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

Recurrent Models for Auditory Attention in Multi-Microphone Distant Speech Recognition

Suyoun Kim[1], *Ian Lane*[1]

[1]Electrical Computer Engineering
Carnegie Mellon University
suyoun@cmu.edu, lane@cmu.edu

Abstract

Integration of multiple microphone data is one of the key ways to achieve robust speech recognition in noisy environments or when the speaker is located at some distance from the input device. Signal processing techniques such as beamforming are widely used to extract a speech signal of interest from background noise. These techniques, however, are highly dependent on prior spatial information about the microphones and the environment in which the system is being used. In this work, we present a neural attention network that directly combines multi-channel audio to generate phonetic states without requiring any prior knowledge of the microphone layout or any explicit signal preprocessing for speech enhancement. We embed an attention mechanism within a Recurrent Neural Network based acoustic model to automatically tune its attention to a more reliable input source. Unlike traditional multi-channel preprocessing, our system can be optimized towards the desired output in one step. Although attention-based models have recently achieved impressive results on sequence-to-sequence learning, no attention mechanisms have previously been applied to learn potentially asynchronous and non-stationary multiple inputs. We evaluate our neural attention model on the CHiME-3 task, and show that the model achieves comparable performance to beamforming using a purely data-driven method.

Index Terms: distant speech recognition, multi-microphone processing

1. Introduction

Many real-world speech recognition applications, including teleconferencing, robotics and in-car spoken dialog systems, must deal with speech from distant microphones in noisy environments. When a human voice is captured with far-field microphones in these environments, the audio signal is severely degraded by reverberation and background noise. This makes the distant speech recognition task far more challenging than near-field speech recognition, which is commonly used for voice-based interaction today.

Acoustic signals from multiple microphones can be used to enhance recognition accuracy due to the availability of additional spatial information. Many researchers have proposed techniques to efficiently integrate inputs from multiple distant microphones. The most representative multi-channel processing technique is the beamforming approach [1, 2, 3, 4], which generates an enhanced single output signal by aligning multiple signals through digital delays that compensate for the different distances of the input signals. However, the performance of beamforming is highly dependant on prior information about microphone location and the location of the target source. For downstream tasks such as speech recognition, this preprocessing step is suboptimal because it is not directly optimized towards the final objective of interest: speech recognition accuracy [5].

Over the past few years, deep neural networks (DNNs) have been successfully applied to acoustic models in speech recognition [6, 7, 8]. Other works [9, 10, 11, 12, 13, 14] have shown that DNNs can learn suitable representations for distant speech recognition by directly using multi-channel input. These approaches, however, simply concatenated acoustic features from multiple microphones without considering the spatial properties of acoustic signal propagation, used convolutional neural networks (CNNs) to implicitly account for spatial relationships between channels [10, 11, 15, 16], or required pretrained beamforming network [14].

Recently, an "attention mechanism" in neural networks has been proposed to address the problem of learning variable-length input and output sequences [17]. At each output step, the previous output history is used to generate an attention vector over the input sequence. This attention vector enables models to learn to focus attention on specific parts of their input. These attention-equipped frameworks have shown very promising results on many challenging tasks involving inputs and outputs with variable length, including machine translation [17], parsing [18], image captioning [19] and conversational modeling [20]. Specifically, for the speech recognition tasks, [21, 22, 23] attempted to align the input features and the desired character sequence using an attention mechanism. However, no attention mechanisms have been applied to learn to integrate multiple inputs.

In this work, we propose a novel attention-based model that enables to learn misaligned and non-stationary multiple input sources for distant speech recognition. We embed an attention mechanism within a Recurrent Neural Network (RNN) based acoustic model to automatically tune its attention to a more reliable input source among misaligned and non-stationary input sources at each output step. The attention module is learned with the normal acoustic model and is jointly optimized towards phonetic state accuracy. Our attention module is unique in the way that we 1) deal with the problem of integrating different qualities and misalignment of multiple sources, and 2) exploit spatial information between multiple sources to accelerate learning of auditory attention. Our system plays a similar role to traditional multichannel preprocessing through deep neural network architecture, but bypasses the limitations of preprocess-

http://dx.doi.org/10.21437/Interspeech.2016-326

ing, which requires an expensive, separate step and depends on prior information.

