Anchored Speech Detection

Roland Maas, Sree Hari Krishnan Parthasarathi, Brian King, Ruitong Huang
Björn Hoffmeister

Amazon.com, USA.
{rmaas,sparta,bbking,bjornh}@amazon.com, ruitong@cs.ualberta.ca

Abstract

We propose two new methods of speech detection in the context of voice-controlled far-field appliances. While conventional detection methods are designed to differentiate between speech and nonspeech, we aim at distinguishing *desired speech*, which we define as speech originating from the person interacting with the device, from background noise and interfering talkers. Our two proposed methods use the first word spoken by the desired talker, the "anchor" word, as a reference to learn characteristics about that speaker. In the first method, we estimate the mean of the anchor word segment and subtract it from the subsequent feature vectors. In the second, we use an encoder-decoder network with features that are normalized by applying conventional log amplitude causal mean subtraction. The experimental results reveal that both techniques achieve around 10% relative reduction in frame classification error rate over a baseline feed-forward network with conventionally normalized features.

Index Terms: speech detection, voice activity detection, encoder decoder neural network.

1. Introduction

Speech detection has many applications, including pre-processing for automatic speech recognition (ASR) [1], speaker recognition [2], speaker change detection [3, 4], speaker diarization [5], end-pointing [6, 7], and manual transcription [8]. Major efforts have been made to hand-craft suitable feature representations for speech-nonspeech detection, such as zero-crossing rate [9], periodic-aperiodic ratio [10], autocorrelation-based voicing features [11] and more [12]. Recently, deep learning approaches have also yielded promising performance [13, 14].

In this paper we specifically investigate the task of speech detection in the context of voice-controlled far-field appliances. While conventional speech detection methods are designed to differentiate between speech and nonspeech, we aim at distinguishing *desired speech*, which we define as speech originating from the person interacting with the device, from background noise and interfering speakers. We assume that the first word is spoken by the desired speaker, and that this *anchor word* can be employed for learning the desired speaker's representation. Consider the following interaction: "[speaker 1:] Computer, what time is it? [speaker 2:] Close the door!". In this example, we consider "computer" as the anchor word to wake up the device, the utterance by speaker 1 as desired speech, and the utterance by speaker 2 as interfering speech. The aim of this paper is hence to detect desired speech exploiting that the first word originates from the desired speaker. We term this approach *anchored speech detection* (ASD). In this contribution, we focus on frame classification accuracy as final metric. However, ASD can be employed to improve the performance of other ASR system components, as the ones listed before.

In addition to speech-nonspeech detection, ASD has similarities to speaker change detection [15], speaker diarization, and speaker linking tasks [16]; however, these methods are typically unsupervised or "lightly" supervised, focusing on longer segments like meetings or broadcast news. ASD, in contrast, can employ supervised learning methods, specifically aimed at short device-directed utterances. Furthermore, the present task has constraints on latency and computational time.

The challenge with ASD task is to learn speaker representations from a short speech segment (i.e. the first word - typically 300ms of speech). ASR and speaker recognition communities have proposed a number of signal and model-based methods for learning speaker representations, including maximum likelihood linear regression (MLLR) [17], i-vectors [18], vocal tract length normalization (VTLN) [19], and mean-variance normalization [20]. Recently, in the machine learning community, there has been considerable interest in encoder-decoder networks for their ability to represent variable length sequences into fixed-length vectors in a fully data-driven fashion [21].

Drawing motivation from these techniques, we propose two methods to encode the anchor word's information for use in desired speech detection. In the first proposed method, we estimate the mean of the anchor word segment in the log-filterbank energy (LFBE) domain, then subtract it from all subsequent feature vectors of the same utterance in order to expose differences in the low frequency components (such as the energy level) relative to the anchor word segment. We call this approach anchored mean subtraction (AS). The normalized features are then classified using a feed-forward deep neural network (DNN). In the second proposed method, we use an encoder-decoder network with LFBE features that are normalized by applying conventional causal mean subtraction (MS). Here, the encoder network provides an embedding of the anchor word segment that is then fed into the decoder DNN together with the feature vector to be classified.

The proposed methods can be implemented in an online system where minimal latency is key. The experimental results reveal that both proposed techniques achieve a 10% relative reduction in frame classification error rate over a baseline feed-forward DNN with MS-normalized LFBE features. The results also suggest that the encoder-decoder topology can be seen as a data-driven generalization of acoustic feature normalization.

