INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Multidimensional Residual Learning Based on Recurrent Neural Networks for Acoustic Modeling

Yuanyuan Zhao, Shuang Xu, Bo Xu

Interactive Digital Media Technology Research Center
Institute of Automation, Chinese Academy of Sciences, Beijing, P.R.China
{yyzhao5231, shuang.xu, xubo}@ia.ac.cn

Abstract

Theoretical and empirical evidences indicate that the depth of neural networks is crucial to acoustic modeling in speech recognition tasks. Unfortunately, the situation in practice always is that with the depth increasing, the accuracy gets saturated and then degrades rapidly. In this paper, a novel multidimensional residual learning architecture is proposed to address this degradation of deep recurrent neural networks (RNNs) on acoustic modeling by further exploring the spatial and temporal dimensions. In the spatial dimension, shortcut connections are introduced to RNNs, along which the information can flow across several layers without attenuation. In the temporal dimension, we cope with the degradation problem by regulating temporal granularity, namely, splitting the input sequence into several parallel sub-sequences, which can ensure information flowing across the time axis unimpededly. Finally, we place a row convolution layer on the top of all recurrent layers to comprehend appropriate information from several parallel sub-sequences to feed to the classifier. Experiments are illustrated on two quite different speech recognition tasks and 10% relative performance improvements are observed.

Index Terms: acoustic modeling, multidimensional residual learning, long short-term memory block, row convolution layer

1. Introduction

Acoustic modeling in speech recognition is a temporal sequence labelling task [1]. Recurrent neural networks (RNNs) based methods have shown state-of-the-art performance [2, 3] due to the powerful ability of modeling long-span dependency and the deeper architecture. For instance, Google's deep acoustic model with 5-layer long short-term memory (LSTM) has achieved impressive performance improvements on large vocabulary speech recognition tasks [2]. And the 11-layer architecture composed of gated recurrent unit layer and convolutional layer proposed in DeepSpeech2 [4] has closed the performance gap between machines and human workers on the transcription task.

Unfortunately, training deeper neural networks in practice is much difficult[5]. The more commonly situation is: with the depth increasing, the performance gets saturated and then degrades dramatically. Numerous attempts have been made to address this degradation issue, such as initialization schemes [6, 7], training networks in multiple stages [8, 9], and temporary companion loss functions attached to some layers [10, 11]. Recently, more promising techniques have been proposed: residual learning [12] and highway networks [13]. They have greatly alleviated this problem and shown better performance.

The work was supported by 973 Program in China, grant No. 2013CB329302.

Mathematically, RNNs' hidden state is a function of all previous hidden states. Accordingly, they could be considered as a T depth feedforward neural network unrolled in time where all layers share same model parameters [3]. Therefore, the degradation problem that deeper architectures suffer also exists in the temporal dimension. Due to two obvious differences, approaches to degradation in spatial dimension cannot be simply transferred to temporal dimension. Firstly, the depth of RNNs in time is always diverse and determined by the particular example. However, the depth of neural networks in space is always predesigned and fixed in experiments, which make some design possible, such as shortcut connections in residual networks [12]. Secondly, weights sharing of RNNs requires the dimension of input to every unrolled layer in different time depth to stay the same.

Above all, a deep RNNs architecture is actually multidimensional deep both spatially and temporally. As a result, the degradation problem in deep RNNs is more complex and severe. In this paper, a multidimensional residual learning framework is proposed to address the degradation problem both in space and time simultaneously. In the spatial dimension, instead of stacking layers straightforward, shortcut connections across several layers are brought in deep RNNs. These shortcut connections ensure the information flow forward across several layers and the error pass back across several layers without any attenuation. In temporal dimension, the input sequence to RNNs is split into several parallel subsequences, i.e.temporal granularity is regulated. Through this way, the history dependency information is utilized more efficiently and errors pass back much unimpededly. Furthermore, the property of the short-time stationarity in speech enables this to work well and the dimension of the input to RNNs unrolled in time remains constant. In order to comprehend the information from several parallel sequences, a row convolution layer is placed on the top of all recurrent layers to gather the appropriate information that facilitates the classification. What's more, for unidirectional LSTM networks, row convolution layer can utilize future context at each timestep to make an accurate prediction in an online, low-latency setting. Experimental results on two quite different speech recognition tasks prove that proposed deep multidimensional residual networks are much more trainable and easily converge to a better extremum, and 10% relative improvements are observed.

