SPATIAL AND CHANNEL ATTENTION BASED CONVOLUTIONAL NEURAL NETWORKS FOR MODELING NOISY SPEECH

Sirui Xu, Eric Fosler-Lussier*

Department of Computer Science and Engineering
The Ohio State University

ABSTRACT

In recent years, Residual Networks (ResNets) have significantly increased the modeling power of convolutional neural networks (CNNs) by introducing residual connections. In this paper, we explore the incorporation of spatial and channel attention into the structure of ResNets for noisy speech recognition tasks. In our experiments, we implemented spatial attention as a bottom-up top-down structure where the input features are first down sampled and then up sampled to generate attention maps. At each block of the ResNet, the generated CNN features are composed with spatial attention maps over the temporal-frequency space, learning to attend to salient acoustic features and suppress noise. Our model also includes channel attention that attends to different channels of feature maps. ResNet blocks with spatial and channel attention modules can be easily stacked to construct deeper networks. We show that the proposed network structure has the ability to suppress noisy signals in speech audio without requiring parallel clean speech for training, and achieve promising WER reductions on CHiME2 and CHiME3.

Index Terms— Noisy Speech Recognition, Acoustic Modeling, Residual Neural Networks, Attention Mechanism

1. INTRODUCTION

Recent advances in training neural networks have significantly improved performance in various artificial intelligence tasks. The attention mechanism, which allows selective focus of the network at different points in time, has been extensively studied in areas such as image classification [1, 2], image caption generation [3], machine translation [4], and speech recognition [5, 6]. These studies formulate attention as a sequential process and assign attention weights over the sequence to capture useful information.

Because of the sequential characteristics of attention, recurrent neural networks (RNNs) and long short term memory networks (LSTMs) are widely used to model the attention mechanism in sequential problems with encoder-decoder structures, where the encoder converts the input sequence (e.g., a sequence of words or speech frames) into an embedding sequence and the decoder generates output labels based on the embeddings as well as the attention weights. The use of such structures has demonstrated outstanding performance in many sequence-to-sequence problems.

Attention mechanisms have also found their way into feed forward models, such as DNNs or CNNs, in sequence-to-sequence modeling. For example, Gehring *et al.* [7] applied CNNs to machine translation tasks and achieved significant improvement over RNN models with higher computational efficiency. Another study by Vaswani *et al.* [8] also used feed forward networks for neural machine translation. In their work, the researchers proposed what is called the Transformer Network in handling machine translation problems and achieved state-of-the-art performance.

What is particularly interesting about Vaswani *et al.*'s work is the introduction of self-attention into machine translation. Instead of assigning attention over sequences, self-attention explores local feature structures and is generated based on the dot product of the previous layer's output, thus helping to generate better feature embeddings and improve performance. Povey *et al.* [9] similarly adopted a self-attention layer in a time-restricted manner to jointly attend to different positions centered around the current time frame.

Self-attention has also been widely explored in other research areas such as image semantic segmentation [10], human pose estimation [11], and image classification [12]. In these studies, self-attention is implemented by incorporating a bottom-up top-down (hourglass shape) structure into blocks of convolutional layers to explore feature maps at different levels of the networks for each input image. The results of the studies showed significant improvement in respective tasks achieved by the self-attention structure.

Drawing from previous work, we experiment with a bottom-up top-down structure in automatic speech recognition (ASR). Although the bottom-up top-down structure has been used in autoencoders, the novelty of our work lies in the integration of the structure into CNNs as self-attention for noise reduction without parallel clean speech. We refer to this structure as the spatial attention module and use it to extract spatial feature patterns within CNNs' feature maps. Residual Neural Networks (ResNets) [13], a variant of CNN,

*Currently works at ObEN, Inc.