Through a series of experiments on the CHiME-3 [24] dataset, we show that our proposed approach improves recognition accuracy in various types of noisy environments. In addition, we also compare our approach with the beamforming technique[24, 25, 26, 27]. The paper is organized as follows: in Section 2 we describe our proposed attention based model. In section 3, we evaluate the performance of our model. Finally, in Section 4 we draw conclusions.

2. Model

In this section, we describe our neural attention model, which allows neural networks to focus more on reliable input sources across different temporal locations. We formulate the proposed framework with applications in multi-channel distant speech recognition. While there has been some recent work on end-to-end neural speech recognition systems - from speech directly to transcripts [28, 29, 30, 21] - our model is based on typical hybrid DNN-HMM frameworks [31, 8], wherein the acoustic model estimates hidden Markov model (HMM) state posteriors, because we focus on dealing with the re-weighted input representation of misaligned multiple input sources.

Given a set of input sequences $\mathbf{X} = \{\mathbf{X}^{ch_1}, \cdots, \mathbf{X}^{ch_N}\}$, where $\mathbf{X}^{ch_i}$ is an input sequence $(x_1^{ch_i}, \cdots, x_T^{ch_i})$ from the ith microphone, $i \in \{1, \cdots, N\}$, our system computes a corresponding sequence of HMM acoustic states, $\mathbf{y} = (y_1, \cdots, y_T)$. We model each output $\mathbf{y}_t$ at time t as a conditional distribution over the previous outputs $y_{<t}$ and the multiple inputs $\mathbf{X}_t$ at time t using the chain rule:

$$P(\mathbf{y}|\mathbf{X}) = \prod_t P(y_t|\mathbf{X}, y_{<t}) \tag{1}$$

Our system consists of two subnetworks: $\mathrm{AttendMultiSource}$ and LSTM-AM. $\mathrm{AttendMultiSource}$ is an attention-equipped Recurrent Neural Network (RNN) for learning to determine and focus on reliable channels and temporal locations among the candidate multiple input sequences. $\mathrm{AttendMultiSource}$ produces re-weighted inputs, $\widehat{\mathbf{X}}$, based on the learned attention. This $\widehat{\mathbf{X}}$ is used for the next subnetwork LSTM-AM, which is a Long Short-Term Memory (LSTM) acoustic model to estimate the probability of the output HMM state $\mathbf{y}$. Figure 1 visualizes our overall model with these two components. We describe more details of each component in the following subsections 2.1 and 2.2.

$$\hat{\mathbf{X}} = \mathrm{AttendMultiSource}(\mathbf{X}, \mathbf{y}) \tag{2}$$

$$P(\mathbf{y}|\mathbf{X}) = \mathrm{LSTM\text{-}AM}(\hat{\mathbf{X}}, \mathbf{y}) \tag{3}$$

2.1. Attention mechanism for multiple sources

The challenge we attempt to address with the neural attention mechanism is the problem of misaligned multiple input sources with non-stationary quality over time. Specifically, in multi-channel distant speech recognition, the arrival time of each channel is different because the acoustic path length of each signal differs according to the location of the microphone. This results in the misalignment of input features. These differences in arrival time are even greater when the space between microphones is larger. Even worse, signal quality across channels can also vary over time because the speaker and interfering noise sources may keep changing. Figure 1 describes the asynchronous arrival of multiple inputs due to acoustic path length differences.

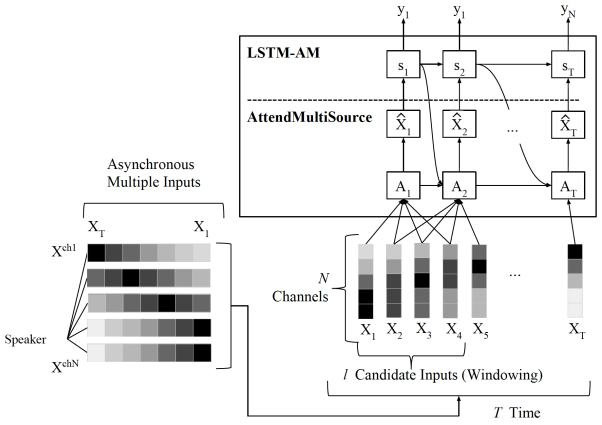

Figure 1: Schematic representation of our neural attention model.