The remainder of this paper is structured as follows: In Section 2, the concept of feature mean subtraction is reviewed and AS is introduced. In Section 3, encoder-decoder neural networks are reviewed and applied to acoustic feature normalization. Finally, experimental results are presented in Section 4 and conclusions are drawn in Section 5.

2. Mean Subtraction Approach

For a far-field speech detection system to work well, it must operate in an acoustic environment that has immense variability, including different speaker characteristics such as identity, gender, position and volume, room characteristics such as room size and the location of the microphone, and noise characteristics such as volume, spectral, and modulation characteristics. There has been a significant amount of work in designing speech systems to be more robust to such conditions [22, 23]. One such technique is cepstral mean subtraction [20]. Cepstral coefficients are created by computing the short-time Fourier transform (STFT) of the time-domain audio signal, combining the filterbank energies into a mel-spaced filterbank, calculating the logarithm of the coefficients, and then transforming them with a discrete cosine transform (DCT). The features we use for our speech detection system are log filterbank energies, or LFBE's, which follow the same processing chain without the final DCT.

2.1. Overview of Mean Subtraction

Mean subtraction helps normalize features: it is particularly well-suited to normalizing the speech transfer function characteristics. A popular model for a far-field speech signal is $x(t) = s(t) * h(t)$, where $x(t)$, $s(t)$, and $h(t)$ are the time-domain far-field recorded signal, speech signal, and transfer function, respectively. With a stationary transfer function, an estimate of the speech signal in the log spectral domain can be retrieved by $\log(S_{k,n}) \approx \log(X_{k,n}) - \log H_k$ with k being the frequency bin, n the frame index, and $S_{k,n}$, $X_{k,n}$, H_k the respective STFT magnitudes of $s(t)$, $x(t)$, and $h(t)$. The transfer function can be estimated in offline and online fashions. In the offline method, the per-feature mean is first calculated over the desired speech segment ($\sum_{n=1}^{N} X_{k,n}$). Then the per-feature means are subtracted from the original features.

The above system works well in environments where the speech and noise characteristics are relatively stationary throughout the analyzed segment. In online system or more dynamic acoustic environments, the mean statistics are instead continually updated over time. One popular choice is to update the time-varying mean estimation using an autoregressive/recursive update [24], which we refer to as causal mean subtraction (MS):

$$\hat{H}_{k,n+1} = \alpha \hat{H}_{k,n} + (1-\alpha) X_{k,n} \text{ for } 0 < \alpha \leq 1, \quad (1)$$

where $\hat{H}_{k,n}$ denotes the estimate of H_k and at frame n. The parameter α is chosen to allow the estimator to capture the static or slowly-changing environmental characteristics without capturing the faster-moving speech characteristics. This estimator can be interpreted as a low-pass filter. As such, it distinguishes what parts of the signal to attenuate according to the features' modulation, or rate of change over time. Thus, the estimator will capture all slowly changing characteristics of the signal, which include both environmental and speaker characteristics. For example, consider two recordings of the same speaker, with one being louder than the other. The mean subtraction would remove the level difference irrespectively of whether it is due to the person talking louder (speaker characteristic change) or the person being closer to the microphone (environmental characteristic change). Thus, normalization in the log amplitude frequency domain can normalize out both low-frequency environmental and low-frequency speaker characteristics.

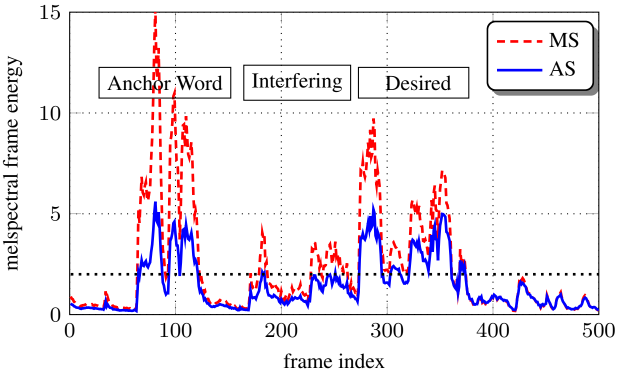

Figure 1: *Comparison of frame energy levels after causal mean subtraction (MS) and anchored mean subtraction (AS) for a recording containing interfering speech at lower volume than the desired speech. For MS, notice how the desired and interfering speech peaks can be of the similar height (e.g., around frames 310 and 370 vs. frames 180 and 250). In contrast, for AS, the energy levels are separable (as indicated by the dotted horizontal line).*

2.2. Applying mean subtraction to ASD

Causal mean subtraction can transform desired and interfering speech features to look more similar, which is in opposition to our goal of desired speech detection. For example, in the case of an anchor word followed by interfering speech (at lower volume) and desired speech, the causal mean subtraction causes energy peaks in interfering and desired speech to be of similar height (Figure 1).

