INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

The I2R's ASR System for the VOiCES from a Distance Challenge 2019

Tze Yuang Chong, Kye Min Tan, Kah Kuan Teh, Changhuai You, Hanwu Sun, Huy Dat Tran

Institute for Infocomm Research, Singapore 138632
{chongty,tankm,tehkk,echyou,hwsun,hdtran}@i2r.a-star.edu.sg

Abstract

This paper describes the development of the automatic speech recognition (ASR) system for the submission to the VOiCES from a Distance Challenge 2019. In this challenge, we focused on the fixed condition, where the task is to recognize reverberant and noisy speech based on a limited amount of clean training data. In our system, the mismatch between the training and testing conditions was reduced by using multi-style training where the training data was artificially contaminated with different reverberation and noise sources. Also, the Weighted Prediction Error (WPE) algorithm was used to reduce the reverberant effect in the evaluation data. To boost the system performance, acoustic models of different neural network architectures were trained and the respective systems were fused to give the final output. Moreover, an LSTM language model was used to rescore the lattice to compensate the weak n-gram model trained from only the transcription text. Evaluated on the development set, our system showed an average word error rate (WER) of 27.04%.

Index Terms: speech recognition, multi-style training, de-reverberation, LSTM language model.

1. Introduction

Real speech recognition applications often operate in reverberant and noisy environment. Usually, the knowledge about the environment is unavailable in prior. Improving the robustness of the automatic speech recognition (ASR) systems has been one of the main focuses of the speech community [1, 2, 3, 4, 5]. Challenges such as VOiCES [6], ASpIRE [7], CHiME [8] and REVERB [9] have been set up to foster technologies to address such issue.

There are various approaches to robust speech recognition. Typically, multi-style training is adopted where data recorded in different background conditions are augmented for training the acoustic models [10, 11, 12, 13]. As obtaining the actual noisy data is costly, the training data is artificially corrupted with reverberation and noise of different profiles. On the other hand, speech enhancement methods are used to reduce the interference in the speech signal either by de-reverberation [14, 15, 16] or noise reduction [17, 13]. Moreover, the speech features can be engineered to alleviate the sensitivity to the recording environment [18, 19, 20], typically replacing the traditional non-linearity in the mel scale with another power-law non-linearity, e.g. power-normalized cepstral coefficients (PNCC) [18]. Also, noise information can be included into the acoustic models by appending a noise descriptor to the feature vectors, generally referred to as the noise-aware models [21, 22].

In this paper, we describe our ASR system submitted to the VOiCES from a Distance Challenge 2019 [23] on the fixed condition task. The final systems are fused from eight subsystems. Each subsystem corresponds to a specific neural network architecture in the acoustic model, generally, stacks of convolutional neural network (CNN), time delay neural network (TDNN) and long short term memory (LSTM) network. In order to enhance the audio quality, the speech is de-reverbed by using the WPE algorithm before being inputted to the ASR system. Finally, an LSTM language model is used to rescore the lattice.

In next section, we provide the details of each module in the ASR system, particularly the front-end processing, acoustic modeling, language modeling and system fusion. Also, the effectiveness of each module will be assessed on a TDNN-LSTM system before more systems are built for the fusion. Section 3 discusses the evaluation results and Section 4 concludes this paper.

2. Data Description

In this challenge, the ASR systems will be built from training data containing clean speech, but, will be fine-tuned and tested on the development data comprising only contaminated speech. Both the training and development data were extracted from the LibriSpeech corpus [24]. However the development data was re-recorded under various reverberant and noise conditions. Specifically, the speech was played-back and re-recorded in two different rooms (referred to as "rm1" and "rm2") to capture the reverberant effects. Moreover, the audio was recorded together with three types of noise, i.e. babble, music and television (referred to as "babb", "musi" and "tele"), played in the background. By placing the playback speakers in different locations in the rooms, reverberant of different characteristics is captured. The duration (in hours) of each subset in the development data is shown Table 1. On average, each subset consists of about 2.34 hours of speech. The "none" condition refers to the subset that contains only the reverberant effect but without any noise. The training set comprises 80.29 hours of clean speech.

