Recognizing Overlapped Speech in Meetings: A Multichannel Separation Approach Using Neural Networks

Takuya Yoshioka, Hakan Erdogan, Zhuo Chen, Xiong Xiao, and Fil Alleva

Microsoft AI and Research, One Microsoft Way, Redmond, WA, USA

{tayoshio, hakan.erdogan, zhuc, xioxiao, fil}@microsoft.com

Abstract

The goal of this work is to develop a meeting transcription system that can recognize speech even when utterances of different speakers are overlapped. While speech overlaps have been regarded as a major obstacle in accurately transcribing meetings, a traditional beamformer with a single output has been exclusively used because previously proposed speech separation techniques have critical constraints for application to real meetings. This paper proposes a new signal processing module, called an unmixing transducer, and describes its implementation using a windowed BLSTM. The unmixing transducer has a fixed number, say J, of output channels, where J may be different from the number of meeting attendees, and transforms an input multi-channel acoustic signal into J time-synchronous audio streams. Each utterance in the meeting is separated and emitted from one of the output channels. Then, each output signal can be simply fed to a speech recognition back-end for segmentation and transcription. Our meeting transcription system using the unmixing transducer outperforms a system based on a state-of-the-art neural mask-based beamformer by 10.8%. Significant improvements are observed in overlapped segments. To the best of our knowledge, this is the first report that applies overlapped speech recognition to unconstrained real meeting audio.

Index Terms: speech separation, overlapped speech recognition, far-field audio, meeting transcription

1. Introduction

Automatic speech recognition (ASR) technology has made a significant stride over the past decade, achieving human parity in some domains [1,2]. However, when it comes to dealing with speech overlaps, the machines still lag far behind humans. Our brains can attend to one speaker in a noisy multi-talker environment and recognize what he/she has spoken even when his/her voice is overlapped by utterances of other speakers, as demonstrated by the cocktail party effect. By contrast, the current ASR systems fail miserably when utterances of two or more speakers overlap. Computational implementation of the ability of transcribing individual utterances that may or may not be overlapping in the multi-talker settings will be a cornerstone of a range of far-field conversation transcription systems, e.g., for meetings [3–6] and doctor-patient dialogs [7, 8].

In this paper, we develop a multi-microphone meeting transcription system that can recognize overlapped speech. While speech recognition in the meeting space has a long history of research, most systems developed in the past was not able to handle speech overlaps. Overlap segments account for 10+% of the speaking time [9], which is too much to ignore.

Challenges that need to be overcome for the ASR systems to be able to recognize overlapped speech in practical far-field settings include an unknown and varying number of speakers, unknown speaker identities, unknown speech activity segments, the presence of background noise and reverberation, and online operation. Numerous approaches have been proposed for speech separation or overlapped speech recognition, such as independent component analysis, time-frequency bin clustering, and deep neural networks [10–14]. However, previous research in these areas was mostly conducted in *in vitro* settings, e.g., by supposing prior knowledge of the number of meeting attendees. While techniques like overlap detection or speaker counting may help close the gap between the laboratory and practical settings, orchestrating many different error-prone components is not so easy as it appears.

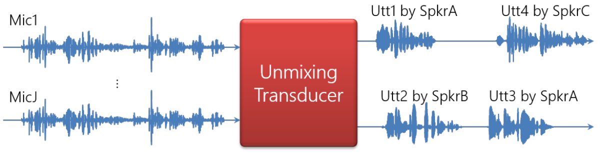

Figure 1: *Unmixing transducer converts an input J-channel signal into a fixed number of time-synchronous audio streams. Each utterance "spurts" from one of the output channels. The ouptut from each channel is fed to an ASR back-end for segmentation and recognition.*

We show that those challenges can be addressed by a novel signal processing module, called an *unmixing transducer*, followed by an array of ASR back-ends. The unmixing transducer continuously receives microphone signals and generates a fixed number of time-synchronous audio streams as illustrated in Fig. 1. The acoustic signal of each utterance found in the input "spurts" from one of the output channels. When the number of active speakers is fewer than that of the outputs, the extra channels generate zero-valued signals. The signal from each output channel is segmented and transcribed by the back-end recognizer connected to that channel.