In this paper, we propose the anchored mean subtraction (AS) method to keep for better distinguishing features between the desired and interfering speech. For our mean estimator, we compute the average feature values over the anchor word segment only and keep it constant for the remaining utterance. In a traditional speech detection system, where we want to detect all speech, this model may be too constrained to detect speech from multiple talkers. In our task, however, we use the anchor word as an example of the desired talker's speech, and then by applying AS, we shift the features corresponding to our desired speaker closer to being zero-mean. This allows us to train a classifier, a DNN, to better detect a desired talker's speech. Figure 1 illustrates how AS can, for example, preserve energy level differences. AS allows the features to be normalized in a dynamic fashion for each utterance because the mean subtraction is always estimated for each new anchor word.

2.3. Interpreting AS in relation to other methods

In the cepstral domain, AS can be interpreted as a special case of i-vectors, with the universal background model (UBM) and the total variability matrix set to $N(0, \mathbb{I})$ and $\mathbb{I}$ respectively [25]. Furthermore it can also be seen as feature-space MLLR, with only the bias being estimated [26].

There are connections between mean subtraction to log-likelihood based methods [3]: when two speaker classes are represented as multivariate Gaussians with the same covariance matrix, specifically as $N(\mu_1, \Sigma)$ and $N(\mu_2, \Sigma)$ respectively, the decision boundary can be shown to be $\Sigma^{-1}(\mu_1 - \mu_2)$. Shifting the coordinates to the mean of the first class, the decision boundary can be seen to be directly dependent on the mean of the second class (i.e. the desired speaker).

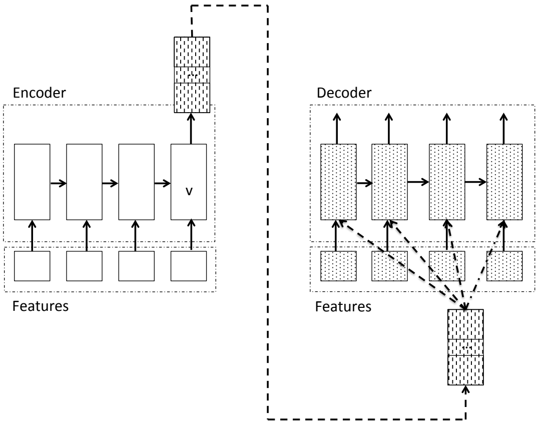

Figure 2: *Encoder-Decoder architecture for anchored speech detection. The anchor word segment is fed into the encoder. Only its last output becomes "visible" (v) as it is appended to the feature vector (window) that is to be classified by the decoder. The decoder has two output nodes: class '0' (non-speech or interfering speech frame) and '1' (desired speech frame).*

3. Encoder-Decoder Approach

Beginning with an overview of recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), this section describes the Encoder-Decoder model. It then discusses the connection between causal mean subtraction and the encoder-decoder model.

3.1. Overview of the Encoder-Decoder model

RNNs are computational graphs consisting of affine transforms and non-linear activation functions, the latter being typically sigmoid for the hidden layers and softmax for the output layer. The activations at a forward recurrent unit can be written as:

$$a_{h,t} = W_h^f y_{h-1,t}^f + W_h^r y_{h,t-1}^r \qquad (2)$$

$$y_{h,t} = \phi(a_{h,t}) \qquad (3)$$

where $\phi(.)$ correspond to activation functions and W_h^f, W_h^r denote weights; $y_{.,.}$ are outputs at hidden layers; $a_{.,.}$ denote activations before nonlinearities.