We now introduce an attention mechanism to cope with the misaligned input problem, and formulate the $\mathrm{AttendMultiSource}$. At every output step t, the $\mathrm{AttendMultiSource}$ function produces a re-weighted input representation $\widehat{\mathbf{X}_c}$, given cth candidate input set $\mathbf{X}_c$. $\mathbf{X}_c$ is a subsequence of time frames. As proposed by [23], we perform similar windowing to limit the exploring temporal location of inputs for computational efficiency and scalability. We limit the range of attention to l=7 time frames (± 3). In our experiments, longer time steps had little impact on overall performance and would rather benefit from microphones placed further apart from each other.

For re-weighting the input $\mathbf{X}_c$, $\mathrm{AttendMultiSource}$ predicts an attention weight matrix $\mathbf{A}_t^{time,ch}$ at each output step t. Unlike previous attention mechanisms, we produce a weight matrix rather than a vector, because our attention mechanism additionally identifies which channel, in a given time step, is more relevant. Therefore, $\mathbf{A}_t^{time,ch}$ is the (*number of channels*) by (*number of candidate input frames*) matrix - here it is N x l matrix. Attention weights are calculated based on four different information sources: 1) attention history $\mathbf{A}_{t-1}^{time,ch}$, 2) content in the candidate sequences $\mathbf{X}_c$, 3) decoding history $\mathbf{s}_{t-1}$, and 4) additional spatial information between multiple microphones based on phase difference information $\mathbf{PD}_c$ corresponding to $\mathbf{X}_c$. The following three formulations describe the $\mathrm{AttendMultiSource}$ function:

$$\mathbf{E}_t^{time,ch} = \mathrm{MLP}(\mathbf{s}_{t-1}, \mathbf{A}_{t-1}^{time,ch}, \mathbf{PD}_c, \mathbf{X}_c) \tag{4}$$

$$\mathbf{A}_t^{time,ch} = \mathrm{softmax}(\mathbf{E}_t^{time,ch}) \tag{5}$$

$$\widehat{\mathbf{X}_c} = \mathbf{A}_t^{time,ch} \cdot \mathbf{X}_c \tag{6}$$

Once we compute the energy $\mathbf{E}_t^{time,ch}$ at time t, then we obtain $\mathbf{A}_t^{time,ch}$ by normalizing $\exp(\mathbf{E}_t^{time,ch}) / \sum_{time,ch} \exp(\mathbf{E}_t^{time,ch})$, such that, $\forall t$, $\mathbf{A}_t^{time,ch} \geq 0$, and $\sum_{time,ch} \mathbf{A}_t^{time,ch} = 1$ (in equation 5). Finally, re-weighted output $\widehat{\mathbf{X}_c}$ is generated by calculating the dot product of the attention weights $\mathbf{A}_t^{time,ch}$ and candidate

input $\mathbf{X}_c$ (in equation 6). Typically, the selection of elements from input candidates is a weighted sum. However, we only calculate the dot product in order to avoid losing information.

To accelerate the learning of the attention mechanism, we use additional spatial information based on analysis of differences in arrival time. It is generally assumed that the human auditory system can localize multiple sounds and attend to the desired signal using information from the interaural time difference (ITD) [32, 33]. A previous study [34] attempted to emulate human binaural processing and estimate ITD indirectly by comparing the phase difference between two microphones at each frequency domain. The authors identified a "close" time-frequency component to the speaker based on the estimated ITD. Similarly, we use the phase difference between two microphones to infer spatial information. The following equations are used to compute phase difference between two microphones i and j, where $i \neq j, i, j \in \{1 \cdots N\}$:

$$pd^{ch_i - ch_j} = \min |\angle x^{ch_i} - \angle x^{ch_j} - 2\pi r| \quad (7)$$

$$\mathbf{PD}^{ch_i - ch_j} = (pd_1^{ch_i - ch_j}, \cdots, pd_T^{ch_i - ch_j}) \quad (8)$$

$$\mathbf{PD} = \{\mathbf{PD}^{ch_1 - ch_2}, \cdots, \mathbf{PD}^{ch_4 - ch_5}\} \quad (9)$$

From these equations, we calculate the phase differences of each time-frequency bin of each pair of multiple microphones. In our work, we use 256 frequency bins for 25ms windows. The phase feature $\mathbf{PD}$ is calculated in every pair of channels, then the MLP network accepts the $\mathbf{PD}_c$ corresponding to the input candidates, with $\mathbf{X}_c$ as an additional input.