The unmixing transducer is implemented by extending our recently proposed method [15], which is based on acoustic beamformers driven by a multi-microphone speech separation neural network using permutation invariant training (PIT) [16]. Our extensions include the use of a windowed BLSTM for handling long audio streams, a new model architecture suitable for beamforming, improved feature normalization taking account of phase wrapping, and addition of spherically isotropic random noise to training data. Dereverberation is also performed by using the weighted prediction error (WPE) method [17, 18] to further improve the reverberation robustness.

Our proposed meeting transcription system is shown to work reasonably well for real meeting data that we collected at our speech group meetings. Compared with a state-of-the-art neural mask-based beamformer, the proposed unmixing transducer is demonstrated to be particularly effective in dealing with overlaps.

2. Unmixing Transducer

This section describes what the functionality of the unmixing transducer is and how it is fulfilled in our proposed system. For simplicity and conciseness, we assume the maximum number of overlaps to be two at each time instant, which is true 98+% of the time according to [9]. Extension to more overlaps is straightforward. No assumption is made on the total number of meeting attendees.

2.1. Problem

As shown in Fig. 1, the unmixing transducer receives acoustic signals from a microphone array, in which utterances from different speakers are reverberated and mixed. It separates each utterance from coincident utterances, if any, and emits the separated signal from either of its two output channels. Each utterance should not be broken up and distributed to multiple output channels. For time segments where zero or one speaker is active, the extra output channel yields a zero-valued signal. In this way, it always produces two time-synchronous audio streams.

While the above description may be sufficient, we provide a more formal definition of the problem in the following. We represent signals in the time-frequency domain by denoting time and frequency by t and f, respectively. For each utterance in the meeting, we consider a padded utterance signal, $u_{k,tf}$, where k is the utterance index. Each padded utterance signal is as long as the meeting and is created by taking the time-localized utterance signal as measured by a reference (e.g., the first) microphone and padding the inactive time segments with zero. Now, we consider mapping $\varphi : \{0, \cdots, K-1\} \mapsto \{0,1\}$ with K being the total number of the utterances. This defines which output channel for each utterance to go. The inverse of the mapping can also be defined as $\varphi^{-1}[i] = \{k; \varphi[k] = i\}$, which, for output channel i, returns the set of the utterances that are mapped to i by φ. We call φ a nonmixing mapping when it meets the following condition for all t values[1]:

$$u_{k,tf} \neq 0,\ \exists f \implies u_{k',tf} = 0,\ \forall f,\ \forall k' \in \varphi^{-1}[\varphi[k]] \setminus \{k\}. \quad (1)$$

Nonmixing mappings keep each utterance isolated from each other, ensuring that the following superimposed signal consists of at most one utterance at any time:

$$s_{i,tf} = \sum_{0 \leq k < K,\ \varphi[k]=i} u_{k,tf}, \quad i \in \{0,1\}. \quad (2)$$

We want to find such "unmixed" signals for some nonmixing mapping.

2.2. Masking approach

We start by a simple approach using spectral masking. While the system we eventually develop does not perform masking, the mask estimation processing constitutes an essential element of our system. Let $x_{j,tf}$ and $y_{i,tf}$ denote the jth input to and ith output from the unmixing transducer, respectivey. For each output channel i, spectral mask $m_{i,tf}$, whose value is bounded between 0 and 1, is estimated. The mask is applied to the reference microphone, with index R, to obtain the ith output signal as $y_{i,tf} = m_{i,tf} x_{\mathrm{R},tf}$. We want $y_{i,tf}$ to be close to $s_{i,tf}$ which can be derived with some nonmixing mapping.

