MEMORY VISUALIZATION FOR GATED RECURRENT NEURAL NETWORKS IN SPEECH RECOGNITION

*Zhiyuan Tang, Ying Shi, Dong Wang**

Center for Speech and Language Technologies (CSLT), RIIT, Tsinghua University

{tangzy,shiying}@cslt.riit.tsinghua.edu.cn

*Corresponding Author:wangdong99@mails.tsinghua.edu.cn

ABSTRACT

Recurrent neural networks (RNNs) have shown clear superiority in sequence modeling, particularly the ones with gated units, such as long short-term memory (LSTM) and gated recurrent unit (GRU). However, the dynamic properties behind the remarkable performance remain unclear in many applications, e.g., automatic speech recognition (ASR). This paper employs visualization techniques to study the behavior of LSTM and GRU when performing speech recognition tasks. Our experiments show some interesting patterns in the gated memory, and some of them have inspired simple yet effective modifications on the network structure. We report two of such modifications: (1) a shortcut path from cells to outputs in LSTM, and (2) a residual learning mechanism for high-level cells. Both the two modifications, without any additional parameters, lead to more comprehensible and powerful networks.

Index Terms— long short-term memory, gated recurrent unit, visualization, residual learning, speech recognition

1. INTRODUCTION

Deep learning has gained brilliant success in a wide spectrum of research areas including automatic speech recognition (ASR) [1]. Among various deep models, recurrent neural network (RNN) is in particular interesting for ASR, partly due to its capability of modeling the complex temporal dynamics in speech signals as a continuous state trajectory, which essentially overturns the long-standing hidden Markove model (HMM) that describes the dynamic properties of speech signals as discrete state transition. Promising results have been reported for the RNN-based ASR [2–4]. A known issue of the vanilla RNN model is that training the network is generally difficult, largely attributed to the gradient vanishing and explosion problem. Additionally, the vanilla RNN model tends to forget things quickly. To solve these problems, a gated memory mechanism was proposed by researchers, leading to gated RNNs that rely a few trainable gates to select the most important information to receive, memorize, and propagate. Two widely used gated RNN structures are the long short-term memory (LSTM), proposed by Hochreiter [5], and the gated recurrent unit (GRU), proposed recently by Cho et al. [6]. Both the two structures have delivered promising performance in ASR [**?**].

Despite the success of gated RNNs, what has happened in the gated memory at run-time remains unclear in speech recognition. This prevents us from a deep understanding of the gating mechanism, and the relative advantage of different gated units can be understood neither intuitively nor systematicaly In this paper, we utilize the visualization technique to study the behavior of gated RNNs when performing ASR. The focus is on the evolution of the gated memory and the activity pattern of the gates. We are more interested in the difference of the two popular gated RNN units, LSTM and GRU, in terms of duration of memorization and quality of activity patterns. By visualization, the behavior of a gated RNN can be better understood, which in turn may inspire ideas for more effective structures. This paper reports two simple modifications inspired by the visualization results, and the experiments demonstrated that they do result in model that are not only more powerful but also more comprehensible.

The rest of the paper is organized as follows: Section 2 describes some related work, and Section 3 presents the experimental settings. The visualization results are presented in Section 4, and two modifications inspired by visualization are presented in Section 5. The entire paper is concluded in Section 6, with some future work discussed.

2. RELATED WORK

Visualization has been used in several research areas to study the behavior of neural models. For instance, in computer vision (CV), visualization is often used to demonstrate the hierarchical feature learning process with deep conventional neural networks (CNN). For example the activation maximization and composition analysis [7–9]. Natural language processing (NLP) is another area where visualization has been widely utilized. Since word/tag sequences are often modeled by an RNN, visualization in NLP focuses on analysis of temporal dynamics of units in RNNs [10–13].

In speech recognition (and other speech processing tasks), visualization has not been employed as much as in CV and NLP, partly because displaying speech signals as visual patterns is not as straightforward as for images and text. The only work we know for RNN visualization in ASR is conducted by Miao et al. [14], which studies the the input and forget gates of an LSTM, and found they are correlated. The visualization analysis presented in this paper differs from Miao's work in that our analysis is based on comparative studies, which identifying the most important mechanism for good ASR performance by comparing the behavior of different gated RNN structures (LSTM and GRU), in terms of memory values, memory residual and the gating effect.

Comparative analysis for LSTM and GRU was conducted by Chung et al. [15]. This paper is different from Chung's work in that we compare the two structures by visualization rather than by reasoning. Moreover, our analysis focus on group behavior of individual units (activity pattern), rather than an all-in-one performance.