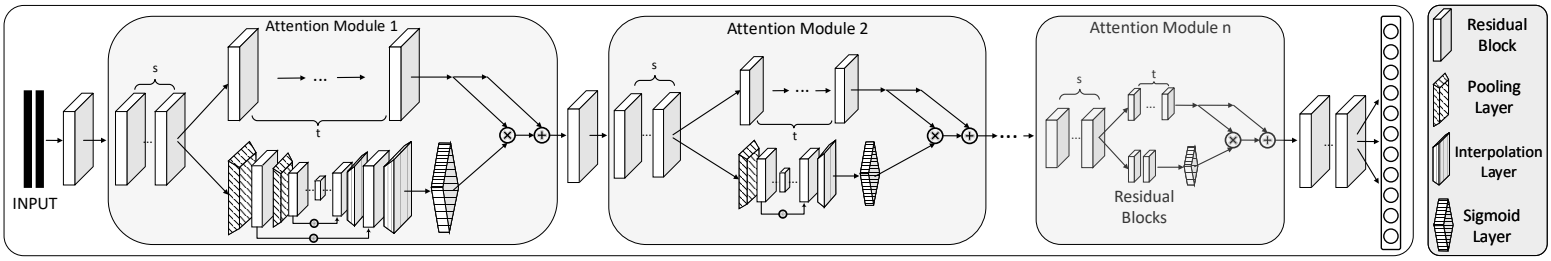

Fig. 1. An example of the network formed by stacking attention modules. The upper branch is the convolutional branch, and the lower branch is the attention branch which uses the bottom-up top-down structure to generate attention weight. The hyperparameter *s* denotes the number of residual blocks used to process features prior to the branches, while *t* denotes the number of residual blocks used in the convolutional branch.

are used for modeling acoustic features and simultaneously trained with the spatial attention modules with the same back-propagated error information.

After training, the spatial attention module generates spatial attention matrices at each time step. However, instead of assigning attention weights over different positions of the acoustic frame sequence, the spatial attention is multiplied with local features, emphasizing the most salient features and suppressing noise in each input context window.

Spatial attention can be viewed as a masking technique, but it has some critical differences from speech enhancement techniques. First, the goal of the learning process is not to generate masks for denoising, but rather for improving the ASR acoustic model's prediction of senone labels. Second, instead of only composing with the input features, the attention matrices are applied on multiple feature extraction levels. Finally, unlike supervised speech enhancement, learning the attention does not require parallel clean speech.

Since multiple feature map channels are generated at each level of a CNN network, we also integrate channel attention to explicitly model the varying importance of different channels. Combining spatial attention within one feature map, and channel attention across channels of feature maps shows improved recognition performance on noisy speech data.

The rest of the paper is organized as follows. Section 2 describes the experimental methods, including the detailed construction of spatial and channel attention structures. Section 3 describes the CHiME2 and CHiME3 datasets as well as the features we used. In Section 4, we present the experimental setup and the results of our experiments. Finally, Section 5 concludes the paper and discusses future work directions.

2. METHODOLOGY

In this study, ResNets serve as the base structure for acoustic modeling [14, 15, 16, 17]. ResNets introduce skip connections between blocks of convolutional layers, improving the propagation of gradients and allowing training even deeper CNNs without optimization difficulties (e.g. gradient vanishing) in a traditional CNN training scheme. We detail in the next sections how spatial and channel attention is integrated into the ResNet architecture.

2.1. Spatial Attention

Our spatial attention module is a weight assigning scheme that can be attend to different parts of the current input feature (locally), differing from traditional attention mechanisms that attend to positions over a sequence. An example network formed by concatenating spatial attention modules is shown in Fig. 1, similar to the structure used in [12].

The process of generating the attention matrix first down samples the input feature map several times until the size of the feature map reaches a pre-set resolution, and then upsample through interpolation until it gets back to the original size. The down-sampling process can be implemented by a max pooling or average pooling operation. Convolution layers are inserted between the pooling layers to make the attention generation process trainable. In traditional CNNs, pooling layers can have some effect on suppressing noise in feature maps, but as receptive fields are local the cannot access context information outside the region. To alleviate the problem, we apply the bottom-up top-down structures on the whole feature map; by explicitly using global information, we can generate attention weights to better suppress noise signals and emphasize salient features.