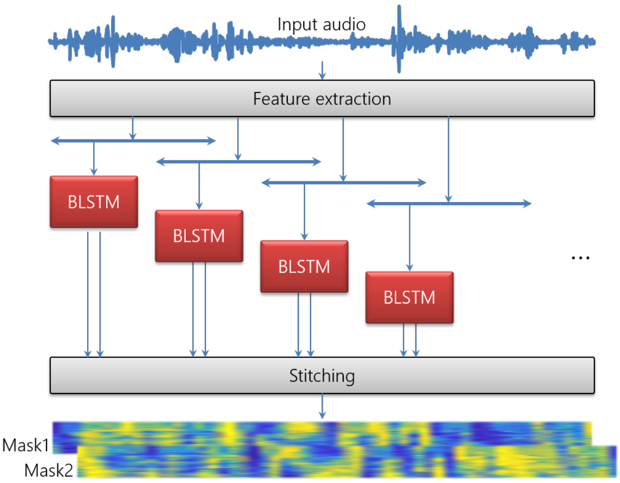

Figure 2: *Mask estimation with windowed BLSTM.*

We propose to calculate the spectral masks with a windowed BLSTM[2] as illustrated in Fig. 2. The incoming audio signals, which may last for tens of minutes to hours in typical office meetings, are broken up into overlapping time windows. Our system uses a 2.4-sec sliding window with a 75% overlap. Feature vectors in each window are fed to a speech separation BLSTM that yields the spectral masks for the respective window. The spectral masks from the adjacent windows are "stitched" to form sequences of spectral masks in a way that does not split an utterance to different output channels (see Sec. 2.2.3 for details).

The windowed BLSTM is chosen for two reasons. First, the model can be trained on a collection of short (i.e., not as long as a typical meeting) speech mixtures, which can be easily created by simulation. Secondly, it can efficiently capture temporal feature dependency, which is critical for the separated speaker signals not to be swapped within each window.

2.2.1. *Input features*

As input to the BLSTM, we make use of both spectral and spatial features. The magnitude spectrum of the reference microphone is used as the spectral features. As regards the spatial features, inter-microphone phase differences (IPDs) relative to the reference microphone are used. All the features are mean-normalized by using a rolling window of four seconds. Unlike in [15], to prevent aliasing at the π/-π boundary, the argument operation is performed after the mean normalization processing. Thus, the IPD features are calculated as

$$\mathrm{Arg}\left(\frac{x_{j,tf}}{x_{\mathrm{R},tf}} - E_\tau \left(\frac{x_{j,\tau f}}{x_{\mathrm{R},\tau f}} \right) \right), \quad j \neq \mathrm{R}, \quad (3)$$

where the time averaging operator, E_τ, is applied over the normalization window.

2.2.2. *Training*

The BLSTM is trained with PIT so that the resultant model can consistently assign each separated utterance to either channel within a window. Our training set comprises simulated multichannel signals of up to 10 seconds. Each signal can be a single utterance or a mixture of two utterances with different lengths,

[1] Condition $u_{k,tf} \neq 0,\ \exists f$ is assumed to be equivalent to the kth utterance being active at frame t. The utterance is regarded as inactive iff $u_{k,tf} = 0,\ \forall f$.

[2] The windowed BLSTM was previously proposed for acoustic modeling [19].

levels, and reverberations, corrupted by background noise. The PIT loss for the lth training sample is defined as

$$\min_{(j_0,j_1)\in\{(0,1),(1,0)\}} \sum_{i=0}^{1}\sum_{tf}\left(m^{(l)}_{i,tf}|x^{(l)}_{\mathrm{R},tf}| - |s^{(l)}_{j_i,tf}|\right)^2, \qquad (4)$$

where $m^{(l)}_{i,tf}$, $x^{(l)}_{\mathrm{R},tf}$, and $s^{(l)}_{i',tf}$ are the ith output from the model, the reference microphone signal, and the i'th source signal as measured at the reference microphone position, respectively. For the training samples involving only one utterance, $s^{(l)}_{i',tf} = 0$ for the extra source. Further details are described in Section 2.4.

2.2.3. Stitching adjacent windows

Because the PIT-trained network has no specific preference as to the ordering of the separated signals, the permutations of the separated signals need to be aligned across the windows at test time. Suppose that the permutations have already been determined up to the previous window. To decide the output signal permutation for the current window, we calculate the cost of each possible permutation and pick the one that provides the lower cost. The cost is defined as the sum of the squared differences between the separated signals of the adjacent windows, where the sum is computed over the overlapping frames.

After the permutation alignment processing, the masks for the nonoverlapping frames of the current window are used. This minimizes the processing latency while being not optimal in terms of accuracy.

2.3. Beamforming approach

While spectral masking provides perceptually enhanced sounds, there is a shared belief that the processing artifacts created by masking are detrimental to the current ASR technology. To overcome this drawback, a mask-based beamforming approach was proposed [20, 21] and showed the state-of-the-art results in far-field ASR tasks [22, 23].