3. EXPERIMENTAL SETUP

We first describe the LSTM and GRU structures whose behavior will visualized in following sections, and then describe the settings of the ASR system that the visualization is based on.

3.1. LSTM and GRU

We choose the LSTM structure described by Chung in [15], as it has shown good performance for ASR. The computation is as follows:

$$
\begin{aligned}
i_t &= \sigma(W_{ix}x_t + W_{im}m_{t-1} + V_{ic}c_{t-1}) && (1)\\
f_t &= \sigma(W_{fx}x_t + W_{fm}m_{t-1} + V_{fc}c_{t-1}) && (2)\\
c_t &= f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cm}m_{t-1}) && (3)\\
o_t &= \sigma(W_{ox}x_t + W_{om}m_{t-1} + V_{oc}c_t) && (4)\\
m_t &= o_t \odot h(c_t). && (5)
\end{aligned}
$$

In the above equations, the W and V terms denote weight matrices, where V's are diagonal. x_t is the input symbol; i_t, f_t, o_t represent respectively the input, forget and output gates; c_t is the cell and m_t is the unit output. $\sigma(\cdot)$ is the logistic sigmoid function, and $g(\cdot)$ and $h(\cdot)$ are hyperbolic activation functions. $\odot$ denotes element-wise multiplication. We ignore bias vectors in the formula for simplification.

GRU was introduced by Cho in [6]. It follows the same idea of information gating as LSTM, but uses a simpler structure. The computation is as follows:

$$
\begin{aligned}
i'_t &= \sigma(W'_{ix}x_t + W'_{ic}c'_{t-1}) && (6)\\
f'_t &= 1 - i'_t && (7)\\
o'_t &= \sigma(W'_{ox}x_t + W'_{oc}c'_{t-1}) && (8)\\
m'_t &= o'_t \odot c'_{t-1} && (9)\\
c'_t &= f'_t \odot c'_{t-1} + i'_t \odot g(W'_{cx}x_t + W'_{cm}m'_t). && (10)
\end{aligned}
$$

3.2. Speech recognition task

System	# of Recurrent Layers	WER%
LSTM	1	10.96
	2	9.97
	4	9.67
	6	9.47
GRU	1	10.76
	2	9.47
	4	9.32
	6	9.32

Table 1: Performance of LSTM and GRU systems

Our experiments are conducted on the WSJ database whose profile is largely standard: $37,318$ utterances for model training, 1049 utterances (involving dev93, eval92 and eval93) for testing. The input feature is 40-dimensional Fbanks, with a symmetric 2-frame window to splice neighboring frames. The number of recurrent layers varies from 1 to 6, and the number of units in each hidden layer is set to 512. The units may be LSTM or GRU. The output layer consists of $3,377$ units, equal to the total number of Gaussian components in the conventional GMM system used to bootstrap the RNN model.

The Kaldi toolkit [16] is used to conduct the model training and performance evaluation, and the training process largely follows the WSJ s5 nnet3 recipe. The natural stochastic gradient descent (NSGD) algorithm [17] is use to train the model. The results in terms of word error rate (WER) are reported in Table 1, where 'LSTM' denotes the system with LSTMs as the recurrent units, and 'GRU' denotes the system with GRUs as the recurrent units. We can observe that the RNN based on GRU units perform slightly better than the one based on LSTM units.

4. VISUALIZATION

We present the visualization results of the two kinds of gated units, focusing on the properties of memory neurons and gating values.

4.1. Neuronal differentiation

The memory neurons of LSTM and GRU show obvious differentiation, and the differentiation between different layers varies. Fig. 1 shows the activation density of the first and final layers of LSTM RNN. Activations in one layer of LSTM shows severe differentiation and the activations of higher layers become much more distributed and stable. According to values in fig. 1, we classify the memory cells of LSTM to different categories: silent neurons, whose values are among -1 and 1 with a possibility of 90%; and except the silent ones, positive neurons, whose values are positive with a possibility of 90%; negative neurons, whose values are negative with a possibility of 90%; regular neurons, whose values are among -5 and 5 with a possibility of 90%; excited neurons, more than half of whose activations are out of regular range; and others, which can not be grouped to any previous category. Fig. 2 shows the evolving of neuronal differentiation of each layer in a 4-layer LSTM RNN. From that figure, we find that the silent cells become more frequently activated in higher layers.