2. Related work

Shortcut connections across different layers [13, 14, 15] have been studied for a long time. While early works [14, 15] mainly focused on networks' convergence, highway networks [13] present shortcut connections with gating functions, which con-

http://dx.doi.org/10.21437/Interspeech.2016-677

cern more about the degradation problem. More recently, a more simple method called identity shortcut connections has proposed in the residual network [12] to address this degradation problem. It just added the activation of lower layer to certain layer parameter-freely.

Row convolution structure was first proposed in deep speech 2 [4] to address the deficiency of the future information in unidirectional models. They define a vector parameter for every frame of τ future time step, and multiply it with the corresponding activation. In addition, feed-forward sequential memory networks (FSMNs) [16] does a similar job, which encoded the N previous activities of the hidden layer to a fixed-size representation. They proposed two different variants: all hidden nodes share the same group of encoding coefficients and different hidden nodes adopt different coefficients.

3. Multidimensional residual learning

In this section, we will elaborate the architecture of deep multidimensional residual networks with LSTM. The solution to degradation problem in spatial dimension will be described firstly. Based on it, temporal dimension residual learning will be detailed on the following subsection and the motivation for this method will be explained as well. At last, the row convolution layer to comprehend information will be introduced.

3.1. Spatial dimension residual learning

The plain LSTM network (Figure 1(a)) typically consists of L layers and the output y_t^l of the l_{th} ($l \in [1, L]$) layer is calculated as follow:

$$y_t^l = h\left(x_t^l;\ y_{t-1}^l\right) \tag{1}$$

where $x_t^l = y_t^{l-1}$, h is the typical LSTM unit in accordance with [3].

A basic building block with shortcut connections skipping one layer of deep spatial dimension residual networks with LSTM is shown as Figure 1(b). Shortcut connections perform an identity mapping, along which information can flow directly and exactly without extra parameters. While the calculations of l_{th} and $(l+1)_{th}$ layers are the same as plain LSTM neural networks, the formulation of the $(l+2)_{th}$ layers is calculate as (2)-(6) and the biases are omitted for simplify:

$$i_t = \sigma\left(W_{ix}\tilde{x}_t + W_{im}m_{t-1} + W_{ic}c_{t-1}\right) \tag{2}$$

$$f_t = \sigma\left(W_{fx}\tilde{x}_t + W_{fm}m_{t-1} + W_{fc}c_{t-1}\right) \tag{3}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g\left(W_{cx}\tilde{x}_t + W_{cm}m_{t-1}\right) \tag{4}$$

$$o_t = \sigma\left(W_{ox}\tilde{x}_t + W_{om}m_{t-1} + W_{oc}c_t\right) \tag{5}$$

$$m_t = o_t \odot h\left(c_t\right) \tag{6}$$

where the major part is still the same as plain LSTM networks except that the inputs $\tilde{x}_t$ are made up of two parts: outputs of the l layer and the $l+1$ layer. It formulates as (7):

$$\tilde{x}_t = y_t^l \oplus y_t^{l+1} \tag{7}$$

there are two ways to deal with the add operation $\oplus$ in formulation (7): a) y_t^l spliced with y_t^{l+1} simply; b) y_t^l interpolation with y_t^{l+1}. Note that the former don't constrain the outputs of different lower layers to be in the same dimension. And the gradient error can pass to the lower layers directly.

As with plain deeper architecture, deep spatial dimensional residual networks with LSTM depicted in Figure 1(c) are built by stacking multiple basic building blocks (Figure 1(b)).