With beamforming, the output signals are computed as

$$y_{i,tf} = \boldsymbol{w}^H_{c,i,f}\boldsymbol{x}_{tf}, \qquad (5)$$

where $\boldsymbol{w}_{c,i,f}$ is a beamformer coefficient vector for output channel i, $\boldsymbol{x}_{tf}$ is a vector stacking the microphone signals, and c is the window index which $y_{i,tf}$ belongs to. By using the MVDR method [24, 25], the optimal beamfomer is obtained as $\boldsymbol{w}_{c,i,f} = \boldsymbol{\Psi}^{-1}_{c,i,f}\boldsymbol{\Phi}_{c,i,f}\boldsymbol{e}/\rho_{c,i,f}$, where the normalization term, $\rho_{c,i,f}$, is calculated as $\rho_{c,i,f} = \mathrm{tr}(\boldsymbol{\Psi}^{-1}_{c,i,f}\boldsymbol{\Phi}_{c,i,f})$. Here, $\boldsymbol{e}$ is the J-dimensional standard basis vector with 1 at the reference microphone position. The two matrices, $\boldsymbol{\Phi}_{c,i,f}$ and $\boldsymbol{\Psi}_{c,i,f}$, represent the spatial covariance matrix of the utterance to be output from the ith channel (which may be referred to as the target utterance) and that of the sounds overlapping the target. These matrices were previously estimated as weighted spatial covariance matrices of the microphone signals, where each microphone signal vector was weighted by $m_{i,tf}$ for the target or $1 - m_{i,tf}$ for the interference [15]. In the following, we propose an improved spatial covariance matrix estimator using a different model architecture.

2.3.1. Speech-speech-noise architecture

The spatial covariance matrix estimator that uses $1 - m_{i,tf}$ as the interference mask is not very accurate. This is because the trained speech separation network cares only about the target signal estimation accuracy, as evident from (4).

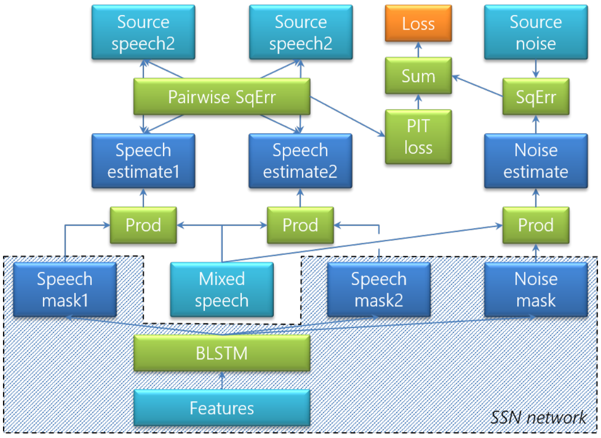

Figure 3: *SSN model and the network for training it.*

A more accurate estimate of the interference spatial covariance matrix can be obtained by explicitly factorizing it to the spatial covariance matrix of the other talker's speech and that of the background noise, $\boldsymbol{\Phi}_{c,\mathrm{N},f}$, as follows:

$$\boldsymbol{\Psi}_{c,i,f} = \boldsymbol{\Phi}_{c,\bar{i},f} + \boldsymbol{\Phi}_{c,\mathrm{N},f}, \qquad (6)$$

where $\bar{i} = 0$ for $i = 1$ and $\bar{i} = 1$ for $i = 0$.

To obtain $\boldsymbol{\Phi}_{c,\mathrm{N},f}$, we add another output channel to the separation network so that the noise masks can also be obtained. Figure 3 shows a diagram of the new model, called the speech-speech-noise (SSN) model[3], and the computational network for training it. As shown in the diagram, we apply the PIT framework only to the first two (i.e., speech) output channels. The loss function is defined as the sum of the PIT loss and the squared error in noise estimation.

The "sig-cov" method of [15] was used when computing the spatial covariance matrices. The gain adjustment technique of [15] was also applied to reduce insertion errors.

2.4. Details

We built an unmixing transducer by using a three-layer 1024-unit BLSTM. Input features were transformed by a 1024-unit projection layer with ReLU nonlinearity before being fed to the BLSTM. On top of the last BLSTM layer, there was a three-head fully connected sigmoid layer, where each head produced spectral masks for either speech or noise. Each of the heads consisted of 257 units, each uniquely associated with a particular frequency bin.