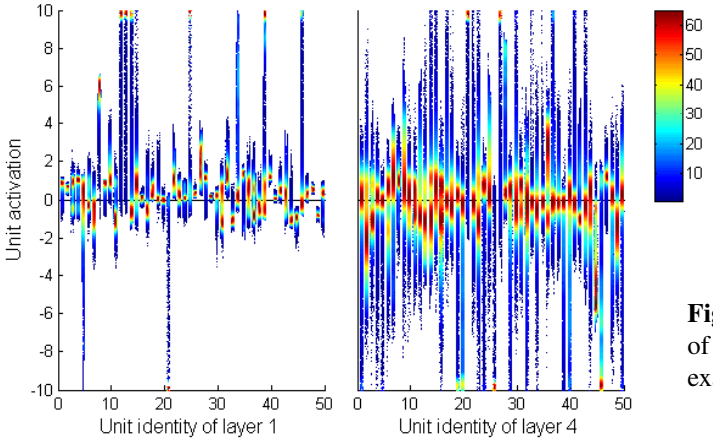

Fig. 1: The activation density of neurons in different layers of a 4-layer LSTM RNN, taking the first and the final layers as examples.

The activations of GRU neurons are much more stable and are always among -1 and 1, as shown in fig. 3. The neurons in the memory units of GRU show clear polarity: unipolar neurons, including positive and negative ones as in LSTM; and bipolar neurons, who activate both positively and negatively with a possibility of at least 40%. Fig. 4 shows that the bipolar memory units show a clear trend of growth.

4.2. Neuronal responsibility

A neuron can easily recognize a special phone by many patterns, such as, by simply activating unusual values. For LSTM, excited neurons are unusual, while silent neurons are much scarer for GRU. We can get a glimpse of the mechanism inside by studying these units' responsibility to recognize phones. We define the responsibility of a neuron to a special phone as the possibility with which this neuron activates unusually when meeting that phone. Fig. 5 shows the number of unusual units of different layers in LSTM responsible to all phones with possibility of more than 80%, and fig. 6 shows the same thing for GRU RNN, except that, we set the considerable responsibility to be more than 50% for GRU's stability. In both two figures, we find that, neurons of lowest layer trend to activate slightly and the presentation of labels are quiet distributed. The difference exits that the first layer of LSTM RNN is less distributed, while the first layer of GRU show more responsibilities.

If a neuron shows much higher/lower responsibility to one phone than any other phones, this neuron is much more possible to recognize this phone. Following this rule, we study a single LSTM, and find that several units can recognize special phones on their own, as shown in fig. 7.

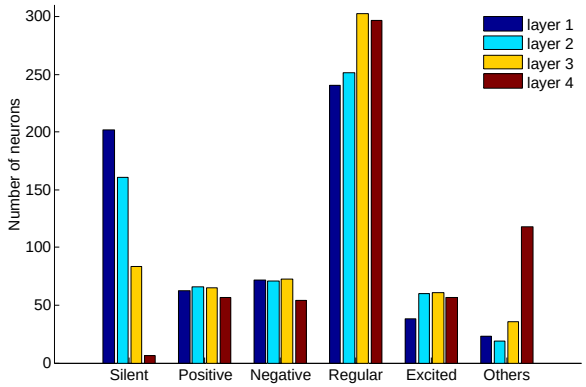

Fig. 2: The evolution of neurons in different layers of a 4-layer GRU RNN.

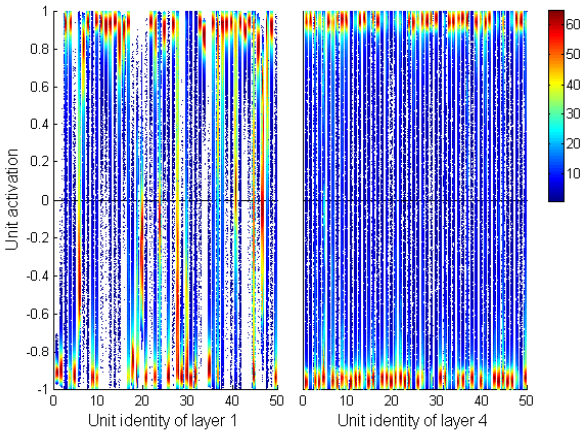

Fig. 3: The activation density of neurons in different layers of a 4-layer GRU RNN, taking the first and the final layers as examples.