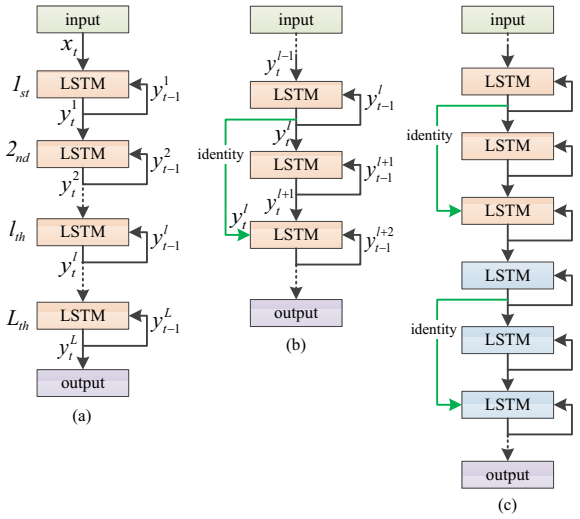

Figure 1: *Comparison of plain LSTM networks and deep spatial dimension residual networks. (a) plain LSTM networks; (b) a basic building block with shortcut connections skipping one layer; (c)deep spatial dimension residual networks with LSTM.*

While features from a given time instant are only processed by a few nonlinear layers before contributing the output in the basic building block, deep architecture can make better use of parameters by distributing them over the space through multiple layers. In other words, feature sequences are transformed from one feature space to another space, which makes the representation more discriminant. Furthermore, shortcut connections across several layers can make better use of the information flow from lower layers efficiently.

3.2. Temporal dimension residual learning

LSTM networks are inherently deep in time, since their hidden state is the function of all previous hidden states. As the degradation problem of neural networks in spatial dimension, the problem probably exists in depth in time as well. Other than methods in space, which add shortcut connections across several layers, we split the input sequence into several parallel subsequences to address the degradation in time. The motivation for this method is as follows. Firstly, the adjacent several feature frames of input sequences always contain much more similar content [17, 18]. That is, speech signal endows with the property of the short-time stationarity and feature frame is typically computed every 10 millisecond of the recordings with windows size of 25 millisecond span long. Secondly, the adjacent several target labels of output sequences always repeated, for the state of HMM usually sustained several frames. The input feature redundancy will make the information propagation congestedly, and the output label redundancy will make the error back-propagation crowdedly. Lastly, the length of speech signals can be up to hundreds or thousands of frames long, it brings much more challenges to LSTM RNN. All of these problems will result in the waste of the model's capabilities.

Based on the above observations, we address the degradation problem in temporal dimension by reducing the time resolution with a factor of j. Like the plain LSTM network, the input to the temporal dimension residual network is ordered according to speech signal sequence. And the hidden layer activa-

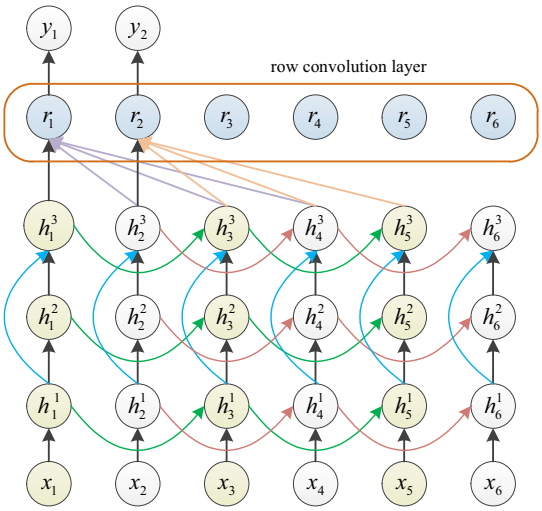

Figure 2: *The architecture of multidimensional residual networks with LSTM and regulating factor* 2 *in each building block and future context size* 3*.*

tion of the l_{th} layer at the t_{th} time step is computed as follow:

$$y_t^l = LSTM\left(x_t^l;\ y_{t-j}^l\right) \tag{8}$$

where x_t^l is y_t^{l-1} or calculated as formula (7) accordingly and j is regulating factor. In our model, the activation of hidden layers will be computed for every input frame, so the factor j can be various in different layers as long as propagating the error of current time back directly. The architecture of multidimensional residual networks with LSTM is shown in Figure 2, where in spatial dimension skipping one is adopted and in temporal dimension the factor of 2 is used for all layers.