Table 1: *Duration (hours) of each subset in the development set.*

	babb	musi	none	tele
rm1	2.19	2.27	2.29	2.36
rm2	2.49	2.37	2.38	2.38

3. System Description

Based on the given data, we identified two challenges in the task: (1) condition mismatch between the training and development data, and (2) limited amount of training data for acoustic and language modeling, i.e. 80 hours. Four strategies were formulated to overcome the problems: (1) multi-style training with data augmentation to reduce the condition mismatch, (2) de-reverberation to reduce the reverberant effect, (3) LSTM language modeling to compensate the weak n-gram model used in the decoding, and (4) system fusion to boost the system performance.

Copyright © 2019 ISCA

http://dx.doi.org/10.21437/Interspeech.2019-2130

The ASR systems were built by using the Kaldi toolkit [25]. Each module in the ASR systems is described as follows.

3.1. Multi-style Training

For acoustic modeling, multi-style training [10, 11, 12, 13] was adopted to reduce the mismatch between the training and development data. Particularly, the training data is artificially corrupted with reverberation and noise of various characteristics. A reverberant and noisy signal can be simulated as follows [11].

$$x_r[t] = x[t] * h_s[t] + \sum_i n_i[t] * h_i[t] + d[t] \qquad (1)$$

where $x_r[t]$ denotes the reverberant and noisy signal, $x[t]$ denotes source signal, $h_s[t]$ denotes the RIR corresponding to the source position, $n_i[t]$ denotes a point-source noise and $h_i[t]$ denotes the corresponding RIR, and $d[t]$ denotes other additive noise sources like isotropic noise.

We followed the procedures in [11] to induce reverberant to the training data. Also, three types of point-source noise were added: foreground noise, babble and music backgroud noise. The foreground noise was sampled from the free-sound subset in the MUSAN corpus [26]; the babble noise was from the TEDLIUM corpus [27]; the music noise was from the MUSAN corpus [26]. All RIRs were sampled from the simulated subset in [11]. As shown in [11], there is no significant difference in performance between the real and simulated RIRs after the point source noise is added.

All sources of noise are convolved with the RIRs as to simulated the reverberant effects induced to the point-source. For the babble noise, up to three different speech in the TEDLIUM corpus were added, i.e. multiple background speakers. Also, the SNR for the training data to babble noise was fixed higher, i.e. $>$ 10dB, in order to simulate closer distance from the listener to the target speaker than the background speakers.

In order to inspect if adding reverberation and noise sources are capable in reducing the mismatch, data augmentation of different settings were used to train the acoustic model. The WER results are shown in Table 2. In this experiment, the ASR system is based on a TDNN-LSTM acoustic model [28], which will be further discussed in Section 4.

As indicated by the results, data augmentation constantly improve the performance, i.e. WER reduced with respect to the number of noise sources and duplication. Also, we notice that the WER on the "babb" subset was reduced even if the music noise was added, and vice versa. Such phenomenon highlights the importance of incorporating noise of different profiles to improve the robustness of the ASR systems.

3.2. WPE De-reverberation

Reverberation is created when a sound or signal is reflected causing a large number of reflections or dispersions (wave propagation) on the surface of objects in the space. The late reverberation is the main cause for increase in the accuracy of speech recognition. Reverberation is generally modeled as the convolution of a Room Impulse Response (RIR) with the original signal denoted by

$$x_r[t] = x[t] * h_s[t] \qquad (2)$$

where $x[t]$ is the source signal, $x_r[t]$ is the signal received at the microphone at time t, and $h_s[t]$ represents the impulse of the channel from the desired source to the microphone.

We apply de-reverberation based on the Weighted Prediction Error (WPE) algorithm [14, 15] as front-end processing. This method is based on robust blind deconvolution using long-term linear prediction, with the motive of reducing the effects of the late reverberation. This method receives speech signal in the time domain follow by complex STFT to compute the coefficients of the finite impulse response (FIR) linear prediction filters with taps w iteratively. Finally, a de-reverberated time waveform is obtained by subtracts it from the observed signal denoted by

$$\hat{w} = \min_w \sum |x_r[t] - \sum_{k=0}^{N-1} \hat{w}[k] x_r[t-k-1]|^2 \qquad (3)$$

We assessed the performance of the WPE algorithm in reducing the WER. The WPE algorithm was applied on the development data and the WER results were compared to the baseline system, as shown in Table 3. The ASR system is based on a TDNN-LSTM acoustic model [28] which will be further discussed in Section 4.

The WPE algorithm improved the performance on all subset in the development data, although the WER reduction on the "babb" subset was not as high as the "musi" and "none" subsets.