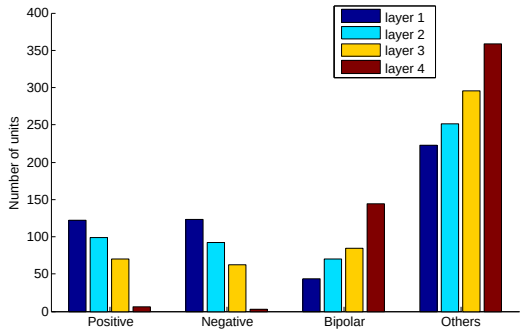

Fig. 4: The evolution of neurons in different layers of a 4-layer LSTM RNN.

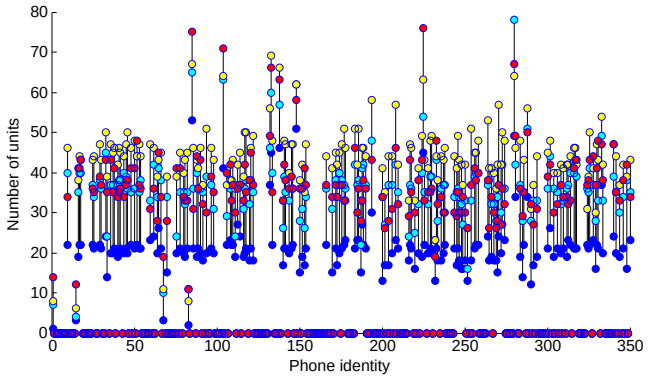

Fig. 5: Total numbers of unusual units with responsibility of more than 80% to all phones in 4-layer LSTM RNN.

4.3. Gating saturation

We use the metric of saturation mentioned in [11]. The visualization of the number of gates prone to close or open is shown in fig. 8.

4.4. Temporal hierarchy

We study the time scale of LSTM and GRU RNN. The results of t-SNE on one long sentence, as presented in fig. 9, show that LSTM tends to keep a longer memory than that of GRU. Comparing the time scale of different layers of two kinds of RNNs, we find that the memory length of LSTM always keeps long, while for GRU, the memory length of last layer is longer that of previous layers.

This can also be confirmed by inserting noise into the context of speech frames and detecting the end of the influence of the noise, as shown in fig. 10.

4.5. Temporal residual

Recurrent neural networks are actually kind of residual learning along time. Fig. 11 shows the original activations of all memory cells of one LSTM on the same sentence used in

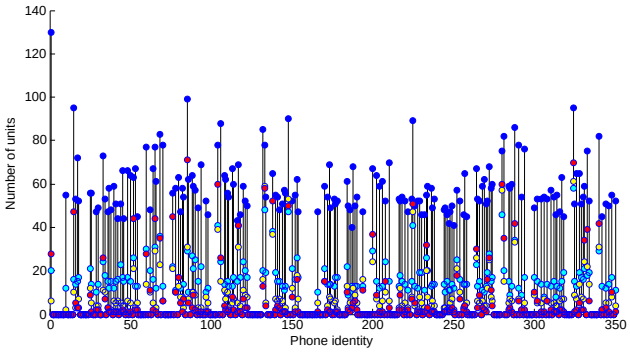

Fig. 6: Total numbers of units with responsibility of more than 50% to all phones in 4-layer GRU RNN.

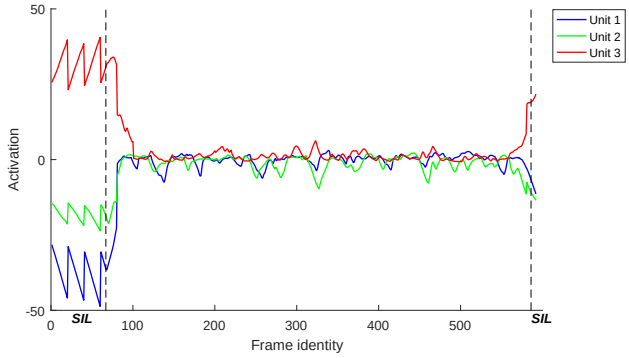

Fig. 7: Single units in a 1-layer LSTM RNN recognize the phone 'SIL'.

fig 7, and the residual activations on the same condition. We find that almost all the residuals are among -1 and 1, similar to that original activations of GRU. And the pattern of residual activations is much plainer and the front-end phone 'SIL' is recognized with ease.

5. APPLICATION TO NEURAL STRUCTURE

By reordering the computation of LSTM, LSTM fits a shorter memory term as GRU. Inspired by the memory pattern, we introduce residual learning into the memory. Both of the two modifications improve the system for ASR.