3.3. Row convolution layer

The sequence of l_{th} hidden layer's activation is denoted as $\{h_1^l, \cdots, h_T^l\}$. Then the output activation of the row convolution layer at time-step t is formulated as follow:

$$r_{t,i}^{l+1} = \sum_{j=1}^{\tau+1} W_{i,j} h_{t+j-1,i}^l \ (1 \le i \le d) \tag{9}$$

where the matrix W is size of $d \times (\tau + 1)$, d and τ are the activation dimension of hidden layers and the future context size respectively.

The row convolution layer placed on the top of all recurrent layers has serval advantages. Firstly, the row convolution layer learns to gather the complementary information from parallel subsequences of deep multidimensional residual networks with LSTM. Moreover, it can utilize enough future context at current time to make an accurate prediction in an online, low-latency setting. Furthermore, it posses an attention mechanism of mining sophisticated information to predict robustly.

4. Experiment

4.1. Datasets and setup

In order to demonstrate the effectiveness of proposed deep multidimensional residual networks with LSTM, extensive experiments have been performed on two speech recognition tasks, namely phoneme recognition task on TIMIT [19] and text transcription task on HKUST Mandarin Chinese conversational telephone speech.

TIMIT corpus configuration is consistent with [20], except that 13-dimensional mel-frequency cepstral coefficients (MFCCs) features with first and second order derivatives are used. The HKUST corpus (LDC2005S15, LDC2005T32) collected and transcribed by Hong Kong University of Science and Technology (HKUST) [21] contains 150-hour speech, and 873 calls in the training set and 24 calls in the test set, respectively. The input to neural networks at each time step is a single frame of 43-dimensional log-mel filterbank features along with their first and second order derivatives.

The hybrid approach [22][3] is used in our experiments: all networks are trained with a cross-entropy loss using the fixed alignments which are generated by a well-trained GMM-HMM systems with 1938 or 2814 senones. In addition, a training strategy of 5 frames delayed target is adopted to make full use of future information.

Models are first pre-trained using the layer-wise pre-training algorithm [23] and then the truncated BPTT is used to update the model parameters [24]. We use a fixed time step T_{bptt} (e.g.20) to forward-propagate the activations and backward-propagate the gradients. For computational efficiency, the model is trained on our asynchronous stochastic gradient descent (ASGD) training platform [25], with four GPUs operating in parallel on 12 utterances at a time for each. The learning rate is decreased exponentially with an initial value of 1.0e-04.

4.2. Baseline systems

Two typical types of systems are established as baselines in our experiments: cross entropy (CE) training of hybrid system with HMM and end-to-end system trained with connectionist temporal classification (CTC). In the hybrid method, traditional deep neural network (DNN) and the popular LSTM RNNs are conducted. "DNN-6L-1024H" has 6 hidden layers with 1024 neurons in each layer and "LSTMP-3L-800C" has 3 layers with 800 LSTM cells projected to 512 units. In the end-to-end system trained with CTC criterion, the results of unidirectional model and bidirectional model on TIMIT task are cited from [20]. "LSTM-3L-421C" represents 3 LSTM layers with 421 cells in each layer and "BiLSTM-3L-500C" means a 3 layers bidirectional LSTM model with 500 cells in each layer. On the HKUST task, unidirectional architecture "LSTMP-3L-800C" and bidirectional architecture "BiLSTMP-3L-800C" are conducted for comparison with the following experiments. In addition, CTC systems on HKUST are trained as [26] with 6,724 grapheme output units for Chinese characters set.