After getting the attention weights, we conduct an element-wise multiplication between the attention weight matrix and the feature map. Similar to the concept of residual learning, we add the output of the convolutional branch to the product, which is used to generate the final weights through a sigmoid function. This can be expressed as:

$$W_i, c^{(x)} = \sigma((1 + M_{i,c}(x)) \otimes F_{i,c}(x)), \qquad (1)$$

where $M_{i,c}(\cdot)$ and $F_{i,c}(\cdot)$ are the outputs of the attention and convolutional branches, and σ is the sigmoid function.

2.2. Channel Attention

In addition to spatial attention, we also incorporate a channel attention/gating mechanism (first proposed in [18]). The idea

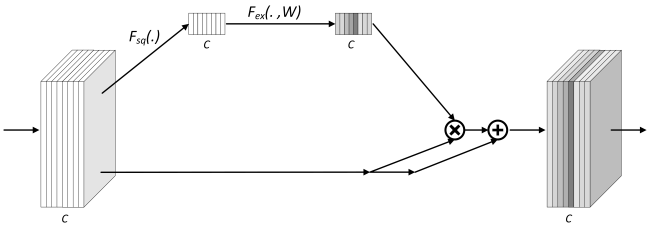

Fig. 2. An example of the channel attention unit.

is similar to spatial attention, with the purpose of trying to incorporate global information to explicitly model the importance between different channels in feature maps. The structure of a channel attention module is shown in Fig. 2. The process of generating channel attention weights can be split into two steps. The first is the squeezing step that produces the global information embedding, calculated as:

$$e_c = F_{sq}(x_c) = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} x_c(i,j), \tag{2}$$

where e_c is the global embedding for channel c, x_c is the feature in the c^{th} channel, and W and H are the width and height of the input feature map. What the squeezing step does is to calculate the average value of each channel.

The second step is the excitation step, which uses global embedding to recalibrate the feature map by first generating a scaling weight vector and scaling channels with the weights. The scaling weight vector is obtained by feeding the global embedding vector into a feed-forward layer and then through a sigmoid function. The process can be expressed as:

$$\mathbf{s} = F_{ex}(\mathbf{e}, \mathbf{W}) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{e})), \tag{3}$$

$$\tilde{\mathbf{x}} = F_{scale}(\mathbf{s}, \mathbf{x}) = \mathbf{s} \otimes \mathbf{x}, \tag{4}$$

where $\mathbf{s}$ is the vector of the scaling weights, and $\delta(\cdot)$ is the rectifier linear unit (ReLU) activation function. $\tilde{\mathbf{x}}$ is the channel weighted feature map, and $\otimes$ is a channel-wise multiplication between the scaling weight vector and the original input feature map.

We also adopt the concept of residual learning in the channel attention module, and add a skip connection between the input feature map to the output of the channel scaling to get the final output:

$$\mathbf{O} = \tilde{\mathbf{x}} + \mathbf{x} = (\mathbf{1} + \mathbf{s}) \otimes \mathbf{x}, \tag{5}$$

In our work, we found that the spatial attention module can help the model focus on speech related features and suppress noises, and the channel attention can learn to automatically assign more weights to channels with more helpful information to better recognize the speech. Combining both attention modules, we achieved significant improvement on noisy datasets.

3. DATASETS AND FEATURES

For our experiments, we used two noisy datasets: CHiME2 and CHiME3. Both datasets are constructed based on the WSJ0 corpus.

The CHiME2 dataset contains 7138 utterances in total, recorded by 83 different speakers. For the noisy speech, the signal-to-noise (SNR) ratio ranges from -6 to 9 db and is randomly selected for each utterance. The training set includes 6 different levels of SNR, i.e. -6, -3, 0, 3, 6, 9 dB. The development and evaluation set respectively contains 409 noisy utterances from 10 speakers and 330 noisy utterances from 8 other speakers.