2.2. LSTM Acoustic Model

Our next subnetwork LSTM-AM serves as a typical RNN-based acoustic model, except that it accepts the re-weighted input $\widehat{\mathbf{X}_c}$ instead of the original input $\mathbf{X}_c$. LSTM-AM uses a Long Short-Term Memory RNN (LSTM)[35], which has been successfully applied to speech recognition tasks due to its ability to handle long-term dependencies. The LSTM contains special units called memory blocks in the recurrent hidden layer, and each block has memory cells c_t with special three-gates (input i_t, output o_t, and forget f_t) to control the flow of information.

In our work, we use a simplified version of an LSTM without peephole connections and biases to reduce the computational expense of learning the standard LSTM models. Although LSTMs have many variations for enhancing their performance, such as BLSTM [36], LSTMP [37], and PBLSTM [22], in our work, we focus on verifying an additional attention mechanism with a simple LSTM architecture, instead of improving LSTM acoustic modeling overall.

LSTM-AM maps a re-weighted input sequence based on the attention mechanism $\widehat{\mathbf{X}} = \{\widehat{\mathbf{x}^{ch_1}}, \cdots, \widehat{\mathbf{x}^{ch_N}}\}$, where $\widehat{\mathbf{x}^{ch_i}} = (\widehat{x_1^{ch_i}}, \cdots, \widehat{x_T^{ch_i}})$, to an output sequence $\mathbf{y}_t = (y_1, \cdots, y_T)$ by calculating the network unit activations using the following equations iteratively from $t = 1$ to T:

$$i_t = \sigma(\widehat{\mathbf{x}_c} W_{xi} + h_{t-1} W_{hi}) \quad (10)$$

$$f_t = \sigma(\widehat{\mathbf{x}_c} W_{xf} + h_{t-1} W_{hf}) \quad (11)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(\widehat{\mathbf{x}_c} W_{xc} + h_{t-1} W_{hc}) \quad (12)$$

$$o_t = \sigma(\widehat{\mathbf{x}_c} W_{xo} + h_{t-1} W_{ho}) \quad (13)$$

$$s_t = o_t \cdot \tanh(c_t) \quad (14)$$

where W terms denote weight matrices, and σ the logistic sigmoid function. i_t, f_t, o_t, and c_t are the input gate, forget gate, output gate and cell activation vectors, respectively. Finally, the output s_t is used to predict the current HMM state label by softmax (in equation 14). s_t is also used to predict the next $t+1$ attention matrix as well as the next c_{t+1} hidden state of LSTM-AM.

3. Experiments

3.1. Dataset

We evaluated the performance of our architecture on the CHiME-3 task. The CHiME-3 [24] task is automatic speech recognition for a multi-microphone tablet device in an everyday environment - a cafe, a street junction, public transport, and a pedestrian area. There are two types of datasets: `REAL` and `SIMU`. The `REAL` data consists of 6-channel recordings. 12 US English speakers were asked to read the sentences from the WSJ0 corpus [38] while using the multi-microphone tablet. They were encouraged to adjust their reading positions, so that the target distance kept changing over time. The simulated data `SIMU` was generated by mixing clean utterances from WSJ0 into background recordings. To verify our method in a real noisy environment, we first chose not to use the simulated dataset but rather to use only the `REAL` dataset, with 5 channels from the five microphones, which were located in each corner of tablet, about 10cm to 20cm away from each other (we excluded one microphone, which faced backward in the tablet device). We then evaluated our system on the full CHiME3 dataset, `MULTI`, including `REAL` and `SIMU`.