5.1. Memory exposure

Comparing the computations of LSTM and GRU in sec 3.1, the most distinct difference between LSTM and GRU is that GRU exposes its memory content directly to the classifier, that may influence the memory scale, as shown in fig. 9.

We introduce this property into LSTM by reordering its computation of the final layer of LSTM. The modification is shown in (a) of fig. 12, and the results are shown in table 2.

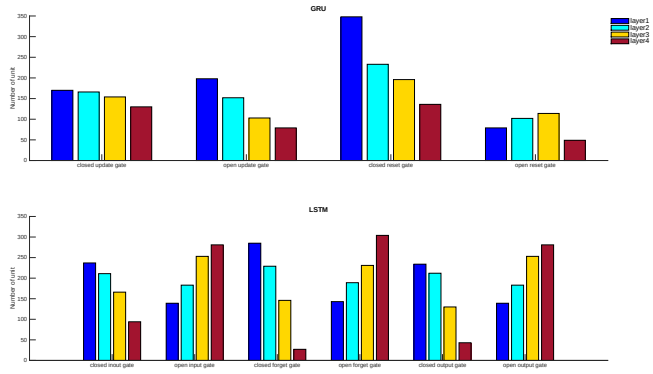

Fig. 8: The number of gates prone to close or open in different layers of 4-layer LSTM and 4-layer GRU RNN.

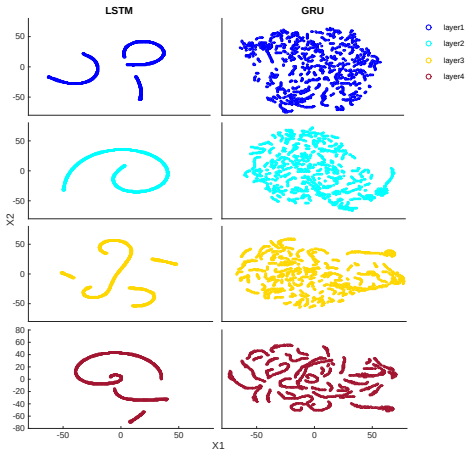

Fig. 9: The evolution of neurons in different layers of a 4-layer LSTM RNN.

# of Layers	WER%
1	10.18
2	9.48
4	9.10

Table 2: Performance of reordered LSTM

5.2. Memory residual

From the visualization of gates, we find that the higher layers' gates show a similar pattern. This implies that the memory in the higher layers are mostly learned by residual. Residual learning along time is presented in fig. 11. We introduce residual learning into the memory cells to make memory residual learning more explicit.

We also apply t-SEN to the memory activations of the two modifications as shown in 13.

6. CONCLUSION

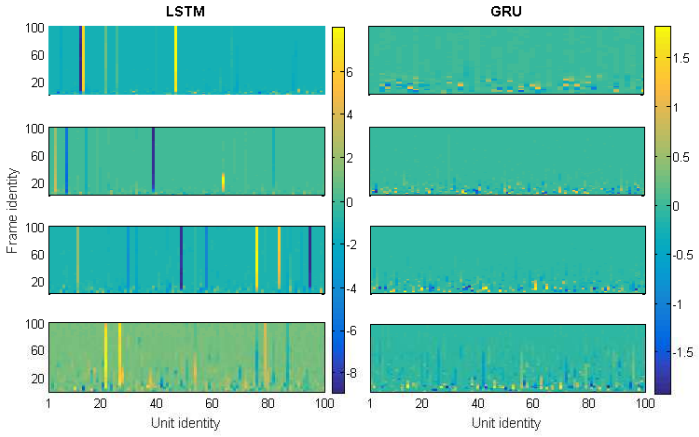

Fig. 10: The evolution of neurons in different layers of a 4-layer LSTM RNN.

Recurrent Type	# of Layers	WER%
LSTM	4	9.53
	6	9.33
GRU	4	9.23
	6	9.10

Table 3: Performance of two units with memory residual

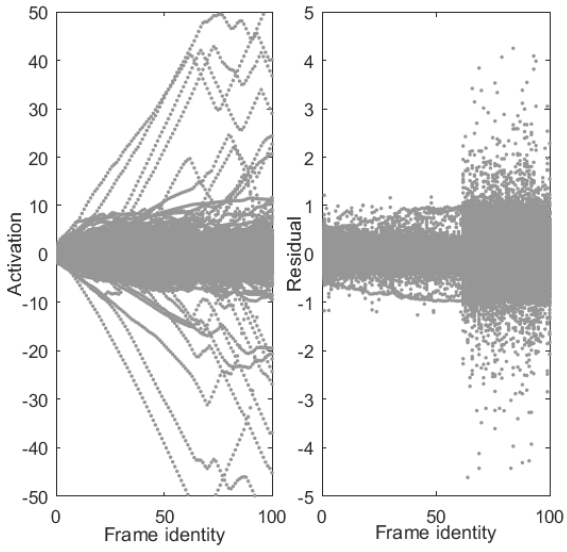

Fig. 11: Single units in a 1-layer LSTM RNN recognize the phone 'SIL'.