567 hours of speech mixture data were created for training. Source speech signals were taken from WSJ SI-284 and LibriSpeech. Each training sample was created as follows. First, the number of speakers (1 or 2) was randomly chosen. For the two-speaker case, the start and end times of each utterance was randomly determined so that we have a balanced mix of the four configurations described in [15]. The source signals were reverberated with the image method [26], mixed together in the two-speaker case, and corrupted by additive noise. The multichannel additive noise signals were simulated by assuming a spherical isotropic noise field. The generated training samples were clipped to 10 seconds.

Distributed training with 1-bit SGD [27] was performed on 16 GPUs by uisng Microsoft Cognitive Toolkit. The learning rate started from 2.0×10^{-4} and divided by 10 after 150 epochs.

[3] SSN also stands for Speech Separation Network.

The model was saved after each epoch. The model snapshot with the lowest validation loss was picked after convergence.

At test time, two additional tricks were utilized. First, for the two network heads producing the speech masks, we estimated the direction of signal arrivals (DOAs). When the DOA difference was less than 15 degrees, we assumed that there were actually only one speaker and thus merged the masks while zeroing out the masks of the less significant head. Secondly, for each time frequency bin, the three masks were normalized to sum to one.

3. Meeting Transcription Experiments

3.1. System build

We developed a meeting transcription system with a seven-channel circular array by using the unmixing transduer described above. The system consists of three kinds of modules, each performing dereverberation, speech separation, or ASR. The dereverberation module estimates a multi-input multi-output dereverberation filter for converting a seven-channel microphone array signal to a less reverberant one with seven channels [17]. The dereverberation filter was updated every second. Then, the unmixing transducer transforms the dereverberated seven-channel audio into two-channel separated speech streams. The model was built as per Section 2.4. Each output signal from the unmixing transducer was provided to an ASR back-end that performs segmentation and recognition.

For ASR, we trained an acoustic model on ∼7K hours of spontaneous speech audio, which were collected from various sources, both public (e.g., Switchboard and Fisher) and private (e.g., Microsoft Research lecture talks). The audio quality was not consistent due to the effects of noise, channel, and so on, which seemed to improve the robustness of the acoustic model. The model input was 40-channel mel filterbank energies compressed with 10th-root nonlinearity. The model consisted of four 1024-unit LSTM layers. It was trained with a cross entropy criterion, followed by sequence training. Decoding was performed with a dictionary of ∼240K words and our internal trigram language model built for conversational tasks.

3.2. Task

To evaluate the meeting transcription system, six meetings were recorded at our speech group and professionally transcribed. The meetings took place in several different rooms and lasted for 30 minutes to an hour. The recordings were made with both headset microphones and a seven-channel circular microphone array. They were manually segmented and transcribed. The transcribers were allowed to access both types of microphones. The average overlap rate was 14.7%, which was twice as high as that of AMI [9]. While the overlap rate value varies depending on the annotation policy, based on an informal inspection, we feel our meetings had noticeably more overlaps than those of AMI. The more frequent overlaps may be attributed to the fact that the participants in our meetings knew each other well and were discussing work-related topics.

The system outputs were scored with asclite tool [28], which aligns multiple hypotheses against multiple references. Tighter reference segmentations were used when calculating a word error rate (WER) for single-speaker segments so as not to discard segments that had overlaps only in nonspeech frames. Note that this might have resulted in a slight WER overestimation because asclite makes use of the reference time stamps to find segments for alignment.

Table 1: *%WER of different front-ends.*

System	Overlapped segments	
	Included	Excluded
No processing (mic0)	44.6	40.9
Dereverb. [17]	42.1	38.7
+BeamformIt [29]	43.2	40.6
+MaskBF [23]	37.9	32.8
+Unmix. Trans. (proposed)	**33.8**	**30.4**
+UT trained only on WSJMix	34.2	30.8
+UT without noise channel	36.8	34.5

3.3. Results

Table 1 lists the WERs obtained with different front-ends including our system. Without microphone array processing, the WER was 44.6%. Dereverberation improved the recognition accuracy by 5.6% relative. BeamformIt [29], the default beamformer used by Kaldi's recipe for AMI [30], provided no improvement. A neural mask-based beamformer, which yielded state-of-the-art results in both CHiME-3/4 [22] and more practical large vocabulary settings [23], was also examined. Here, we used the best model we obtained in [23] and applied it to our meeting data by using a 2.4-sec sliding window. This improved the performance by 10.0%, achieving a WER of 37.9%. However, this beamformer provided a 15.2% gain for single-speaker segments, indicating that this method was not effective at handling overlaps. Nevertheless, we used this single-output beamformer as our baseline because no speech separation method was previously applied to meetings with unknown and varying numbers of attendees.