Table 1: *Baselines of phoneme error rate (PER) or character error rate (CER) of hybrid and end-to-end systems with various configurations on TIMIT and HKUST tasks.*

Corpus	Type	Architecture	PER/CER
TIMIT	CE	DNN-6L-1024H	20.4
TIMIT	CE	LSTMP-3L-800C	18.2
TIMIT	CTC	LSTM-3L-421C [20]	19.6
TIMIT	CTC	BiLSTM-3L-500C [20]	18.6
HKUST	CE	DNN-6L-1024H	40.34
HKUST	CE	LSTMP-3L-800C	34.30
HKUST	CTC	LSTMP-3L-800C	35.63
HKUST	CTC	BiLSTMP-3L-800C	34.22

4.3. Results of spatial dimension residual learning

We compare plain LSTM RNNs with proposed spatial dimension residual networks with LSTM in different depth. Table 2 shows experimental results on the TIMIT and HKUST tasks. Note that the basic building block of deep spatial dimension residual network with LSTM is regarded as a whole in layer-wise pretraining. In the table, residual net represents our proposed deep spatial dimension residual network with LSTM and "LSTMP-9×800P512" means three basic building blocks are used. The equivalent parameter size of plain net without shortcut connections are also listed for comparison.

It can be seen that with the depth increasing, the performance of plain LSTM networks quickly gets saturated and then degrades on both two tasks. Stacking more layers straightforwardly do not obtain extra performance improvement in experiments. In contrast, the proposed deep spatial dimension residual networks with LSTM get continued performance improvements with the depth increasing, which confirms that the proposed methods can solve degradation problem to some extent. Comparing with the 9-layer plain LSTM RNNs, the proposed residual net obtains relative 6-8% performance improvement on two tasks. A major difference between the proposed network and the plain one is that the former is added with shortcut connections. A conclusion can be drawn that the shortcut connection is beneficial to information flow from lower layers to high layers without attenuation.

Table 2: *Comparison of plain LSTM and deep spatial dimension residual networks with LSTM in different depth on TIMIT and HKUST speech recognition tasks.*

corpus	model	plain net	residual net
TIMIT	LSTMP-3×800P512	18.2	18.2
TIMIT	LSTMP-6×800P512	18.2	17.3
TIMIT	LSTMP-9×800P512	18.3	**17.1**
HKUST	LSTMP-3×800P512	34.30	33.64
HKUST	LSTMP-6×800P512	34.60	32.71
HKUST	LSTMP-9×800P512	35.16	**32.52**

4.4. Results of temporal dimension residual learning

Since the temporal dimension residual learning is motivated by the redundance of information between adjacent frames and the short-time stationarity property of speech, we do some statistical analysis on the average target duration of the two datasets. Table 3 shows the statistical results. It can be seen that HKUST average duration is a little longer than that of TIMIT. According to this, we regulate the input granularity with a factor around 3 or 4 in HKUST tasks, while the factor around 2 is explored for TIMIT tasks.

Table 3: *Statistics on the average duration and percentage on different speech recognition tasks.*

corpus	aver duration	different duration portion		
		dura2	dura3	dura4
TIMIT	2.57	28.8%	7.97%	4.26%
HKUST	3.30	24.7%	13.2%	6.86%

Based on the statistics of average duration in Table 3, experiments on various configurations of different factors in deep multidimensional residual networks with three building blocks are conducted and the results are summarized in Table 4. The notation of "LSTMP-9×800P512-F124" represents that: 9 LSTM layers consisting of three basic building blocks with shortcut connections; each layer has 800 cells with 512 projection units; the regulating factors of three basic building blocks from bottom to up are 1, 2, 4 respectively. We obtain best results with the regulating factor around the average duration showing more than 10% relative improvement over plain LSTM networks with equivalent parameter size on both two tasks. It partially verify our assumption mentioned in section 3.2. And note that when the factor is too larger than average duration, the performance degrades. This is probably due to information loss of building a complete sentence in training.

Table 4: *Results of different regulating factor in deep multidimensional residual networks with LSTM on TIMIT and HKUST speech recognition tasks.*

corpus	model	PER/CER(%)
TIMIT	LSTMP-9×800P512-F222	**16.3**
TIMIT	LSTMP-9×800P512-F333	16.8
TIMIT	LSTMP-9×800P512-F444	16.9
TIMIT	LSTMP-9×800P512-F123	**16.5**
TIMIT	LSTMP-9×800P512-F124	17.0
HKUST	LSTMP-9×800P512-F222	33.58
HKUST	LSTMP-9×800P512-F333	**31.59**
HKUST	LSTMP-9×800P512-F444	**31.46**
HKUST	LSTMP-9×800P512-F123	32.55
HKUST	LSTMP-9×800P512-F124	32.14

4.5. Results of row convolution layer

The effect of the future context size in the row convolution layer is investigated in this section and results are listed in the Table 5. Model with a small future window size τ around the factor of last LSTM layer can obtain the best performance. Too large or too small future window size always get inferior performance. This is because too small τ can not acquire sufficient information and too large τ will bring more noise to the classifier.

