experiments at Section 4.2, 4.3 and 4.4.

The computational network toolkit (CNTK) [26] is used to train and decode three acoustic models. Truncated back-propagation through time (BPTT) is used to train LSTM networks with 20 frames for each truncation. Cross-entropy loss function is used with L2 regularization.

For decoding, reduced 50k-word Fisher dictionary is used for lexicon and based on this lexicon, tri-gram language model is interpolated from AMI training transcript. As a decoding option, word error rate (WER) can be calculated based on non-overlapped speaking or overlapped speaking. Recognizing overlapped speaking is to decode up to 4 concurrent speeches. Decoding overlapped speaking is a big challenge considering a network is trained to only recognize a main speaker. Following sections will provide WERs for both options.

4.2. Training Performance with increasing Depth

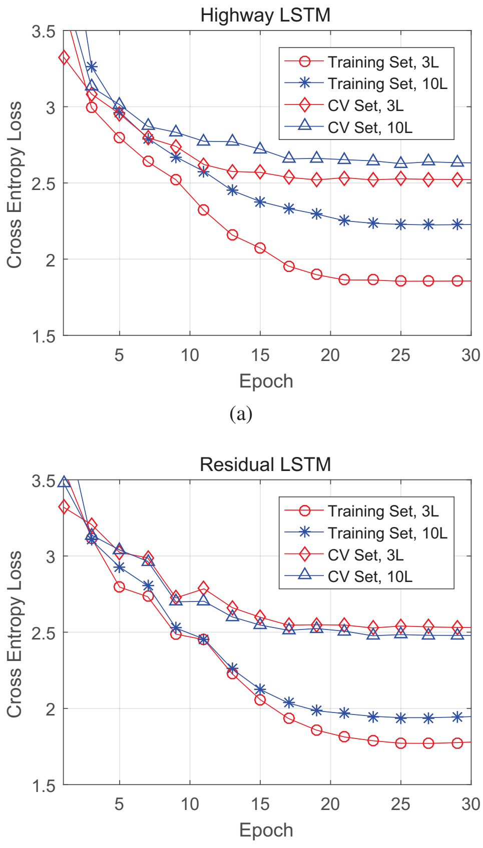
Figure 2 compares training and cross-validation (CV) cross-entropies for highway and residual LSTMs. The cross-validation set is only used to evaluate cross-entropies of trained networks.

In Figure 2a, training and CV cross-entropies for a 10-layer highway LSTM increased 15% and 3.6% over 3-layer one, respectively. 3.6% CV loss for a 10-layer highway LSTM does not come from overfitting because the training cross-entropy was increased as well. The training loss from increased network depth was observed in many cases such as Figure 1 of [17]. A 10-layer highway LSTM revealed the similar training loss pattern, which implies highway LSTM does not completely resolve this issue.

In Figure 2b, a 10-layer residual LSTM showed that its CV cross-entropy does not degrade with increasing depth. On the contrary, the CV cross-entropy improved. Therefore, residual LSTMs did not show any training loss observed in [17]. One thing to note is that the 10-layer residual LSTM also showed 6.7% training cross-entropy loss. However, the increased training loss for the residual LSTM network resulted in better generalization performance like regularization or early-stopping techniques. It might be due to better representation of input features from the deep architecture enabled by residual LSTM.

4.3. WER Evaluation with SDM corpus

Table 1 compares WER for plain LSTM, highway LSTM and residual LSTM with increasing depth. All three networks were trained by SDM AMI corpus. Both overlapped and non-overlapped WERs are shown. For each layer, internal mem-



(a)

(b)

Figure 2: *Training and CV PERs on AMI SDM corpus. (a) shows training and cross-validation (CV) cross-entropies for 3 and 10-layer highway LSTMs. (b) shows training and cross-validation (CV) cross-entropies for 3 and 10-layer residual LSTMs.*

ory cell size is set to be 1024 and output node size is fixed as 512. A plain LSTM performed worse with increasing layers. Especially, the 10-layer LSTM degraded up to 13.7% over the 3-layer LSTM for non-overlapped WER. A highway LSTM showed better performance over a plain LSTM but still could not avoid degradation with increasing depth. The 10-layer highway LSTM presented 6.2% increase in WER over the 3-layer network.

On the contrary, a residual LSTM improved with increasing layers. 5-layer and 10-layer residual LSTMs have 1.2% and 2.2% WER reductions over the 3-layer network, respectively. The 10-layer residual LSTM showed the lowest 41.0% WER, which corresponds to 3.3% and 2.8% WER reduction over 3-layer plain and highway LSTMs.