3.2. System Training

All the networks were trained on the 1,600 utterance (about 2.9 hours) `REAL` dataset and then on the 8,738 utterance (about 18 hours) `MULTI` dataset. The dataset was represented with 25ms frames of 40-dimensional log-filterbank energy features computed every 10ms. We produced 1,992 HMM state labels from a trained GMM-HMM system using near-field microphone data, and these state labels were used in all subsequent experiments. We use one layer of LSTM architecture with 512 cells. The weights in all the networks were initialized to the range (-0.03, 0.03) with a uniform distribution, and the initial attention weights were initialized to $1/n$ in n dimensions. We set the configuration of the learning rate to 0.4 and after two epochs it decays during training. All models resulted in a stable convergence range from 1e-04 to 5e-04. To avoid the exploding gradient problem, we limited the norm of the gradient to 1 [39]. Apart from the gradient clipping, we did not limit the activations of the weights.

During training, we evaluated frame accuracies (i.e. phone state labeling accuracy of acoustic frames) on the development set of 1,640 utterances in `REAL` and 3,280 utterances in `MULTI`. The trained models were evaluated in a speech recognition system on a test set of 1,320 utterances. For all the decoding experiments, we used a size 18 beam and size 10 lattices. There is a mismatch between the Kaldi baseline [40] and our results because we did not perform sequence training (sMBR) or language model rescoring (5-gram rescoring or RNNLM). The inputs for all networks were log-filterbank features, with 5 channels stacking, and then with 7 frames stacking (+3-3).

Table 1: Comparison of WERs(%) on development and evaluation set of the subset (REAL) of the CHiME-3 task between the three baseline systems, and our proposed framework, ALSTM. The models are trained on on real data (3hrs).

MODEL (Input)	Dev	Eval
Baselines - Real (3hrs)		
LSTM (Preprocessing noisy 5 mics)	35.2	52.1
LSTM (single noisy mic)	39.1	57.1
LSTM (5 noisy mics)	43.0	60.1
Proposed - Real (3hrs)		
ALSTM	35.9	52.3
ALSTM (with phase)	33.9	50.0

Table 2: Comparison of WERs(%) on development and evaluation set of the subset (REAL) of the CHiME-3 task between the baseline system, and our proposed framework, ALSTM. The models are trained on on real + simulated data (18hrs).

MODEL (Input)	Dev	Eval
Baselines - Real + Simu (18hrs)		
LSTM (Preprocessing noisy 5 mics)	18.6	32.0
Proposed - Real + Simu (18hrs)		
ALSTM (with phase)	16.5	26.5

3.3. Results

In Table 1 and 2, we summarize word error rates (WERs) obtained on the subset of the CHiME3 task. ALSTM is our proposed model, which has an attention mechanism for multiple inputs as described in 2.1, and ALSTM (with phase) used phase information in addition to ALSTM.

As our baselines, we built three models on the `REAL` dataset and used the same simple version of the LSTM architecture that we described in Section 2.2 with three different inputs. LSTM (Preprocessing 5 noisy-channel) was trained on the enhanced signal from 5 noisy channels. We obtained the enhanced signal from the beamforming toolkit, which was provided by the CHiME3 organizer [24, 25, 26, 27]. LSTM (single noisy-channel) was trained on a single noisy channel, and LSTM (5 noisy-channels) used the concatenated 5 noisy channels. We also built LSTM (Preprocessing 5 noisy-channel) on the `MULTI` dataset.

As expected, LSTM (Preprocessing 5 noisy-channel) provided a substantial improvement in WER compared to LSTM (single noisy-channel) and LSTM (5 noisy-channel), showing a 13.3% and 5.0% relative improvement in WER, respectively. We also found that the model, which simply combined 5 features across microphones, did not perform very well. It showed poorer results than even the model trained with single microphone data. This result underscores the importance of integrating channels based on analysis of differences in arrival times.

Our model with the attention mechanism provided a significant improvement in WER compared to LSTM (5 noisy-channel). Compared to LSTM (5 noisy-channel), ALSTM (with phase) achieved a 17% reduction in relative error rate on the evaluation set, and ALSTM achieved a 13% relative error rate. These results suggest that we can leverage the attention mechanism to integrate multiple channels efficiently. To ensure the improvement of the system was coming from our time-channel attention mechanism, we compared our model to a model with an attention mechanism across time only on single-channel input. This comparison model helped to improve accuracy by 3%, a lower gain than that achieved by the time-channel attention mechanism.