RNN described above can be extended in a number of ways: (i) LSTMs consist of one or more memory cells per recurrent layer and three gating operations [27]; (ii) extension from unidirectional RNNs to bidirectional RNNs. Recent work explore novel recurrent neural architectures: a split network architecture, with an encoder and a decoder network [21, 28]. The encoder is an RNN/LSTM computing a fixed-length representation from a variable-length input sequence, while the decoder is typically another recurrent network that employs this representation towards generating a variable-length target sequence.

Optimization of the encoder and the decoder network parameters is done jointly: the parameters of the decoder are updated every frame through the standard backpropagation (BP) algorithm, while the encoder's parameters are updated once per sequence, again through BP. The error signal for the encoder is obtained by accumulating the error signals per frame from the decoder.

Table 1: *Comparison of data set length and number of frames. Both interfering and nonspeech frames are aligned to target '0' while desired speech frames are aligned to target '1'.*

data set	hours	interfering speech/ nonspeech frames	desired speech frames
dev	3.5h	360k / 460k	445k
test	4h	416k / 518k	500k
train	28h	2.9M / 3.6M	3.6M

3.2. Adapting the Encoder-Decoder model

The standard encoder-decoder model is adapted for ASD task: a typical encoder network consumes a variable length sequence which is used by the decoder to produce another variable length sequence [21]. However, in ASD, the goal is to employ a representation of the anchor word towards a frame-level classification. Figure 2 depicts this adapted model, where the anchor word segment is fed into the encoder whose output is then appended to the acoustic feature vector (window) that is to be classified by the decoder. While the decoder is usually a recurrent network, for ASD, we also explore a feed-forward network. The encoder itself has an LSTM layer. The parameters are jointly optimized by minimizing the cross-entropy training criterion (with mini-batch stochastic gradient descent).

3.3. Encoder-Decoder as generalization of AS

Section 2 described the interpretation of AS as an autoregressive update. This can be constructed as a special case of the encoder-decoder network, with the following changes (with no parameter updates): (a) With an identity activation $\phi = \mathbb{I}$. (b) With forward and recurrent weight matrices set to $\alpha \cdot \mathbb{I}$ and $(1 - \alpha) \cdot \mathbb{I}$ respectively. (c)The corresponding biases set to 0.

Furthermore AS is an unsupervised estimate obtained from such a linear encoding network, while the encoder-decoder provides a supervised estimate of a nonlinear encoding network.

4. Experiments

In this section, we evaluate the frame classification accuracy of the proposed speech detection approaches for different types of feature normalization and neural network topologies.

4.1. Experimental Setup

As dataset, we use real recordings of natural human interactions with voice-controlled domestic far-field appliances. These recordings are not controlled in any way and hence contain desired speech, interfering speech, and nonspeech segments but may also contain overlapping speech, multiple talkers, background noise, etc. at any position in the utterance. Every utterance contains an anchor word segment with boundaries that are assumed to be known for both training and testing. In practice, these boundaries can, e.g., be obtained by the keyword spotter that is used to detect the device wake-word. Furthermore, all utterances start with the same anchor word. The fact the anchor word is the same for all utterances is not a conceptual requirement of proposed methods but a characteristic of the employed dataset. The datasets' length and number of desired, interfering, and nonspeech frames are depicted in Table 1. Both nonspeech and interfering speech frames are aligned to target '0' while de-

Table 2: *Comparison of binary classification error rates in % for classifying nonspeech and interfering speech frames (class 0) vs. desired speech frames (class 1) using different DNN topologies and raw LFBEs, LFBEs normalized with causal mean subtraction (MS), and anchored mean subtraction (AS). The decision thresholds on the DNN posteriors are optimized for lowest error rate on the dev set. The ROC curves for the networks in rows 1 and 3 with LFBE+MS and LFBE+AS features are depicted in Figure 3.*

Encoder	Decoder	raw LFBE	LFBE +MS	LFBE +AS
None	FF	19.4	17.2	**15.4**
None	RNN	19.5	17.3	15.5
LSTM	FF	**15.7**	**15.2**	**15.2**
LSTM	RNN	15.8	15.4	15.6

sired speech frames are aligned to target '1'. The ground-truth labels are obtained by transcribing the desired speech and interfering speech and performing forced alignment using an ASR acoustic model. Note that in case of desired speech overlapping with interfering speech, the forced alignment is run against the transcription of the desired speech and the according frames are hence aligned to target '1'.