We have also experimented combining the WPE algorithm with the neural network based spectral masking noise reduction techniques [29, 30]. Although applying the noise reduction technique alone slightly reduced the WER, cascading it with the WPE algorithm, in both order, did not perform better than the WPE algorithm alone.

3.3. LSTM Language Model Rescoring

For language modeling, an LSTM language model [31, 32, 33] was used to rescore the lattice produced by the ASR systems. The language model was trained by using the Kaldi-RNNLM tookit [32, 33] that gives a much faster runtime for training and testing. The toolkit also provides flexibility to include additional textual features in the model, such as sub-words which is particularly useful under sparse data situation.

We assessed the usefulness of the LSTM language model rescoring in reducing the WER. The results before and after rescoring are shown in Table 4. The LSTM language model was built based on the most frequent 10K words and additional 1K subword features. The ASR system is based on a TDNN-LSTM acoustic model which will be further discussed in Section 4.

The WER was reduced drastically up to 20.88%, significantly higher than the reduction contributed by data augmentation and de-reverberation. Such phenomenon is due to the weak n-gram model used in the decoding. Since there are only 80 hours of transcription text available for the training, the traditional n-gram model is hurt badly by the severe data scarcity problem. The neural network language model, which operates in the lower dimensional space and subword features, is more resistant to the data scarcity problem and predicts words more accurately.

3.4. Systems Fusion

Based on different network architectures and configurations in the acoustic models, multiple ASR systems were built and combined to give the final hypothesis. Specifically, the lattice produced by the ASR systems were combined and decoded to give the minimum Bayes risk (MBR) estimate, i.e. minimum word error [34]. As the acoustic information is captured differently by the subsystems, lattice combination allows the search space

Table 2: *Data augmentation constantly reduced the WER as more noise sources and duplication were used.*

	rm1				rm2			
	babb	**musi**	**none**	**tele**	**babb**	**musi**	**none**	**tele**
RIR with 3 x noise	50.05	47.47	40.24	48.48	50.93	48.46	39.04	47.73
RIR with 3 x noise + 1 x music	46.5	43.99	38.96	45.06	46.62	45.89	38.03	44.71
RIR with 3 x noise + 1 x music + 1 x babble	44.32	42.48	38.22	43.58	43.87	44.36	36.66	43.1
RIR with 3 x noise + 3 x music + 3 x babble	41.83	40.79	36.56	41.17	40.81	42.23	34.92	40.19

Table 3: *De-reverberation by using the WPE algorithm reduced the WER by 4.5% on average.*

	rm1				rm2			
	babb	**musi**	**none**	**tele**	**babb**	**musi**	**none**	**tele**
Baseline	50.05	47.47	40.24	48.48	50.93	48.46	39.04	47.73
De-reverberation	48.16	43.86	37.96	46.15	49.16	45.99	37.09	45.86
Reduction (%)	3.78	7.6	5.67	4.81	3.48	5.1	4.99	3.92

Table 4: *Lattice rescoring by using the LSTM language model reduced the WER by 17.45% on average.*

	rm1				rm2			
	babb	**musi**	**none**	**tele**	**babb**	**musi**	**none**	**tele**
Baseline	50.05	47.47	40.24	48.48	50.93	48.46	39.04	47.73
LSTM	42.21	39.27	32.55	39.85	42.69	41.11	30.89	39.37
Reduction (%)	15.66	17.27	19.11	17.8	16.18	15.17	20.88	17.52

to be expanded to include words and arcs which might not be hypothesized by a single system.

4. Results and Discussions

In the final submission, eight subsystems were deployed, each corresponds to a specific neural network architecture in the acoustic model. The features fed to the acoustic models consist of 40-dimension of mel-frequency cepstral coefficients (MFCC) and 100-dimension of i-vector. Generally, architectures such as convolutional neural network (CNN), time delay neural network (TDNN) and long short term memory (LSTM) neural network were stacked to give more accurate results [28]. For certain LSTM layers, we applied the attention mechanism. Also, in one of the LSTM-TDNN models, we explored the variation of backstitching training [35]. The network configurations in the acoustic models are shown in Table 5.

All acoustic models were trained by using the multi-style training as dicussed in Section 3.1. For each type of noise, the training data was duplicated three folds, as indicated by the last row in Table 2. Also, the training data was duplicated three folds again by adjusting the speaking rates, i.e. speed perturbance with the scales of 0.9, 1.0 and 1.1 [36]. Finally, the size of the given training data was multiplied 27x larger in total, i.e. 2,160 hours.