We also found that the additional phase information can help to learn attention and WER improved by 4.6% relatively. In comparison with LSTM (Preprocessing 5 noisy-channel), we found that our proposed model achieved comparable performance to beamforming without any preprocessing. Although ALSTM shows a slightly lower performance as compared to LSTM (Preprocessing 5 noisy-channel), a 4.0% relative error rate was obtained by ALSTM (with phase). When we used LSTM-AM with the additional phase features without any attention mechanism, it had a negative influence on learning. Thus, using the phase features for the attention mechanism is more effective than using the phase features as direct inputs of the acoustic model.

We also evaluated the models on the `MULTI` dataset. We found that our system outperformed LSTM (Preprocessing 5 noisy-channel) by 5%, and the gain from the time-channel attention mechanism increased.

We then analyzed the computational aspects of our system. As the multi-microphone is performed as part of the acoustic model computation we have actually found it to be more computationally efficient than performing beamforming followed by an LSTM acoustic model. On our development machine (Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.60GHz), the proposed multi-microphone model with attention and phase operated 0.08 real-time, which was significantly faster than the beamforming followed by acoustic model computation which operated at 0.6 real-time.

4. Conclusions

We proposed an attention-based model (ALSTM) that uses asynchronous and non-stationary inputs from multiple channels to generate outputs. For a distant speech recognition task, we embedded a novel attention mechanism within a RNN-based acoustic model to automatically tune its attention to a more reliable input source. We presented our results on the CHiME3 task and found that ALSTM showed a substantial improvement in WER. Our model achieved comparable performance to beamforming without any prior knowledge of the microphone layout or any explicit preprocessing.

The implications of this work are significant and far-reaching. Our work suggests a way to build a more efficient ASR system by bypassing preprocessing. Our findings suggest that this approach will likely do well on tasks that need to exploit misaligned and non-stationary inputs from multiple sources, such as multimodal problems and sensory fusion. We believe that our attention framework can greatly improve these tasks by maximizing the benefits of using inputs from multiple sources.

5. Acknowledgements

The authors would like to acknowledge Richard M. Stern and William Chan for their valuable and constructive suggestions. This research was supported by LGE.

6. References

[1] D. Van Compernolle, W. Ma *et al.*, "Speech recognition in noisy environments with the aid of microphone arrays," *Speech Communication*, vol. 9, no. 5, pp. 433–442, 1990.

[2] M. L. Seltzer, B. Raj *et al.*, "Likelihood-maximizing beamforming for robust hands-free speech recognition," *Speech and Audio Processing, IEEE Transactions on*, vol. 12, no. 5, pp. 489–498, 2004.

[3] K. Kumatani, J. McDonough *et al.*, "Microphone array processing for distant speech recognition: From close-talking microphones to far-field sensors," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 127–140, 2012.

[4] P. Pertilä and J. Nikunen, "Distant speech separation using predicted time–frequency masks from spatial features," *Speech Communication*, vol. 68, pp. 97–106, 2015.

[5] M. L. Seltzer, "Bridging the gap: Towards a unified framework for hands-free speech recognition using microphone arrays," in *Hands-Free Speech Communication and Microphone Arrays, 2008. HSCMA 2008*. IEEE, 2008, pp. 104–107.

[6] F. Seide, G. Li *et al.*, "Conversational speech transcription using context-dependent deep neural networks." in *Interspeech*, 2011, pp. 437–440.

[7] A.-r. Mohamed, G. E. Dahl *et al.*, "Acoustic modeling using deep belief networks," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 14–22, 2012.

[8] G. Hinton, L. Deng *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[9] Y. Liu, P. Zhang *et al.*, "Using neural network front-ends on far field multiple microphones based speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5542–5546.

[10] S. Renals and P. Swietojanski, "Neural networks for distant speech recognition," in *Hands-free Speech Communication and Microphone Arrays (HSCMA), 2014 4th Joint Workshop on*. IEEE, 2014, pp. 172–176.