The following feature types are considered. Global mean and variance normalization (estimated on the training set) is applied to all features before any per-utterance normalization:

- Raw LFBE: LFBE features without per-utterance normalization,
- LFBE+MS: LFBE features normalized with causal mean subtraction [24],
- LFBE+AS: LFBE features, normalized by the mean value estimated over the anchor word segment.

As a classifier (referred to as decoder), we employed a feedforward (FF) network consisting of 3 hidden layers with sigmoid nonlinearities, 250 neurons each and a ± 8 input frame context. We also used an RNN decoder, where the first layer of the FF network is replaced by a fully recurrent sigmoid layer. The encoder network, when used, consists of one LSTM layer with 90 units and output size of 90, one cell per unit, and a ± 8 input frame context. The DNNs are trained using our in-house system [29] with conventional cross-entropy criterion and stochastic gradient descent. We denote the different topologies by concatenating the employed network types. For example, the term *LSTM-FF* refers to *LSTM-encoder and FF-decoder*. With *FF-only*, we denote *FF-decoder without encoder*.

4.2. Experimental Results

Table 2 summarizes the frame classification error rates for the different feature types and DNN topologies. It can be seen that the error rate of the FF-only network (row 1) strongly depends on the employed feature normalization method as it ranges from 19.4% (for raw LFBEs) to 15.4% (for AS features). In contrast, the performance of the LSTM-FF network (row 3) is evidenced to be much more robust with error rates ranging from 15.7% (for raw LFBEs) to 15.2% (for MS and AS types). We can also observe that the error rate reduction of the LSTM-FF over the FF-only network is highest for raw LFBEs (19% relative) and lowest for LFBE+AS (1.3% relative). We furthermore note that both the LSTM-FF network (for MS and AS features) as well as the FF-only network (for AS features) achieve a relative error rate reduction of more than 10% over the FF-only network with conventional MS features. To confirm these findings, the ROC curves for the LFBE+MS and LFBE+AS features are depicted in Figure 3. For the former feature type, it can be seen that the LSTM-FF network consistently outperforms the FF-only network. For LFBE+AS features, we observe that the LSTM-FF slightly outperforms the FF-only network, especially at lower false alarm rates. Table 2 also shows that changing the decoder type from FF to RNN does not significantly affect the frame error rates. Our internal experiments also confirmed that increasing the decoder size and/or memory (by, e.g., using LSTMs instead of RNNs) did not yield any improvements.

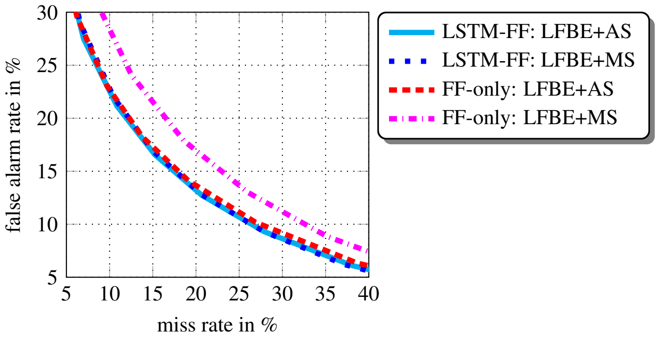

Figure 3: *ROC curve for the networks in rows 1 and 3 of Table 2 with LFBE+MS and LFBE+AS features.*

The experimental results allow for two fundamental conclusions: Firstly, the difference in low frequency components (i.e., those captured by mean subtraction) relative to the anchor word segment is indeed an important cue for detecting desired speech frames. Secondly, the encoder-decoder networks appear to implicitly learn a normalization fingerprint from the anchor word segment — almost independently of the employed feature type. From a more generalized perspective, these results could be interpreted to that the encoder-decoder method represents a data-driven approach of acoustic feature normalization.

5. Conclusions

We presented two approaches for detecting desired speech, i.e., speech originating from the person interacting with a voice-controlled far-field device. The first approach was to normalize the feature vectors using the mean estimated on the anchor word segment only in order to expose differences in the low frequency components relative to the anchor word segment. The second approach employed an encoder-decoder neural network for learning an anchor word embedding. The experimental results evidenced that both techniques achieve a 10% relative reduction in frame classification error rate over a baseline feedforward network with conventionally normalized features. We furthermore showed that the performance of encoder-decoder approach is almost independent of the employed feature normalization technique, which suggests that it can be seen as data-driven acoustic feature normalization.