The proposed system achieved a WER of 33.8%, outperforming the strong baseline system by 10.8%. It is noteworthy that the gain was 7.3% when the overlapped segments were excluded. This means that, while the proposed approach provided a modest improvement even for single-speaker segments, its advantage was prominent in overlapped segments. When the unmixing transducer was trained only on the WSJ-derived data, which amounted to 219 hours, the recognition accuracy slightly deteriorated. When the network estimated only speech masks, i.e., when the SSN architecture was not used, the recognition performance was degraded to 36.8%. The degradation was profound especially in single-speaker segments, indicating the importance of explicitly estimating the noise masks. Overall, the proposed meeting transcription system, comprising the dereverberator, unmixing transducer, and ASR back-ends, improved the WER by 24.2% compared with the single distant microphone system.

4. Conclusion

In this paper, we described a meeting transcription system that can handle speech overlaps. The system is based on the unmixing transducer, a novel signal processing module for converting multi-channel audio signals into a fixed number of separated speech streams. We implemented it by using a windowed BLSTM. The SSN architecture was proposed to effectively leverage beamforming capability. Significant gains in meeting transcription performance were obtained compared with a strong neural mask-based beamformer. Further results on both public and private data will be reported in a follow-up paper.

As far as we know, this is the first overlapped speech recognition system that has been demonstrated to work for actual unconstrained meetings. We believe the proposed approach is promising and anticipate further investigation in this direction.

5. References

[1] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *CoRR*, vol. abs/1610.05256, 2016. [Online]. Available: http://arxiv.org/abs/1610.05256

[2] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, J. Chen, M. Chrzanowski, A. Coates, G. Diamos, E. Elsen, J. Engel, L. Fan, C. Fougner, T. Han, A. Y. Hannun, B. Jun, P. LeGresley, L. Lin, S. Narang, A. Y. Ng, S. Ozair, R. Prenger, J. Raiman, S. Satheesh, D. Seetapun, S. Sengupta, Y. Wang, Z. Wang, C. Wang, B. Xiao, D. Yogatama, J. Zhan, and Z. Zhu, "Deep speech 2: End-to-end speech recognition in english and mandarin," *CoRR*, vol. abs/1512.02595, 2015. [Online]. Available: http://arxiv.org/abs/1512.02595

[3] J. G. Fiscus, N. Radde, J. S. Garofolo, J. A. A. Le, and C. Laprun, "The rich transcription 2005 spring meeting recogntion evaluation," in *Proc. Machine Learning, Multimodal Interaction Worksh.*, 2005, pp. 369–389.

[4] T. Hain, L. Burget, J. Dines, P. N. Garner, F. Grézl, A. El Hannani, M. Huijbregts, M. Karafiát, M. Lincoln, and V. Wan, "Transcribing meetings with the AMIDA systems," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 2, pp. 486–498, 2012.

[5] T. Hori, S. Araki, T. Yoshioka, M. Fujimoto, S. Watanabe, T. Oba, A. Ogawa, K. Otsuka, D. Mikami, K. Kinoshita, T. Nakatani, A. Nakamura, and J. Yamato, "Low-latency real-time meeting recognition and understanding using distant microphones and omni-directional camera," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 2, pp. 499–513, 2011.

[6] S. Renals and P. Swietojanski, *Distant Speech Recognition Experiments Using the AMI Corpus*. Springer International Publishing, 2017, pp. 355–368.

[7] E. Edwards, W. Salloum, G. P. Finley, J. Fone, G. Cardiff, M. Miller, and D. Suendermann-Oeft, "Medical speech recognition: Reaching parity with humans," in *Speech, Computer*, 2017, pp. 512–524.

[8] C. Chiu, A. Tripathi, K. Chou, C. Co, N. Jaitly, D. Jaunzeikare, A. Kannan, P. Nguyen, H. Sak, A. Sankar, J. Tansuwan, N. Wan, Y. Wu, and X. Zhang, "Speech recognition for medical conversations," *CoRR*, vol. abs/1711.07274, 2017. [Online]. Available: http://arxiv.org/abs/1711.07274

[9] O. Çetin and E. Shriberg, "Analysis of overlaps in meetings by dialog factors, hot spots, speakers, and collection site: Insights for automatic speech recognition," in *Proc. Interspeech*, 2006, pp. 293–296.