The CHiME3 dataset includes both simulated and real noisy speech utterances. In total, the training set contains 7138 simulated utterances of 83 speakers of the WSJ0 SI-84 training set and 1600 real noisy utterances of 4 speakers. Both the simulated and real noisy utterances are generated in four different noise conditions: café (CAF), street junction (STR), public transport (BUS) and pedestrian area (PED). The development set contains a total of 3280 utterances (1640 simulated and real utterances respectively) and the test set includes 2640 utterances (1320 simulated and real utterances respectively).

The features used to train the ResNets are 40-dimensional fbank features extended with deltas and double-deltas. The context window length is 5, which makes the input feature include 11 frames. For CNNs, the input features are organized as $40 \times 11 \times 3$ tensors.

4. EXPERIMENTAL SETUP AND RESULTS

We used Kaldi to build the speech recognition pipeline and TensorFlow [19] to train the network for acoustic modeling. Senone labels were generated by a GMM/HMM system in Kaldi; for CHiME2, labels were generated from the clean data but the acoustic model was trained on nosiy data. For CHiME3, these were trained from noisy data (no clean speech or clean labels were used): beamforming was first performed to extract enhanced speech from the 5-channel microphone array signals, and then features were created based on the enhanced wave files.

In our experiments, we used attention modules with two residual blocks prior to the branches in the modules ($s = 2$) and two residual blocks inside the convolutional branch ($t = 2$). The residual block adopts the structure in [13], where each one includes two convolutional layers; batch normalization is used after each convolutional layer, followed by a ReLU activation function.

For both CHiME2 and CHiME3, we constructed two attention-based ResNets with 20 and 40 convolutional layers.

Network	Component	Output size	Filter
Att-ResNet-20	Conv layer	40 × 11	7 × 7, 128
	Attention module	40 × 11	3 × 3, 128
	Residual block	10 × 11	3 × 3, 256
	Attention module	10 × 11	3 × 3, 256
	Residual block	5 × 6	3 × 3, 512
Att-ResNet-40	Conv layer	40 × 11	7 × 7, 64
	Attention module	40 × 11	3 × 3, 64
	Residual block	20 × 11	3 × 3, 128
	Attention module	20 × 11	3 × 3, 128
	Residual block	10 × 11	3 × 3, 256
	Attention module	10 × 11	3 × 3, 256
	Residual block	5 × 6	3 × 3, 512
	Attention module	5 × 6	3 × 3, 512
	Residual block	5 × 6	3 × 3, 1024

Table 1. Structures of networks in experiments. Att-ResNet-20 is the network with 20 convolutional layers; Att-ResNet-40 has 40 convolutional layers.

Network	CHiME 2	CHiME 3	
		Real	Simu
7-layer DNN	17.8%	20.5%	24.3%
ResNet-20	17.1%	20.3%	24.0%
Att-ResNet-20-sp	16.5%	19.8%	23.7%
Att-ResNet-20-sp+ch	16.1%	19.7%	23.5%
ResNet-40	16.2%	20.1%	23.8%
Att-ResNet-40-sp	14.1%	19.7%	23.4%
Att-ResNet-40-sp+ch	13.7%	19.5%	23.4%

Table 2. WER for different models. Att-ResNet-sp refers to spatial attention based ResNet; Att-ResNet-sp+ch refers to spatial and channel attention based ResNet. 20 and 40 refer to the number of convolutional layers.

Table 1 shows the components of both networks and their respective properties.

We compared the performance of the attention ResNets with two baseline systems. The first one is a 7-layer feed-forward DNN, and the second is a ResNet with the same number of convolutional layers. The results are shown in Table 2.

Introducing attention helps to improve recognition performance. As Table 2 shows, for CHiME2, the 20-layer spatial attention based ResNet achieved 0.6% absolute WER reduction compared with the 20-layer ResNet with no attention, and the 40-layer spatial attention based ResNet outperformed the baseline by 2.1%. The added channel attention further reduced the WER for both the 20-layer and 40-layer ResNets. The best result came from the 40-layer spatial and channel attention based ResNet, achieving a WER reduction of 2.5% and 4.1% over the 40-layer ResNet and 7-layer DNN.