The subsystems were fused based on three different weighting schemes, referred to as w_1, w_2 and w_3. The weights were determined arbitrarily based on the single system performance (as will be shown in Table 6, 7 and 8), i.e. higher weight was assigned to lower WER systems, and vice versa. We avoided fine-tuning the weights by using the development data in order to keep the configuration general enough for the evaluation data.

In Table 6, the WER of the subsystem and fused systems are shown. Model fusion constantly gives lower WER than any single subsystem alone. These results will be used as the baseline for the follow up processes.

Table 5: *Weighting scheme for lattice combination.*

	w1	w2	w3
CNN-TDNN	0.08	0.03	0.00
CNN-TDNN-LSTM	0.2	0.25	0.25
TDNN[1]	0.08	0.01	0.00
TDNN-BLSTM	0.08	0.1	0.125
TDNN-LSTM	0.2	0.25	0.25
TDNN-LSTM[2]	0.08	0.1	0.125
TDNN-LSTM[3]	0.2	0.25	0.25
TDNN-LSTM[2,3]	0.08	0.01	0.00

[1]: self attention
[2]: backstitching training
[3]: attention LSTM

Table 7 shows the contribution of the WPE de-reverberation front-end processing. In all considered cases, the WER was reduced by about 2%. Finally, the LSTM language model rescoring further reduced the WER by about 6% on absolute (see Table 8). The results are consistent to the ones obtained earlier as in Table 3 and 4.

5. Conclusion

In this paper, we have described the development of the ASR systems for the submission to the VOiCES from a Distance Challange 2019. In order to tackle the condition mismatch and sparse training data problems, we adopted four strategies: (1) multi-style training for acoustic modeling, (2) WPE de-reverberation front-end processing (3) LSTM language model lattice rescoring, and (4) system fusion. In this fixed condition task, data augmentation has shown to drastically improve the quality of the acoustic models. Also, lattice rescoring by using the LSTM language model much alleviated the data scarcity problem and contributed drastic reduction to the WER. Overall, the ASR systems showed an average WER of 27.04%.

Table 6: *The baseline performance of the fused systems and their respective subsystems. Fusing the system constantly gives lower WER.*

	rm1				**rm2**				
	babb	**musi**	**none**	**tele**	**babb**	**musi**	**none**	**tele**	**average**
CNN-TDNN	48.93	47.36	43.37	47.96	47.84	48.75	41.07	46.85	46.52
CNN-TDNN-LSTM	41.24	40.02	35.89	40.33	39.9	40.76	34	38.89	38.88
TDNN[1]	52.87	51.53	47	51.62	51.38	52.83	44.94	50	50.27
TDNN-BLSTM	45.7	44.49	40.02	45.06	44.51	45.7	37.93	43.51	43.37
TDNN-LSTM	41.83	40.79	36.56	41.17	40.81	42.23	34.92	40.19	39.81
TDNN-LSTM[2]	45.98	44.24	38.54	44.64	45.5	47.65	37.2	44.2	43.49
TDNN-LSTM[3]	42.28	40.53	36.02	41.29	40.86	42.47	34.85	40.56	39.86
TDNN-LSTM[2,3]	51.7	50.84	45.61	50.54	50.98	52.06	43.54	50.25	49.44
fusion: w_1	37.17	36.31	32.02	36.39	36.29	37.63	30.45	35.47	35.22
fusion: w_2	36.86	35.97	31.69	36.18	35.78	37.04	30.27	35.18	34.87
fusion: w_3	36.91	35.81	31.65	36.14	35.86	37.14	30.19	35.16	34.86

Table 7: *WPE de-reverberation reduced the WER about 2% in all considered cases.*

	rm1				**rm2**				
	babb	**musi**	**none**	**tele**	**babb**	**musi**	**none**	**tele**	**average**
CNN-TDNN	46.22	44.56	41.03	45.58	46.15	46.25	39.18	45.15	44.27
CNN-TDNN-LSTM	39.31	37.38	33.92	38.16	38.44	38.73	32.33	37.3	36.95
TDNN[1]	50.22	48.1	44.37	48.94	49.2	50.13	42.71	48.1	47.72
TDNN-BLSTM	43.67	41.42	37.93	42.76	42.8	43.34	36.26	41.73	41.24
TDNN-LSTM	40.01	38.39	34.93	39.16	39	40.32	33.41	38.48	37.96
TDNN-LSTM[2]	43.6	41.04	36.48	42.25	43.58	44.41	35.18	42.08	41.08
TDNN-LSTM[3]	39.93	37.87	34.14	38.9	39.12	40.39	33.16	38.76	37.78
TDNN-LSTM[2,3]	48.71	47.2	42.75	48.06	48.96	49.29	41.19	48.26	46.8
fusion: w_1	35.4	33.61	30.2	34.65	34.71	35.53	28.93	33.89	33.37
fusion: w_2	35.19	33.14	30.03	34.6	34.36	35.05	28.71	33.64	33.09
fusion: w_3	35.12	33.14	30.04	34.54	34.43	35.08	28.6	33.53	33.06