One thing to note is that WERs for 3-layer plain and highway LSTMs are somewhat worse than results reported in [20]. The main reason might be that forced alignment labels used to train LSTM networks are not the same as the ones used in [20]. 1-2% WER can easily be improved or degraded depending on the quality of aligned labels. Since the purpose of our evaluation is to measure relative performance between different LSTM architectures, small absolute difference of WER would not be any issue. Moreover, reproduce of highway LSTM is based on the open source code provided by the author in [20] and therefore, it would be less likely to have big experimental mismatch in our evaluation.

Table 1: *All three LSTM networks have the same size of layer parameters:1024 memory cells and 512 output nodes. Fixed-size layers are stacked up when the number of layers increases. WER(over) is overlapped WER and WER (non-over) is non-overlapped WER.*

Acoustic Model	Layer	WER (over)	WER (non-over)
Plain LSTM	3	51.1%	42.4%
	5	51.4%	42.5%
	10	56.3%	48.2%
Highway LSTM	3	50.8%	42.2%
	5	51.0%	42.2%
	10	53.5%	44.8%
Residual LSTM	3	50.8%	41.9%
	5	50.0%	41.4%
	10	50.0%	41.0%

Table 2: *Highway and residual LSTMs are trained with combined SDM and IHM corpora.*

Acoustic Model	Layer	WER (over)	WER (non-over)
Highway LSTM	3	51.3%	42.3%
	5	49.5%	40.7%
	10	52.1%	43.4%
Residual LSTM	3	50.8%	41.9%
	5	49.4%	40.5%
	10	48.7%	39.3%

4.4. WER Evaluation with SDM and IHM corpora

Table 2 compares WER of highway and residual LSTMs trained with combined IHM and SDM corpora. With increased corpus size, the best performing configuration for a highway LSTM is changed into 5-layer with 40.7% WER. However, a 10-layer highway LSTM still suffered from training loss from increased depth: 6.6% increase in WER (non-over). On the contrary, a 10-layer residual LSTM showed the best WER of 39.3%, which corresponds to 3.1% WER (non-over) reduction over the 5-layer one, whereas the prior experiment trained only by SDM corpus presented 1% improvement. Increasing training data provides larger gain from a deeper network. Residual LSTM enabled to train a deeper LSTM network without any training loss.

5. Conclusion

In this paper, we proposed a novel architecture for a deep recurrent neural network: residual LSTM. Residual LSTM provides a shortcut path between adjacent layer outputs. Unlike highway network, residual LSTM does not assign dedicated gate networks for a shortcut connection. Instead, projection matrix and output gate are reused for a shortcut connection, which provides roughly 10% reduction of network parameters compared with highway LSTMs. Experiments on AMI corpus showed that residual LSTMs improved significantly with increasing depth, meanwhile 10-layer plain and highway LSTMs severely suffered from training loss.

6. References

[1] G. E. 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, 2012.

[2] 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.

[3] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[5] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Large vocabulary continuous speech recognition with context-dependent DBN-HMMs," in *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2011, pp. 4688–4691.

[6] 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.

[7] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional neural networks for distant speech recognition," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1120–1124, 2014.

[8] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *2012 IEEE international conference on Acoustics, speech and signal processing (ICASSP)*. IEEE, 2012, pp. 4277–4280.

[9] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 8614–8618.

[10] O. Abdel-Hamid, A.-r. 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*, vol. 22, no. 10, pp. 1533–1545, 2014.

[11] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 273–278.

[12] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv preprint arXiv:1402.1128*, 2014.

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

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

[15] N. Morgan, "Deep and wide: Multiple layers in automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 7–13, 2012.

[16] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Advances in neural information processing systems*, 2015, pp. 2377–2385.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[18] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *Deep Learning Workshop(ICML 2015)*, 2015.

[19] T. Mikolov, A. Deoras, D. Povey, L. Burget, and J. Černockỳ, "Strategies for training large scale neural network language models," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE, 2011, pp. 196–201.

[20] Y. Zhang, G. Chen, D. Yu, K. Yaco, S. Khudanpur, and J. Glass, "Highway long short-term memory RNNs for distant speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5755–5759.

[21] K. Yao, T. Cohn, K. Vylomova, K. Duh, and C. Dyer, "Depth-gated LSTM," in *Presented at Jelinek Summer Workshop on August*, vol. 14, 2015, p. 1.

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

[23] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The AMI meeting corpus: A pre-announcement," in *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005, pp. 28–39.

[24] Y. Zhang, W. Chan, and N. Jaitly, "Very deep convolutional networks for end-to-end speech recognition," *arXiv preprint arXiv:1610.03022*, 2016.

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The KALDI speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[26] D. Yu, A. Eversole, M. Seltzer, K. Yao, Z. Huang, B. Guenter, O. Kuchaiev, Y. Zhang, F. Seide, H. Wang *et al.*, "An introduction to computational networks and the computational network toolkit," Technical report, Tech. Rep. MSR, Microsoft Research, 2014, 2014. research. microsoft. com/apps/pubs, Tech. Rep., 2014.