Table 5: *The effect of future context size in row convolution layer on TIMIT and HKUST speech recognition tasks.*

corpus	model	$\tau 3$	$\tau 6$
TIMIT	LSTMP-9×800P512-F222	**16.1**	17.0
TIMIT	LSTMP-9×800P512-F123	16.3	17.2
HKUST	LSTMP-9×800P512-F444	**30.79**	32.08
HKUST	LSTMP-9×800P512-F124	31.24	32.92

5. Conclusion

In this paper, we proposed a deep multidimensional residual networks with LSTM for acoustic modeling to solve the degradation problem of deep RNNs. Shortcut connections are added along the spatial dimension and the sequence granularity are regulated in the temporal dimension. Based on this, a row convolution layer are stacked on the top of all recurrent layers. Experimental results on TIMIT and HKUST tasks with proposed networks show 10% relative performance improvement comparing with plain LSTM networks.

6. References

[1] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer-Verlag Berlin Heidelberg, 2012.

[2] H. Sak, A. W. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *CoRR*, vol. abs/1507.06947, 2015. [Online]. Available: http://arxiv.org/abs/1507.06947

[3] 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.

[4] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, J. Chen, M. Chrzanowski, A. Coates, G. Diamos *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," *arXiv preprint arXiv:1512.02595*, 2015.

[5] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks." *ICML (3)*, vol. 28, pp. 1310–1318, 2013.

[6] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[7] A. M. Saxe, J. L. McClelland, and S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks," *CoRR*, vol. abs/1312.6120, 2013. [Online]. Available: http://arxiv.org/abs/1312.6120

[8] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[9] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle *et al.*, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

[10] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," *arXiv preprint arXiv:1409.5185*, 2014.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

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

[13] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *CoRR*, vol. abs/1505.00387, 2015. [Online]. Available: http://arxiv.org/abs/1505.00387

[14] B. D. Ripley, *Pattern recognition and neural networks*. Cambridge university press, 2007.

[15] W. N. Venables and B. D. Ripley, *Modern applied statistics with S-PLUS*. Springer Science & Business Media, 2013.

[16] S. Zhang, C. Liu, H. Jiang, S. Wei, L. Dai, and Y. Hu, "Feedforward sequential memory networks: A new structure to learn long-term dependency," *arXiv preprint arXiv:1512.08301*, 2015.

[17] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," *CoRR*, vol. abs/1508.01211, 2015. [Online]. Available: http://arxiv.org/abs/1508.01211

[18] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," *arXiv preprint arXiv:1508.04395*, 2015.

[19] J. Garfolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, and N. Dahlgren, "The darpa timit acoustic-phonetic continuous speech corpus," *US Dept. of Commerce, NIST, Gaithersburg, MD, Feburary*, 1993.

[20] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *ICASSP*. IEEE, 2013, pp. 6645–6649.

[21] Y. Liu, P. Fung, Y. Yang, C. Cieri, S. Huang, and D. Graff, "Hkust/mts: A very large scale mandarin telephone speech corpus," in *Chinese Spoken Language Processing*. Springer, 2006, pp. 724–735.

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

[23] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE, 2011, pp. 24–29.

[24] R. J. Williams and J. Peng, "An efficient gradient-based algorithm for on-line training of recurrent network trajectories," *Neural Computation*, vol. 2, pp. 490–501, 1990.

[25] S. Zhang, C. Zhang, Z. You, R. Zheng, and B. Xu, "Asynchronous stochastic gradient descent for dnn training," in *ICASSP*. IEEE, 2013, pp. 6660–6663.

[26] J. Li, H. Zhang, X. Cai, and B. Xu, "Towards end-to-end speech recognition for chinese mandarin using long short-term memory recurrent neural networks," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.