For CHiME2, Bagchi *et al.* [20] introduced a spectral mapper to generate clean speech, reporting a WER of 14.7% on denoised speech. Although we only worked on acoustic modeling on noisy speech, we achieved slightly better performance compared to [20]. This shows that the attention modules are capable of modeling some types of noisy speech with no clean speech during training, and can be useful in situations where parallel clean speech is not accessible.

For CHiME3, the results also showed improvement with the added attention modules, although it was not as significant as that for the CHiME2 dataset. Both the 20-layer and 40-layer spatial and channel attention based ResNets achieved WER reductions compared with the respective baselines.

5. DISCUSSION AND CONCLUSION

This paper explores integrating spatial and channel attention into training ResNets for noisy speech recognition. Our experiments showed considerable performance improvements; the best result came from the 40-layer spatial and channel attention based ResNet on CHiME2, with an absolute WER reduction of 2.5% and 4.1% respectively over a 40-layer ResNet without attention and a 7-layer DNN. This result is even better than some recent research that uses clean speech.

The performance discrepancy between CHiME2 and CHiME3 may be a result of the availability/unavailability of clean labels. CHiME2 provides clean speech, which we used to generate clean labels. In CHiME3, however, clean speech is only available for the simulated set but not the real set; therefore, we used labels generated from the noisy speech for both sets, which could have an impact on the performance. Another possible reason is that our proposed attention model may be more effective handling the types of noise in CHiME2. The noisy conditions in CHiME3 may pose further challenges for the model to distinguish between speech and noise signals.

Overall, integrating spatial and channel attention into CNNs can be helpful for noisy speech recognition, particularly in situations where parallel clean speech is inaccessible. In our experiments, we applied attention based ResNets on acoustic modeling, but the model structure can also be used for other problems. It can be generally used as a feature extractor and plugged into places where CNNs are used. In addition, the spatial and channel attention modules are not restricted to ResNets only; they can be coupled with other types of network structures as well. For future work, we would like to explore the application of attention mechanisms with different network structures and use them in other domains of research such as denoising.

6. ACKNOWLEDGEMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Quadro P6000 GPU used for this research. Additional computing resources are provided by the Ohio Supercomputer Center [21].

7. REFERENCES

[1] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al., "Recurrent models of visual attention," in *Advances in neural information processing systems*, 2014, pp. 2204–2212.

[2] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu, "Multiple object recognition with visual attention," *CoRR*, vol. abs/1412.7755, 2014.

[3] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015, pp. 2048–2057.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014.

[5] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.

[6] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4945–4949.

[7] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin, "Convolutional sequence to sequence learning," *CoRR*, vol. abs/1705.03122, 2017.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 6000–6010.

[9] Daniel Povey, Hossein Hadian, Pegah Ghahremani, Ke Li, and Sanjeev Khudanpur, "A time-restricted self-attention layer for ASR," 2018.

[10] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1520–1528.

[11] Alejandro Newell, Kaiyu Yang, and Jia Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*. Springer, 2016, pp. 483–499.

[12] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang, "Residual attention network for image classification," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[14] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Michael L Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig, "Toward human parity in conversational speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 25, no. 12, pp. 2410–2423, 2017.

[15] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig, "The Microsoft 2016 conversational speech recognition system," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5255–5259.

[16] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, et al., "English conversational telephone speech recognition by humans and machines," *Interspeech*, pp. 132–136, 2017.

[17] Yu Zhang, William Chan, and Navdeep Jaitly, "Very deep convolutional networks for end-to-end speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4845–4849.

[18] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[19] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al., "TensorFlow: A system for large-scale machine learning.," in *OSDI*, 2016, vol. 16, pp. 265–283.

[20] Deblin Bagchi, Peter Plantinga, Adam Stiff, and Eric Fosler-Lussier, "Spectral feature mapping with mimic loss for robust speech recognition," *ICASSP*, 2018.

[21] Ohio Supercomputer Center, "Ohio supercomputer center," http://osc.edu/ark:/19495/f5s1ph73, 1987.