Table 8: *Lattice rescoring by using the LSTM language model reduces the WER by about 6% on average.*

	rm1				**rm2**				
	babb	**musi**	**none**	**tele**	**babb**	**musi**	**none**	**tele**	**average**
CNN-TDNN	38.21	36.16	31.78	36.58	37.91	38.03	30.6	36.27	35.69
CNN-TDNN-LSTM	32.92	30.9	26.77	31.23	31.82	32.42	25.49	30.38	30.24
TDNN[1]	42.11	39.35	35.1	39.73	41.49	42.46	33.59	39.25	39.14
TDNN-BLSTM	36.27	34	29.84	34.58	35.49	36.25	28.76	33.99	33.65
TDNN-LSTM	32.8	30.66	26.86	30.83	31.93	33.11	25.23	30.88	30.29
TDNN-LSTM[2]	39.02	35.75	30.49	36.65	39.04	39.58	29.3	37.29	35.89
TDNN-LSTM[3]	32.79	30.25	26.54	30.63	32.06	32.61	25.29	30.72	30.11
TDNN-LSTM[2,3]	43.52	41.69	36.7	42.13	43.31	43.13	35.08	42.21	40.97
fusion: w_1	30	27.48	23.69	27.93	29.21	29.9	22.6	27.71	27.32
fusion: w_2	29.52	27.04	23.53	27.45	28.85	29.47	22.33	27.36	26.94
fusion: w_3	29.63	27.11	23.62	27.56	28.91	29.57	22.41	27.48	27.04

[1]: self attention
[2]: backstitching training
[3]: attention LSTM

6. References

[1] J. Dennis and T. H. Dat, "Single and multi-channel approaches for distant speech recognition under noisy reverberant conditions: I2R'S system description for the ASpIRE challenge," in *ASRU*, 2016, pp. 518–524.

[2] Z. Chen, S. Watanabe, H. Erdogan, and J. R. Hershey, "Speech enhancement and recognition using multi-task learning of long short-term memory recurrent neural networks," in *INTERSPEECH*, 2015, pp. 3274–3278.

[3] Z. Q. Wang and D. L. Wang, "A joint training framework for robust automatic speech recognition," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 24, no. 4, pp. 796–806, 2016.

[4] R. Hsiao, J. Ma, W. Hartmann, M. Karafiát, F. Grézl, L. Burget, I. Szöke, J. H. Černocky, S. Watanabe, Z. Chen, S. H. Mallidi, H. Hermansky, S. Tsakalidis, and R. Schwartz, "Robust speech recognition in unknown reverberant and noisy conditions," in *ASRU*, 2016, pp. 533–538.

[5] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech and Language*, vol. 46, pp. 535–557, 2017.

[6] C. Richey, M. A. Barrios, Z. Armstrong, C. Bartels, H. Franco, M. Graciarena, A. Lawson, M. K. Nandwana, A. Stauffer, J. Van Hout, P. Gamble, J. Hetherly, C. Stephenson, and K. Ni, "Voices Obscured in Complex Environmental Settings (VOiCES) corpus," in *INTERSPEECH*, 2018, pp. 1566–1570.

[7] M. Harper, "The Automatic Speech recogition in Reverberant Environments (ASpIRE) challenge," *in Proc. ASRU*, pp. 547–554, 2016.

[8] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *INTERSPEECH*, 2018, pp. 1561–1565.

[9] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, "The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *WASPAA*, 2013, pp. 1–4.

[10] R. Lippmann, E. Martin, and D. Paul, "Multi-style training for robust isolated-word speech recognition," in *ICASSP*, 2005, pp. 705–708.

[11] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *ICASSP*, 2017, pp. 5220–5224.