[10] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: discriminative embeddings for segmentation and separation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 31–35.

[11] L. Drude and R. Haeb-Umbach, "Tight integration of spatial and spectral features for BSS with deep clustering embeddings," in *Proc. Interspeech*, 2017, pp. 2650–2654.

[12] Z. Chen, J. Droppo, J. Li, and W. Xiong, "Progressive joint modeling in unsupervised single-channel overlapped speech recognition," *CoRR*, vol. abs/1707.07048, 2017. [Online]. Available: http://arxiv.org/abs/1707.07048

[13] X. Zhang and D. Wang, "Binaural reverberant speech separation based on deep neural networks," in *Proc. Interspeech*, 2017, pp. 2018–2022.

[14] K. Žmolíková, M. Delcroix, K. Kinoshita, T. Higuchi, A. Ogawa, and T. Nakatani, "Learning speaker representation for neural network based multichannel speaker extraction," in *Proc. Worksh. Automat. Speech Recognition, Understanding*, 2017, pp. 8–15.

[15] T. Yoshioka, H. Erdogan, Z. Chen, and F. Alleva, "Multi-microphone neural speech separation for far-field multi-talker speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, to appear.

[16] M. Kolbæk, D. Yu, Z. H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 10, pp. 1901–1913, 2017.

[17] T. Yoshioka and T. Nakatani, "Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 10, pp. 2707–2720, 2012.

[18] B. Li, T. N. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Punduk, K. Chin, K. C. Sim, R. J. Weiss, K. W. Wilson, E. Variani, C. Kim, O. Siohan, M. Weintrauba, E. McDermott, R. Rose, and M. Shannon, "Acoustic modeling for Google Home," in *Proc. Interspeech*, 2017.

[19] A. Mohamed, F. Seide, D. Yu, J. Droppo, A. Stolcke, G. Zweig, and G. Penn, "Deep bi-directional recurrent networks over spectral windows," in *Proc. Worksh. Automat. Speech Recognition, Understanding*, 2015, pp. 78–83.

[20] T. Yoshioka, N. Ito, M. Delcroix, A. Ogawa, K. Kinoshita, M. Fujimoto, C. Yu, W. J. Fabian, M. Espi, T. Higuchi, S. Araki, and T. Nakatani, "The NTT CHiME-3 system: advances in speech enhancement and recognition for mobile multi-microphone devices," in *Proc. Worksh. Automat. Speech Recognition, Understanding*, 2015, pp. 436–443.

[21] J. Heymann, L. Drude, A. Chinaev, and R. Haeb-Umbach, "BLSTM supported GEV beamformer front-end for the 3rd CHiME challenge," in *Proc. Worksh. Automat. Speech Recognition, Understanding*, 2015, pp. 444–451.

[22] J. Heymann, L. Drude, C. Boeddeker, P. Hanebrink, and R. Haeb-Umbach, "BeamNet: End-to-end training of a beamformer-supported multi-channel ASR system," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 5325–5329.

[23] C. Boeddeker, H. Erdogan, T. Yoshioka, and R. Haeb-Umbach, "Exploring practical aspects of neural mask-based beamforming for far-field speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, accepted.

[24] M. Souden, J. Benesty, and S. Affes, "On optimal frequency-domain multichannel linear filtering for noise reduction," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 2, pp. 260–276, 2007.

[25] H. Erdogan, J. R. Hershey, S. Watanabe, M. Mandel, and J. Le Roux, "Improved MVDR beamforming using single-channel mask prediction networks," in *Proc. Interspeech*, 2016, pp. 1981–1985.

[26] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Am.*, vol. 65, no. 4, pp. 943–950, 1979.

[27] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and application to data-parallel distributed training of speech DNNs," in *Interspeech*, 2014, pp. 1058–1062.

[28] J. G. Fiscus, J. Ajot, N. Raddle, and C. Laprum, "Multiple dimension Levenshtein edit distance calculations for evaluating automatic speech recognition systems during simulaneous speech," in *Proc. Int. Conf. Language Resources, Evaluation*, 2006, pp. 803–808.

[29] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 7, pp. 2011–2022, 2007.

[30] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *Proc. Autom. Speech Recog. Understand.*, 2011.