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

[12] T. Yoshioka, S. Karita *et al.*, "Far-field speech recognition using cnn-dnn-hmm with convolution in time," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4360–4364.

[13] I. Himawan, P. Motlicek *et al.*, "Learning feature mapping using deep neural network bottleneck features for distant large vocabulary speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, no. EPFL-CONF-207946, 2015.

[14] X. Xiao, S. Watanabe *et al.*, "Deep beamforming networks for multi-channel speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5745–5749.

[15] T. N. Sainath, R. J. Weiss *et al.*, "Speaker location and microphone spacing invariant acoustic modeling from raw multichannel waveforms," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 30–36.

[16] ——, "Factored spatial and spectral multichannel raw waveform cldnns," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5075–5079.

[17] D. Bahdanau, K. Cho *et al.*, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[18] O. Vinyals, L. Kaiser *et al.*, "Grammar as a foreign language," *arXiv preprint arXiv:1412.7449*, 2014.

[19] K. Xu, J. Ba *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," *arXiv preprint arXiv:1502.03044*, 2015.

[20] O. Vinyals and Q. Le, "A neural conversational model," *arXiv preprint arXiv:1506.05869*, 2015.

[21] J. Chorowski, D. Bahdanau *et al.*, "End-to-end continuous speech recognition using attention-based recurrent nn: First results," *arXiv preprint arXiv:1412.1602*, 2014.

[22] W. Chan, N. Jaitly *et al.*, "Listen, attend and spell," *arXiv preprint arXiv:1508.01211*, 2015.

[23] D. Bahdanau, J. Chorowski *et al.*, "End-to-end attention-based large vocabulary speech recognition," *arXiv preprint arXiv:1508.04395*, 2015.

[24] J. Barker, R. Marxer *et al.*, "The third 'chime' speech separation and recognition challenge: Dataset, task and baselines," *Submitted to IEEE 2015 Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2015.

[25] B. Loesch and B. Yang, "Adaptive segmentation and separation of determined convolutive mixtures under dynamic conditions," in *Latent Variable Analysis and Signal Separation*. Springer, 2010, pp. 41–48.

[26] C. Blandin, A. Ozerov *et al.*, "Multi-source tdoa estimation in reverberant audio using angular spectra and clustering," *Signal Processing*, vol. 92, no. 8, pp. 1950–1960, 2012.

[27] X. Mestre, M. Lagunas *et al.*, "On diagonal loading for minimum variance beamformers," in *Signal Processing and Information Technology, 2003. ISSPIT 2003. Proceedings of the 3rd IEEE International Symposium on*. IEEE, 2003, pp. 459–462.

[28] A. Graves, S. Fernández *et al.*, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[29] A. Graves and N. 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.

[30] A. Hannun, C. Case *et al.*, "Deepspeech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[31] N. Morgan and H. Bourlard, "Connectionist speech recognition: a hybrid approach," 1994.

[32] R. M. Stern, E. Gouvêa *et al.*, "Binaural and multiple-microphone signal processing motivated by auditory perception," in *Hands-Free Speech Communication and Microphone Arrays, 2008. HSCMA 2008*. IEEE, 2008, pp. 98–103.

[33] H.-M. Park and R. M. Stern, "Spatial separation of speech signals using amplitude estimation based on interaural comparisons of zero-crossings," *Speech Communication*, vol. 51, no. 1, pp. 15–25, 2009.

[34] C. Kim, K. Kumar *et al.*, "Signal separation for robust speech recognition based on phase difference information obtained in the frequency domain." in *INTERSPEECH*, 2009, pp. 2495–2498.

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

[36] A. Graves, N. Jaitly *et al.*, "Hybrid speech recognition with deep bidirectional lstm," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 273–278.

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

[38] J. Garofalo, D. Graff, D. Paul *et al.*, "Csr-i (wsj0) complete," *Linguistic Data Consortium, Philadelphia*, 2007.

[39] R. Pascanu, T. Mikolov *et al.*, "On the difficulty of training recurrent neural networks," *arXiv preprint arXiv:1211.5063*, 2012.

[40] D. Povey, A. Ghoshal *et al.*, "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.