[12] M. Doulaty, R. Rose, and O. Siohan, "Automatic optimization of data perturbation distributions for multi-style training in speech recognition," in *SLT*, 2017, pp. 21–27.

[13] C. Donahue, B. Li, and R. Prabhavalkar, "Exploring speech enhancement with generative adversarial networks for robust speech recognition," in *ICASSP*, 2018, pp. 5024–5028.

[14] M. M. T. Yoshioka, T. Nakatani and H. G. Okuno, "Blind separation and dereverberation of speech mixtures by joint optimization," *IEEE Transaction on Audio, Speech, and Language Processing*, vol. 19, no. 1, pp. 69–84, 2011.

[15] X. C. Takuya Yoshioka and M. J. F. Gales, "Impact of single-microphone dereverberation on dnn-based meeting transcription systems," in *ICASSP*, 2014, pp. 5527–5531.

[16] D. S. Williamson and D. Wang, "Speech dereverberation and denoising using complex ratio masks," in *ICASSP*, 2017, pp. 5590–5594.

[17] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," *Lecture Notes in Computer Science*, vol. 9237, pp. 91–99, 2015.

[18] C. Kim and R. M. Stern, "Power-Normalized Cepstral Coefficients (PNCC) for Robust Speech Recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 24, no. 7, pp. 1315–1329, 2016.

[19] V. Mitra, G. Sivaraman, H. Nam, C. Espy-Wilson, E. Saltzman, and M. Tiede, "Hybrid convolutional neural networks for articulatory and acoustic information based speech recognition," *Speech Communication*, vol. 89, pp. 103–112, 2017.

[20] W. N. Hsu and J. Glass, "Extracting Domain Invariant Features by Unsupervised Learning for Robust Automatic Speech Recognition," in *ICASSP*, 2018, pp. 5614–5618.

[21] R. Giri, M. L. Seltzer, J. Droppo, and D. Yu, "Improving speech recognition in reverberation using a room-aware deep neural network and multi-task learning," in *ICASSP*, 2015, pp. 5014–5018.

[22] Y. Qian, T. Tan, and D. Yu, "Neural Network Based Multi-Factor Aware Joint Training for Robust Speech Recognition," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 24, no. 12, pp. 2231–2240, 2016.

[23] M. K. Nandwana, J. van Hout, M. McLaren, C. Richey, A. Lawson, and M. A. Barrios, "The VOiCES from a Distance Challenge 2019 Evaluation Plan," 2019. [Online]. Available: http://arxiv.org/abs/1902.10828

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[25] 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 *ASRU*, 2011.

[26] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," 2015, arXiv:1510.08484v1.

[27] A. Rousseau, P. Deléglise, and Y. Estève, "TED-LIUM: an Automatic Speech Recognition dedicated corpus." in *LREC*, 2012, pp. 125–129.

[28] V. Peddinti, Y. Wang, D. Povey, and S. Khudanpur, "Low Latency Acoustic Modeling Using Temporal Convolution and LSTMs," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 373–377, 2018.

[29] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *ICASSP*, 2013, pp. 7092–7096.

[30] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *ICASSP*, vol. 2016-May, 2016, pp. 196–200.

[31] M. Sundermeyer, R. Schl, and H. Ney, "LSTM neural networks for language modeling," in *INTERSPEECH*, 2012, pp. 194–197.

[32] H. Xu, T. Chen, D. Gao, Y. Wang, K. Li, N. Goel, Y. Carmiel, D. Povey, and S. Khudanpur, "A Pruned Rnnlm Lattice-Rescoring Algorithm for Automatic Speech Recognition," in *ICASSP*, vol. 2018-April, 2018, pp. 5929–5933.

[33] H. Xu, K. Li, Y. Wang, J. Wang, S. Kang, X. Chen, D. Povey, and S. Khudanpur, "Neural Network Language Modeling with Letter-Based Features and Importance Sampling," in *ICASSP*, vol. 2018-April, 2018, pp. 6109–6113.

[34] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum Bayes Risk decoding and system combination based on a recursion for edit distance," *Computer Speech and Language*, vol. 25, no. 4, pp. 802–828, 2011.

[35] Y. Wang, V. Peddinti, H. Xu, X. Zhang, D. Povey, and S. Khudanpur, "Backstitch: Counteracting finite-sample bias via negative steps," in *INTERSPEECH*, vol. 2017-Augus, 2017, pp. 1631–1635.

[36] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *INTERSPEECH*, 2015, pp. 3586–3589.