6. Acknowledgements

The authors thank S. Matsoukas, N. Strom, R. Thomas, L. Mathias, A. Rastrow, T. Drugman, S. Wiesler, and A. Khare for their feedback on the manuscript.

7. References

[1] T. Pfau, D. P. Ellis, and A. Stolcke, "Multispeaker speech activity detection for the ICSI meeting recorder," in *Proceedings IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2001, pp. 107–110.

[2] D. A. Reynolds, "An overview of automatic speaker recognition technology," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 4, 2002, pp. IV–4072–IV–4075.

[3] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[4] D. Lilt and F. Kubala, "Online speaker clustering," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2004.

[5] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1557–1565, 2006.

[6] W.-H. Shin, B.-S. Lee, Y.-K. Lee, and J.-S. Lee, "Speech/non-speech classification using multiple features for robust endpoint detection," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3, 2000, pp. 1399–1402.

[7] B. Liu, B. Hoffmeister, and A. Rastrow, "Accurate endpointing with expected pause duration," in *Proceedings of Interspeech*, 2015.

[8] N. Morgan, D. Baron, J. Edwards, D. P. W. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke, "The meeting project at ICSI," in *Proceedings of Human Language Technology Research*, 2001, pp. 246–252.

[9] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 7, pp. 504–516, 2002.

[10] M. Fujimoto, K. Ishizuka, and T. Nakatani, "Study of integration of statistical model-based voice activity detection and noise suppression." in *Proceedings of Interspeech*, 2008, pp. 2008–2011.

[11] T. T. Kristjansson, S. Deligne, and P. A. Olsen, "Voicing features for robust speech detection," in *Proceedings of Interspeech*, 2005, pp. 369–372.

[12] S. O. Sadjadi and J. H. L. Hansen, "Unsupervised speech activity detection using voicing measures and perceptual spectral flux," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 197–200, 2013.

[13] N. Ryant, M. Liberman, and J. Yuan, "Speech activity detection on youtube using deep neural networks." in *Proceedings Interspeech*, 2013, pp. 728–731.

[14] Xiao-Lei Zhang and Ji Wu, "Deep belief networks based voice activity detection," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 697–710, 2013.

[15] S. H. K. Parthasarathi, M. Magimai-Doss, D. Gatica-Perez, and H. Bourlard, "Speaker change detection with privacy-preserving audio cues," in *Proceedings of International Conference on Multimodal Interfaces*, 2009.

[16] D. A. Van Leeuwen, "Speaker linking in large data sets," 2010.

[17] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Language*, vol. 10, pp. 249–264, 1996.

[18] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 788–798, 2011.

[19] L. Lee and R. Rose, "A frequency warping approach to speaker normalization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 6, pp. 49 – 60, 1998.

[20] F.-H. Liu, R. M. Stern, X. Huang, and A. Acero, "Efficient cepstral normalization for robust speech recognition," in *Proceedings of the Workshop on Human Language Technology*, 1993, pp. 69–74.

[21] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proceedings of Advances in Neural Information Processing Systems*, 2014, pp. 3104–3112.

[22] J. Barker, E. Vincent, N. Ma, H. Christensen, and P. Green, "The PASCAL CHiME speech separation and recognition challenge," *Computer Speech & Language*, vol. 27, no. 3, pp. 621–633, 2013.

[23] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An Overview of Noise-Robust Automatic Speech Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 745–777, Apr. 2014.

[24] S. Tibrewala and H. Hermansky, "Multi-band and adaptation approaches to robust speech recognition." in *Proceedings Eurospeech*, 1997.

[25] S. Garimella, A. Mandal, N. Strom, B. Hoffmeister, S. Matsoukas, and S. H. K. Parthasarathi, "Robust i-vector based adaptation of DNN acoustic model for speech recognition," in *Proceedings Interspeech*, 2015.

[26] S. H. K. Parthasarathi, B. Hoffmeister, S. Matsoukas, A. Mandal, N. Strom, and S. Garimella, "fMLLR based feature-space speaker adaptation of DNN acoustic models," in *Proceedings Interspeech*, 2015.

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

[28] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv 1406.1078*, 2014.

[29] N. Strom, "Scalable distributed DNN training using commodity GPU cloud computing," in *Proceedings of Interspeech*, 2015.