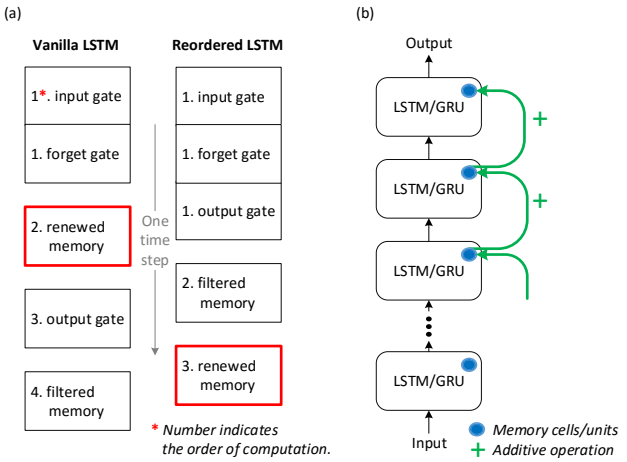

Fig. 12: Two modifications applied to the neural structures. (a) Reordering computation of LSTM to make its memory expose. (b) Memory residual into higher layers of LSTM/GRU.

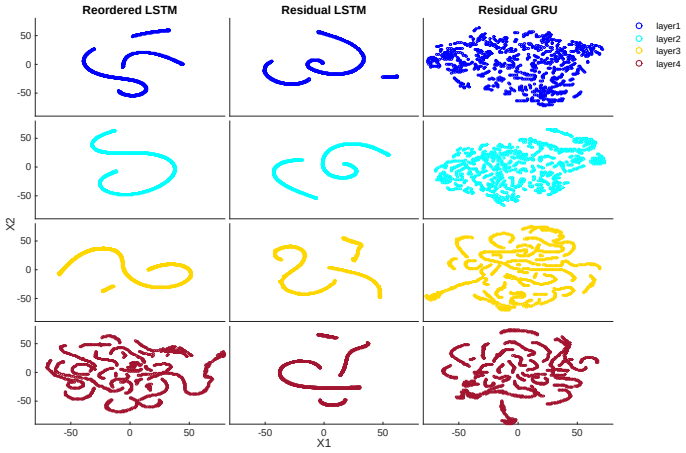

Fig. 13: t-SNE results of memory cells/units of modified LSTM and GRU during one sentence.

7. REFERENCES

[1] Li Deng and Dong Yu, "Deep learning: Methods and applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3-4, pp. 197–387, 2013.

[2] Alex Graves, A-R Mohamed, and Geoffrey Hinton, "Speech recognition with deep recurrent neural networks," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 6645–6649.

[3] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1764–1772.

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

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

[6] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[7] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Visualizing higher-layer features of a deep network," *University of Montreal*, vol. 1341, 2009.

[8] Matthew D Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[9] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[10] Michiel Hermans and Benjamin Schrauwen, "Training and analysing deep recurrent neural networks," in *Advances in Neural Information Processing Systems*, 2013, pp. 190–198.

[11] Andrej Karpathy, Justin Johnson, and Li Fei-Fei, "Visualizing and understanding recurrent networks," *arXiv preprint arXiv:1506.02078*, 2015.

[12] Ákos Kádár, Grzegorz Chrupała, and Afra Alishahi, "Representation of linguistic form and function in recurrent neural networks," *arXiv preprint arXiv:1602.08952*, 2016.

[13] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky, "Visualizing and understanding neural models in nlp," *arXiv preprint arXiv:1506.01066*, 2015.

[14] Yajie Miao, Jinyu Li, Yongqiang Wang, Shi-Xiong Zhang, and Yifan Gong, "Simplifying long short-term memory acoustic models for fast training and decoding," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 2284–2288.

[15] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[16] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[17] Daniel Povey, Xiaohui Zhang, and Sanjeev Khudanpur, "Parallel training of dnns with natural gradient and parameter averaging," *arXiv preprint arXiv:1410.7455*, 2014.